

N. R. Shetty
L. M. Patnaik
N. H. Prasad *Editors*

Emerging Research in Computing, Information, Communication and Applications

Proceedings of ERCICA 2022

Lecture Notes in Electrical Engineering

Volume 928

Series Editors

Leopoldo Angrisani, Department of Electrical and Information Technologies Engineering, University of Napoli Federico II, Naples, Italy

Marco Arteaga, Departament de Control y Robótica, Universidad Nacional Autónoma de México, Coyoacán, Mexico

Bijaya Ketan Panigrahi, Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, Delhi, India

Samarjit Chakraborty, Fakultät für Elektrotechnik und Informationstechnik, TU München, Munich, Germany

Jiming Chen, Zhejiang University, Hangzhou, Zhejiang, China

Shanben Chen, Materials Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

Tan Kay Chen, Department of Electrical and Computer Engineering, National University of Singapore, Singapore, Singapore

Rüdiger Dillmann, Humanoids and Intelligent Systems Laboratory, Karlsruhe Institute for Technology, Karlsruhe, Germany

Haibin Duan, Beijing University of Aeronautics and Astronautics, Beijing, China

Gianluigi Ferrari, Università di Parma, Parma, Italy

Manuel Ferre, Centre for Automation and Robotics CAR (UPM-CSIC), Universidad Politécnica de Madrid, Madrid, Spain

Sandra Hirche, Department of Electrical Engineering and Information Science, Technische Universität München, Munich, Germany

Faryar Jabbari, Department of Mechanical and Aerospace Engineering, University of California, Irvine, CA, USA

Limin Jia, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Alaa Khamis, German University in Egypt El Tagamoa El Khames, New Cairo City, Egypt

Torsten Kroeger, Stanford University, Stanford, CA, USA

Yong Li, Hunan University, Changsha, Hunan, China

Qilian Liang, Department of Electrical Engineering, University of Texas at Arlington, Arlington, TX, USA

Ferran Martín, Departament d'Enginyeria Electrònica, Universitat Autònoma de Barcelona, Bellaterra, Barcelona, Spain

Tan Cher Ming, College of Engineering, Nanyang Technological University, Singapore, Singapore

Wolfgang Minker, Institute of Information Technology, University of Ulm, Ulm, Germany

Pradeep Misra, Department of Electrical Engineering, Wright State University, Dayton, OH, USA

Sebastian Möller, Quality and Usability Laboratory, TU Berlin, Berlin, Germany

Subhas Mukhopadhyay, School of Engineering and Advanced Technology, Massey University,

Palmerston North, Manawatu-Wanganui, New Zealand

Cun-Zheng Ning, Department of Electrical Engineering, Arizona State University, Tempe, AZ, USA

Toyooki Nishida, Graduate School of Informatics, Kyoto University, Kyoto, Japan

Luca Oneto, Department of Informatics, Bioengineering, Robotics and Systems Engineering, University of Genova, Genova, Italy

Federica Pascucci, Dipartimento di Ingegneria, Università degli Studi "Roma Tre", Rome, Italy

Yong Qin, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Gan Woon Seng, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, Singapore

Joachim Speidel, Institute of Telecommunications, Universität Stuttgart, Stuttgart, Germany

Germano Veiga, Campus da FEUP, INESC Porto, Porto, Portugal

Haitao Wu, Academy of Opto-electronics, Chinese Academy of Sciences, Beijing, China

Walter Zamboni, DIEM—Università degli studi di Salerno, Fisciano, Salerno, Italy

Junjie James Zhang, Charlotte, NC, USA

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering—quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Editorial Director (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** This series is indexed by EI Compendex and Scopus databases. ****

N. R. Shetty · L. M. Patnaik · N. H. Prasad
Editors

Emerging Research in Computing, Information, Communication and Applications

Proceedings of ERCICA 2022

 Springer

Editors

N. R. Shetty
Nitte Meenakshi Institute of Technology
Bengaluru, Karnataka, India

L. M. Patnaik
National Institute of Advanced Studies
(NIAS)
Bengaluru, Karnataka, India

N. H. Prasad
Nitte Meenakshi Institute of Technology
Bengaluru, Karnataka, India

ISSN 1876-1100

ISSN 1876-1119 (electronic)

Lecture Notes in Electrical Engineering

ISBN 978-981-19-5481-8

ISBN 978-981-19-5482-5 (eBook)

<https://doi.org/10.1007/978-981-19-5482-5>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Organizing Committee

ERCICA-2022

The Seventh International Conference on “Emerging Research in Computing, Information, Communication and Applications,” (ERCICA 2022), was held online during February 25–26, 2022, at Nitte Meenakshi Institute of Technology (NMIT), Bengaluru.

ERCICA 2022—Committees

Chief Patrons

Dr. N. V. Hegde, President, Nitte Education Trust, Mangalore, India

Dr. N. R. Shetty, Chancellor-Central University of Karnataka, Kalburgi, and Advisor, Nitte Education Trust, Mangalore, India

Conference Chair

Dr. H. C. Nagaraj, Principal, NMIT, Bengaluru, India

Program Chairs

Dr. N. Hamsavath Prasad, HOD, MCA, NMIT, Bengaluru, India

Dr. N. Nalini, Professor, CSE, NMIT, Bengaluru, India

Publication

Springer-LNEE Series

Advisory Chairs

Dr. K. Sudha Rao, Advisor-Admin and Management, NMIT, Bengaluru, India

Mr. Rohit Punja, Administrator, NET, Mangalore, India

Dr. V. Sridhar, Dean (Academic), NMIT, Bengaluru, India

Advisory Committee

Dr. L. M. Patnaik, INSA Senior Scientist, NIAS, Bengaluru, India

Dr. K. D. Nayak, Former OS&CC, R&D (MED&MIST), DRDO, India

Dr. Kalidas Shetty, Founding Director of Global Institute of Food Security and International Agriculture (GIFSIA), North Dakota State University, Fargo, USA

Dr. Sathish Udpa, Dean and Professor, Michigan State University, Michigan, USA

Dr. Vincenzo Piuri, 2021–2022 IEEE Region Eight Director-Elect, Università degli Studi di Milano, Dipartimento Di Informatica Via Celoria 18, 20133, Milan, Italy

Dr. K. N. Bhat, Visiting Professor, Center for Nano Science and Engineering-CeNSE, IISc, Bengaluru, India

Dr. Karisiddappa, Vice-Chancellor, Visvesvaraya Technological University, Belagavi, Karnataka, India

Dr. K. R. Venugopal, Vice-Chancellor, Bangalore University, Bengaluru, India

Prof. Sonajharia Minz, Vice-Chancellor, Sido Kanhu Murmu University, Dumka, Jharkhand, India

Dr. Navakanta Bhat, Chairperson, Center for Nano Science and Engineering-CeNSE, IISc, Bengaluru, India

Shri. M. P. Dubey, Joint Director, Software Technology Parks of India-Visakhapatnam, Ministry of E&IT, Government of India

Dr. Narushan Pillay, School of Engineering, Howard College Campus, UKZN, South Africa

Mrs. Rajnish Meenalochani, Engineering Director, Unisys, Bengaluru, India

Program Committee

Dr. Ankit Singhal, Power System Research Engineer, Pacific Northwest National Laboratory, USA

Dr. M. A. Ajay Kumara, D&H Schort School of Computing Sciences and Mathematics, Lenoir-Rhyne University, Hickory, NC, USA

Hardik Gohel, Department of Computer Science, School of Art and IT Sciences, University of Houston, Victoria, TX, USA

Dr. Sanjeev K. Cowlessur, Department of Software Engineering, Faculty of Information and Communication Technology, Université des Mascareignes, Beau Plan, Pamplémousses, Mauritius

Mr. Sreenivas Divi, Director of IT and Product Management, 46030, Manekin Plaza, Suite 150, Sterling, VA 20166, USA

Mr. Hemant Gaur, Principal Program Manager, Microsoft Power Apps Redmond , WA 98054, USA

Prof. Subarna Shakya, Professor, Department of Electronics and Computer engineering, Pulchowk Campus, Institute of Engineering, Tribhuvan University, Pulchowk, Lalitpur, Nepal

Prof. Savitri Bevinakoppa, School of Information Technology and Engineering (SITE) |Academic Department, The Argus, Level 6, 284–294 La Trobe Street, Melbourne Victoria 3000, Australia

Dr. Tom Wallingo, School of Electrical, Electronics and Computer Engineering, Howard College Campus, University of KwaZulu-Natal, Durban, South Africa

Organizing Co-Chairs

Dr. M. N. Thippeswamy, Professor and Head, Department of CSE, NMIT, Bengaluru, India

Dr. H. A. Sanjay, Professor and Head, Department of ISE, NMIT, Bengaluru, India

Dr. A. C. Ramachandra, Professor and Head, Department of ECE, NMIT, Bengaluru, India

Dr. N. Hamsavath Prasad, Professor and Head, Department of MCA, NMIT, Bengaluru, India

Dr. Vasudha Hegde, Professor and Head, Department of EEE, NMIT, Bengaluru, India

Dr. N. Nalini., Professor, Department of CSE, NMIT, Bengaluru, India

Preface

The Seventh International Conference on “Emerging Research in Computing, Information, Communication and Applications,” ERCICA 2022, is an annual event organized online at the Nitte Meenakshi Institute of Technology (NMIT), Yelahanka, Bengaluru, India.

ERCICA aims to provide an interdisciplinary forum for discussion among researchers, engineers and scientists to promote research and exchange of knowledge in computing, information, communications and related applications. This conference will provide a platform for networking of academicians, engineers and scientists and also will encourage the participants to undertake high-end research in the above thrust areas.

ERCICA 2022 received more than 300 papers from all over the world, viz. from China, UK, Africa, Saudi Arabia and India. The ERCICA Technical Review Committee has followed all necessary steps to screen more than 300 papers by going through six rounds of quality checks on each paper before selection for presentation/publication in Springer’ LNEE (Lecture Notes in Electrical Engineering) proceedings, is SCOPUS indexed.

The acceptance ratio is 1:4

Bengaluru, India
September 2022

N. R. Shetty
L. M. Patnaik
N. H. Prasad

Acknowledgments

First of all, we would like to thank Professor N. R. Shetty who has always been the guiding force behind this event's success. It was his dream that we have striven to make a reality. Our thanks to Dr. H. C. Nagaraj, who has monitored the whole activity of the conference from the beginning till its successful end.

Our special thanks to Springer and especially the editorial staff who were patient, meticulous and friendly with their constructive criticism on the quality of papers and outright rejection at times without compromising the quality of the papers as they are always known for publishing the best international papers.

We would like to express our gratitude to all the review committee members of all the themes of computing, information, communication and applications and the best paper award review committee members.

Finally, we would like to express our heartfelt gratitude and warmest thanks to the ERCICA 2022 organizing committee members for their hard work and outstanding efforts. We know how much time and energy this assignment demanded, and we deeply appreciate all the efforts to make it a grand success.

Our special thanks to all the authors who have contributed to publish their research work in this conference and participated to make this conference a grand success. Thanks to everyone who have directly or indirectly contributed to the success of this conference ERCICA 2022.

Program Chairs
ERCICA 2022

About the Conference

ERCICA 2022

The Seventh International Conference on “Emerging Research in Computing, Information, Communication and Applications,” ERCICA 2022, is an annual event held online during February 25–26, 2022, at Nitte Meenakshi Institute of Technology (NMIT), Yelahanka, Bengaluru, India.

ERCICA 2022 is organized under the patronage of Prof. N. R. Shetty, Advisor, Nitte Education Trust. Dr. H. C. Nagaraj, Principal, served as Conference Chair, and Program Chairs of the conference were Dr. N. H. Prasad, Professor and Head, MCA, and Dr. N. Nalini, Professor, CSE, NMIT, Bengaluru, Karnataka.

ERCICA aims to provide an interdisciplinary forum for discussion among researchers, engineers and scientists to promote research and exchange of knowledge in computing, information, communications and related applications. This conference will provide a platform for networking of academicians, engineers and scientists and also will encourage the participants to undertake high-end research in the above thrust areas.

For ERCICA-2023, authors are invited to submit the manuscripts of their original and unpublished research contributions to ercica.chair@gmail.com (ERCICA website: <https://nmit.ac.in/ercica/ercica.html>). All the submitted papers will go through a peer review process, and the corresponding authors will be notified about the outcome of the review process. There will be six rounds of quality checks on each paper before selection for presentation/publication. Authors of the selected papers may present their papers during the conference.

Contents

Teaching Learning-Based Optimization with Learning Enthusiasm Mechanism for Optimal Control of PV Inverters in Utility Grids for Techno-Economic Goals	1
Nirmala John, Varaprasad Janamala, and Joseph Rodrigues	
Machine Learning Framework for Prediction of Parkinson’s Disease in Cloud Environment	15
K. Aditya Shastry, V. Sushma, Naman Bansal, Ujjwal Saxena, Shrey Srivastava, and Suvang Samal	
A Comparative Analysis to Measure Scholastic Success of Students Using Data Science Methods	27
Saleem Malik, K. Jothimani, and U. J. Ujwal	
Gesture-Controlled Speech Assist Device for the Verbally Disabled	43
Shreeram V. Kulkarni, Shruti Gatade, Vasudha Hegde, and G. Manohar	
Highly Classified with Two Factor Authentication Encrypted Secured Mail	53
N. H. Prasad, B. N. Lakshmi Narayan, and S. GovindaKishora	
Efficacious Intrusion Detection on Cloud Using Improved BES and HYBRID SKINET-EKNN	61
C. U. Om Kumar, Ponsy R. K. Sathia Bhama, and Prasad	
An Amalgamated and Personalized System for the Prognosis and Detecting the Presence of Parkinson’s Disease at Its Early Onset	73
K. Harshitha, T. R. Vinay, K. Keerti, and M. Shreya	

Design and Implementation of Flyback Converter Topology for Dual DC Outputs 87
C. H. V. Ramesh, Sudeep Shetty, Shreeram V. Kulkarni, and Rajkiran Ballal

Gesture Detection Using Accelerometer and Gyroscope 99
Raghav Gupta, Shashank Chaudhary, Akshat Vedant, Niladri Paul Choudhury, and Vandana Ladwani

To Monitor Yoga Posture Without Intervention of Human Expert Using 3D Kinematic Pose Estimation Model—A Bottom-Up Approach 117
A. V. Navaneeth and M. R. Dileep

Upshot and Disparity of AI Allied Approaches Over Customary Techniques of Assessment on Chess—An Observation 127
A. V. Navaneeth and M. R. Dileep

Network Intrusion Detection Using Neural Network Techniques 137
B. A. Manjunatha, Aditya Shastry, G. Nishchala, N. Pavithra, and S. Raheela Banu

Performing Cryptanalysis on the Secure Way of Communication Using Purple Cipher Machine 149
V. P. Srinidhi, B. Vineetha, K. Shabarinath, and Prasad B. Honnavali

Artificial Intelligence and Machine Learning for Foundry Industry—A Case Study of Belagavi Foundry Industry 161
Praveen M. Kulkarni, Prayag Gokhale, L. V. Appasaba, K. Lakshminarayana, and Basavaraj S. Tigadi

Conversion of Sign Language to Text Using Machine Learning 175
Aishwarya Bhagwat, Poonam Gupta, and Nivedita Kadam

A Hexagonal Sierpinski Fractal Antenna for Multiband Wireless Applications 189
Mahesh Mathpati, Veerendra Dakulagi, K. S. Sheshidhara, Mohammed Bakhar, H. K. Bhaladar, A. A. Jadhav, and Dhanashree Yadav

Multi-level Hierarchical Information-Driven Risk-Sensitive Routing Protocol for Mobile-WSN: MHIR-SRmW 195
B. V. Shruti and M. N. Thippeswamy

Target Classification Using CNN-LSTM Network with Reduced Sample Size in Surveillance Radar 221
Vinit R. Waingankar, Vijay Surya Vempati, Santhosh, and G. Malarkannan

Iron Oxide Nanoparticle Image Analysis Using Machine Learning Algorithms 233
 Parashuram Bannigidad, Namita Potraj,
 Prabhuodeyara Gurubasavaraj, and Lakkappa Anigol

Bankruptcy Prediction Using Bi-Level Classification Technique 241
 Abhinav Antani, B. Annappa, Shubham Dodia,
 and M. V. Manoj Kumar

Infant Brain MRI Segmentation Using Deep Volumetric U-Net with Gamma Transformation 251
 Gunda Sai Yeshwanth, B. Annappa, Shubham Dodia,
 and M. V. Manoj Kumar

Analysis of Deep Learning Architecture-Based Classifier for the Cervical Cancer Classification 263
 R. Chandraprabha and Seema Singh

Covid Vaccine Adverse Side-Effects Prediction with Sequence-to-Sequence Model 275
 Shyam Zacharia and Ashwini Kodipalli

Comparison Between ResNet 16 and Inception V4 Network for COVID-19 Prediction 283
 P. J. Rachana, Ashwini Kodipalli, and Trupthi Rao

Computational Deep Learning Models for Detection of COVID-19 Using Chest X-Ray Images 291
 Srirupa Guha, Ashwini Kodipalli, and Trupthi Rao

An Ensemble Approach for Detecting Malaria Using Classification Algorithms 307
 S. Ruban, A. Naresh, and Sanjeev Rai

IoT-Enabled Intelligent Home Using Google Assistant 317
 Balarama Krishna Veeramalla, S. Aruna, and K. Srinivasa Naik

Analysis of a Microstrip Log-Periodic Dipole Antenna with Different Substrates 327
 Suresh Prasad, S. Aruna, and K. Srinivasa Naik

Detection of Diabetic Retinopathy Using Fundus Images 339
 S. V. Viraktamath, Deepak Hiremath, and Kshama Tallur

Artificial Intelligence in the Tribology: Review 351
 Manoj Rajankunte Mahadeshwara, Santosh Kumar,
 and Anushree Ghosh Dastidar

House Price Prediction Using Advanced Regression Techniques 369
 Hemin Vasani, Harshil Gandhi, Shrey Panchal, and Shakti Mishra

Product Integrity Maintenance and Counterfeit Avoidance System Based on Blockchain	383
Sagar Ramesh Pujar, Girish R. Deshpande, S. T. Naitik, Raghavendra Vijay Pail, and B. Naveenkumar	
Efficient Building Fire Detection Using Synergistic Interaction of Activation Functions in Artificial Neural Network	397
Tanushree Roy and Saikat Kumar Shome	
Performance Evaluation of Spectral Subtraction with VAD and Time–Frequency Filtering for Speech Enhancement	407
G. Thimmaraja Yadava, B. G. Nagaraja, and H. S. Jayanna	
A Unified Libraries for GDI Logic to Achieve Low-Power and High-Speed Circuit Design	415
Jebashini Ponnian, Senthil Pari, Uma Ramadass, and Ooi Chee Pun	
Detection of Diabetic Retinopathy Using Convolution Neural Network	427
K. S. Swarnalatha, Ullal Akshatha Nayak, Neha Anne Benny, H. B. Bharath, Daivik Shetty, and S. Dileep Kumar	
Drone-Based Security Solution for Women: DroneCop	441
Sukanya Bharati, T. R. Vinay, M. G. Prasanna, N. M. Sangeetha, and Shreya Roy	
Analysis of Granular Parakeratosis Lesion Segmentation: BCE U-Net vs SOTA	455
Sheetal Janthakal and Girisha Hosalli	
Real-Time Health Monitoring of Relays and Circuit Breakers	467
S. Harshitha, B. S. Arpitha, H. Shwetha, N. Sinchana, B. Smitha, and M. J. Nagraj	
AI-Based Live-Wire News Categorization	475
S. Jagdeesh Patil, Aashna Sinha, M. M. Anusha Jadav, and Nidhi	
Implementation of STFT for Auditory Compensation on FPGA	483
S. L. Pinjare, B. R. Rajeev, Kajal Awasthi, and M. B. Vikas	
Smart Headgear for Unsafe Operational Environment	499
P. Dhanush, S. Jagdeesh Patil, R. U. Girish, G. Chethan, and S. K. Chethan	
IMPROVE the Solar Panel Proficiency by Using of Free Energy from Street Light	509
A. Saravanan and N. Sivaramakrishnan	
Hardware Implementation of Machine Vision System for Component Detection	519
P. Smruthi, K. B. Prajna, Jibin G. John, and Aslam Taj Pasha	

DEOMAC—Decentralized and Energy-Efficient Framework for Offloading Mobile Applications to Clouds 537
A. L. Shanthi and V. Ramesh

A Novel Approach for Identification of Healthy and Unhealthy Leaves Using Scale Invariant Feature Transform and Shading Histogram-PCA Techniques 549
K. S. Shashidhara, H. Girish, M. C. Parameshwara, B. Karunakara Rai, and Veerendra Dakulagi

A Comprehensive Review on the Issue of Class Imbalance in Predictive Modelling 557
Prashanth P. Wagle and M. V. Manoj Kumar

An Ameliorate Analysis of Cryptocurrencies to Determine the Trading Business with Deep Learning Techniques 577
Neeshad Kumar Sakure, M. V. Manoj Kumar, B. S. Prashanth, H. R. Sneha, and Likewin Thomas

Gender Prediction Using Iris Features 587
Bhuvaneshwari Patil and Mallikarjun Hangarge

Hardware Implementation of an Activation Function for Neural Network Processor 597
Shilpa Mayannavar and Uday Wali

Using Big Data and Gamification to Incentivize Sustainable Urban Transportation 609
A. Mariyan Richard and Prasad N. Hamsavath

Optical Character Recognition System of Telugu Language Characters Using Convolutional Neural Networks 615
K. V. Charan and T. C. Pramod

Implementation of Python in the Optimization of Process Parameters of Product Laryngoscope Manufactured in the Injection Mold Machine 625
Balachandra P. Shetty, J. Sudheer Reddy, B. A. Praveena, and A. Madhusudhan

A Practical Approach to Software Metrics in Beehive Requirement Engineering Process Model 635
K. S. Swarnalatha

Suitability of Process Models for Software Development 643
K. S. Swarnalatha

Solving Problems of Large Codebases: Uber’s Approach Using Microservice Architecture	653
K. S. Swarnalatha, Adithya Mallya, G. Mukund, and R. Ujwal Bharadwaj	
Circular Economy with Special Reference to Electrical and Electronic Waste Management in India	663
S. Veena, H. R. Sridevi, and T. C. Balachandra	
Analysis and Evaluation of Pre-processing Techniques for Fault Detection in Thermal Images of Solar Panels	673
Sujata P. Pathak and Sonali A. Patil	
Comparative Analysis of Medical Imaging Techniques Used for the Detection of Thyroid Gland with an Emphasis on Thermogram	691
G. Drakshaveni and Prasad Naik Hamsavath	
The AgroCart Android Application to Manage Agriculture System	701
N. Sreenivasa, B. A. Mohan, Roshan Fernandies, H. Sarojadevi, E. G. Satish, and Abrar Ahmed	
Dielectric Recovery and Insulating Properties of Coconut Oil and Transformer Oil	719
T. C. Balachandra and Shreeram V. Kulkarni	
Predictive Maintenance of Lead-Acid Batteries Using Machine Learning Algorithms	729
H. R. Sridevi and Shrey Bothra	
Cloud-Aided IoT for Monitoring Health Care	739
Aparna Manikonda and N. Nalini	
Energy-Efficient Dynamic Source Routing in Wireless Sensor Networks	749
Dileep Reddy Bolla, P. Ramesh Naidu, Jijesh J J, Vinay T.R, Satya Srikanth Palle, and Keshavamurthy	
Impact on Squeeze Film Lubrication on Long Cylinder and Infinite Plane Surface Subject to Magnetohydrodynamics and Couple Stress Lubrication	765
C. K. Sreekala, B. N. Hanumagowda, R. Padmavathi, J. Santhosh Kumar, and B. V. Dhananjayamurthy	
Collapse Detection Using Fusion of Sensor	775
Sushmita A. Pattar, A. C. Ramachandra, N. Rajesh, and C. R. Prashanth	

Secured Storage of Information Using Audio Steganography 791
 M. R. Sowmya, K. N. Shreenath, Saritha Shetty, Savitha Shetty,
 Salman Wajid, and Yashas Kantharaj

Run-time Control Flow Model Extraction of Java Applications 803
 Gokul Saravanan, Goutham Subramani, P. N. S. S. Akshay,
 Nithesh Kanigolla, and K. P. Jevitha

**Accelerating Real-Time Face Detection Using Cascade Classifier
 on Hybrid [CPU-GPU] HPC Infrastructure** 817
 B. N. Chandrashekhar and H. A. Sanjay

MedArch—Medical Archive and Analytical Solution 835
 Jagadevi N. Kalshetty, V. Venkata Sree Harsha,
 and Pushpesh Prashanth

Pneumonia Prediction Using Deep Learning 847
 B. G. Mamatha Bai and V. Meghana

**An Efficient Blockchain-Based Security Framework
 for PUF-Enabled IoT Devices in Smart Grid Infrastructure** 869
 M. Prasanna Kumar and N. Nalini

The Abstraction of XOR Gate Using Reversible Logic 879
 Uttkarsh Sharma, Shruti Gatade, and N. Samanvita

**VisionX—A Virtual Assistant for the Visually Impaired Using
 Deep Learning Models** 891
 Akula Bhargav Royal, Balimidi Guru Sandeep,
 Bandi Mokshith Das, A. M. Bharath Raj Nayaka, and Sujata Joshi

**Analyzing the Performance of Novel Activation Functions
 on Deep Learning Architectures** 903
 Animesh Chaturvedi, N. Apoorva, Mayank Sharan Awasthi,
 Shubhra Jyoti, D. P. Akarsha, S. Brunda, and C. S. Soumya

Hybrid Model for Stress Detection of a Person 917
 A. C. Ramachandra, N. Rajesh, K. Mohan Varma, and C. R. Prashanth

**Machine Learning-Based Social Distance Detection:
 An Approach Using OpenCV and YOLO Framework** 931
 Deepthi Shetty, H. Sarojadevi, Onkar Bharatesh Kakamari,
 Savitha Shetty, Saritha Shetty, Radhika V. Shenoy, B. N. Rashmi,
 M. S. Sneha Dechamma, and G. Tanmaya

Autism Spectrum Disorder Prediction Using Machine Learning 947
 A. C. Ramachandra, N. Rajesh, G. Sai Harshitha, and C. R. Prashanth

Early Detection of Infection in Tomato Plant and Recommend the Solution 963
A. C. Ramachandra, N. Rajesh, N. B. Megha, Apoorva Singh, and C. R. Prashanth

Design and System Level Simulation of a MEMS Differential Capacitive Accelerometer 971
S. Veena, Newton Rai, H. L. Suresh, and Veda Sandeep Nagaraj

Auto-Load Shedding and Restoration Using Microcontroller 983
G. L. Harsha, A. S. Prathibha, S. A. Rakshit Kumar, P. G. Suraj, M. J. Nagaraj, and V. Shantha

A Review on Design and Performance Evaluation of Privacy Preservation Techniques in Data Mining 993
Jagadevi N. Kalshetty and N. Nalini

Controller Area Network (CAN)-Based Automatic Fog Light and Wiper Controller Prototype for Automobiles 1003
Sowmya Madhavan, Supriya Kalmath, R. Ramya Rao, Shreya P. Patil, and M. D. Tejaswini

Multivariate Long-Term Forecasting of T1DM: A Hybrid Econometric Model-Based Approach 1013
Rekha Phadke and H. C. Nagaraj

Master and Slave-Based Test-Bed for Waste Collection and Disposal: A Dissertation 1037
Shreeram V. Kulkarni, N. Samanvita, Shruti Gatade, and Sowmya Raman

About the Editors

N. R. Shetty is Chancellor of the Central University of Karnataka, Kalaburagi, and Chairman of Review Commission for the State Private University Karnataka. He is currently serving as an advisor to the Nitte Meenakshi Institute of Technology (NMIT), Bengaluru. He is also Founder Vice-President of the International Federation of Engineering Education Societies (IFEES), Washington DC, USA. He served as Vice-Chancellor of Bangalore University for two terms and President of the ISTE, New Delhi, for three terms. He was also Member of the Executive Committee of the AICTE and Chairman of its South West Region Committee.

L. M. Patnaik obtained his Ph.D. in 1978 in the area of real-time systems and D.Sc. in 1989 in the areas of computer systems and architectures, both from the Indian Institute of Science (IISc), Bengaluru. From March 2008 to August 2011, he was Vice-Chancellor, Defence Institute of Advanced Technology, Deemed University, Pune. Currently, he is Honorary Professor with the Department of Electronic Systems Engineering, Indian Institute of Science, Bengaluru, and INSA Senior Scientist and Adjunct Professor with the National Institute of Advanced Studies, Bengaluru. During the last 50 years of his long service, his teaching, research, and development interests have been in the areas of parallel and distributed computing, computer architecture, CAD of VLSI systems, high-performance computing, mobile computing, theoretical computer science, real-time systems, soft computing and computational neuroscience including machine cognition. In these areas, he has 1286 publications in refereed international journals and refereed international conference proceedings including 30 technical reports, 43 books, and 26 chapters in books.

N. H. Prasad is currently working as Professor and Head of the Department of Master of Computer Applications at Nitte Meenakshi Institute of Technology, Bengaluru. He completed his Ph.D. at Jawaharlal Nehru University, New Delhi, India. He has more than 18 years of experience in different roles in both public and private sector enterprises, including the Ministry of Human Resource and Development, New Delhi, Government of India. He has received the prestigious “Dr. Abdul Kalam Life Time Achievement Award” and also received a “Young Faculty” award at the 2nd Academic Brilliance Awards.

Teaching Learning-Based Optimization with Learning Enthusiasm Mechanism for Optimal Control of PV Inverters in Utility Grids for Techno-Economic Goals



Nirmala John , Varaprasad Janamala , and Joseph Rodrigues 

1 Introduction

Depletion of raw materials required for conventional coal power plants, changing rain patterns, environmental concerns such as pollution and climatic changes and growth of deregulated energy markets have turned the focus in the energy sector to naturally available renewable energy sources (RESs). Being situated near to the load end sites, these sources are also referred to as distributed generation sources. In addition to being environmental friendly, these small-scale sources, when properly sized and located, can provide technical benefits such as reduced system losses, improved voltage profile, and voltage stability for the system due to their nearness to the load ends. Though DG sources were earlier thought to be more the privy of independent owners, with the distributed sources becoming more and more an integral part of the electrical system, utilities today are looking at themselves as cost-effective distributed generation providers.

Optimal deployment of DG has been a topic of continued interest among researchers. Earlier research mainly concentrated on DG sources (DGs) which supplied only active power [1, 2] or only reactive power [3]. But with high penetration of DGs, expectation of reactive power from the sources also increased. Later studies have therefore concentrated on optimal power factor (pf) determination of DG along with their positions and sizes. The methods have been analytical or heuristic

N. John (✉) · V. Janamala · J. Rodrigues
Department of Electrical and Electronics Engineering, School of Engineering and Technology,
Christ (Deemed to Be University), Bangalore, Karnataka 560 074, India
e-mail: nirmala.john@christuniversity.in

V. Janamala
e-mail: varaprasad.janamala@christuniversity.in

J. Rodrigues
e-mail: joseph.rodrigues@christuniversity.in

with single objective or multiple objective optimizations. Minimization of losses and improvement of voltage stability have been the main objectives considered to optimize the DG deployment. Analytical and heuristic approaches have been widely used in literature for finding solutions for optimal DG placement (ODGP) problem. An analytical approach for DG deployment with optimal power factor (pf) for loss minimization has been considered in [4, 5]. A hybrid approach combining an analytical approach to determine sizes and PSO algorithm to determine locations of DG sources has been employed in [6]. Due to computational ease and the availability of newer and efficient algorithms, heuristic algorithms have become popular for solving ODGP problems. Hybrid grey wolf optimizer algorithm was used to optimize DG allocation considering different types of DG in [7]. In [8], the influence of various operational pfs for the DG, after DG placement, was addressed for loss minimization and voltage stability enhancement with evolutionary algorithm. Differential evolution-based algorithm was implemented in [9] to optimize DG locations, sizes, and pf to minimize distribution losses. DG allocation with a fixed pf of 0.9 has been solved using dragonfly algorithm in [10]. A DG placement index based on loss sensitivity factor, voltage stability index, and reliability-based factor has been employed for optimal DG placement in [11]. With the new index, the author has worked out DG placement at both unity pf and fixed pf of 0.8.

While most authors have focused on the technical benefits of optimal placement, few authors have also considered economic and environmental considerations while determining optimal DG allocation. Both technical and economic aspects have been considered for simultaneous deployment of capacitor banks and DG using a hybrid method of imperial competitive algorithm (ICA) and genetic algorithm (GA) [12]. Operational cost incurred by DisCo has been considered along with technical objectives of voltage stability, voltage deviation index, and loss minimization for DG allocation [13, 14]. Both papers have considered fixed DG power factors while determining optimal DG allocation.

Ownership of DG can rest with a third party or with the DisCo. The DisCo's major goals are cost reduction and technical improvement of network, whereas the DG owners' purpose is to maximize revenue by selling as much electricity as possible to the distribution network. Distribution companies may install their own DG to fulfill their electricity demand partially. This reduces the power purchased by them from the grid, thereby reducing their operational cost especially during peak hours. With fuel cost of renewable sources being zero or negligible, the energy cost incurred by the DisCo is dependent on the power drawn from the grid.

Teaching learning-based optimization (TLBO) algorithm was proposed by Rao et al. in 2011 [15]. The effectiveness of swarm-based and evolutionary algorithms depends on adjustment of parameters which are unique to each algorithm. TLBO is notable with the absence of such parameters and uses only common control parameters of population size and no. of generations. The algorithm also requires less computational effort and has become widely popular among researchers. The evidence of TLBO algorithm for solving the optimal DG placement problem has been observed in [16, 17]. Since the introduction of TLBO, many variants have been proposed

to improve the exploration and exploitation capabilities of the basic TLBO algorithm. The basic TLBO algorithm considers the learning enthusiasm of all learners to be same. Learning enthusiasm-based TLBO (LebTLBO) was proposed by the authors in [18]. LebTLBO considers a learning enthusiasm value for each learner. This improves the search efficiency of the algorithm.

This paper considers the optimal deployment and operational power factor of DGs in a distribution system embedded with DisCo-owned DGs. With PV systems constituting 50% of the renewable energy sources [19], the DGs considered are PV systems with inverter control capability. A multi-objective function has been formulated with the aim of reducing distribution losses, improving voltage stability, and reducing energy costs. The proposed solution is based on LebTLBO algorithm and backward/forward sweep load flow method [20]. The study assumes availability of storage backed PV sources for continuous supply.

2 Problem Formulation

2.1 Objective Functions

2.1.1 Distribution Losses

If the distribution losses are minimized, the active power delivery to the load increases. The distribution losses minimization can be formulated as in (1), where nbr is number of branches and I_k and R_k are current and resistance of k th branch.

$$f_1 = \sum_{k=1}^{\text{nbr}} I_k^2 R_k \quad (1)$$

2.1.2 Voltage Stability Index

Improvement of voltage stability can be analyzed with voltage stability indices. The voltage stability index (VSI), introduced in [21], can be determined at each node. The least value is considered as the index of the system. The maximum value of the voltage stability index is considered as 1. With a line connecting buses i and j , the voltage stability index at the j th bus can be determined as shown in Eq. (2). V_i is the voltage of i th bus and R_{line} and X_{line} the resistance and reactance of the line.

$$f_2 = |V_i^4| - 4 \times (P_j \times X_{\text{line}} - Q_j \times R_{\text{line}})^2 - 4 \times \frac{P_j \times R_{\text{line}} + Q_j \times X_{\text{line}}}{V_i^2} \quad (2)$$

2.1.3 Energy Cost

Real power demand (P) from the grid is given by (3). As the distribution companies are the DG owners, it is important that the company derives the maximum operational benefit by minimizing the amount of purchased energy from grid. P_i is the active load at each bus I , and P_{DGn} is the real power delivered by the n th DG. N_{DG} is no. of buses with DG.

$$P = \sum_{i=1}^N P_i - \sum_{n=1}^{N_{DG}} P_{DGn} + P_{\text{loss}} \quad (3)$$

With DG sources considered as renewable energy sources, the fuel costs are negligible. The energy cost incurred by the utility (DG operator) consists of the cost to be paid for energy bought from grid. The energy cost can be calculated as in (4). C_p is the cost of energy, and P the number of units bought from the grid.

$$f_3 = P \times C_p \quad (4)$$

2.2 Constraints

Constraints in this optimization problem are given by Eqs. (5) and (6).

2.2.1 Voltage Constraint

The voltage at every bus must be kept within standard limits. Equation (5) represents the voltage constraint. The limits considered are -6% to $+6\%$, i.e., 0.94 pu and 1.06 pu, respectively.

$$V_{i,\min} < V_i < V_{i,\max} \quad (5)$$

2.2.2 DG Power Factor Constraint

$$0.8 \leq \text{pf}_{DG} \leq 1 \quad (6)$$

Equation (6) represents the DG power factor constraint.

2.3 Multi Objective Function

The objective is to maximize technical benefits of reduced distribution losses and improved voltage stability and minimize energy cost. The multi-objective function for optimal operation of the PV inverters is given by (7).

$$\text{OF} = \min\{k_1 f_1 + k_2(1 - f_2) + k_3 f_3\} \quad (7)$$

3 Learning Enthusiasm-Based TLBO (LebTLBO)

The basic TLBO algorithm is based on the influence of teacher and peers on the learning of a student. The algorithm involves two phases—the teacher phase and the student phase. The teacher phase accounts for the influence of teacher, and the learner phase accounts for the influence of peers on a student's result. However, the learning enthusiasm (LE) of each student is different in the real-world scenario. Therefore, in LebTLBO, the learning enthusiasm mechanism has been introduced into the teacher phase and the learner phase. LebTLBO, in addition, has a poor student tutoring phase.

3.1 Basic TLBO

In basic, TLBO learners represent the population and subjects represent the design variables. Assume $x_i = (x_i^1, x_i^2, x_i^3, \dots, x_i^d)$ be the i th learner vector, where $1, 2, \dots, d$ are the design variables (subjects). After evaluating the fitness function, the learner with best fitness is chosen as the teacher. In a class of NL learners, the mean position can be taken as (8)

$$x_{\text{mean}}^d = \sum_{i=1}^{\text{NL}} x_i^d \quad (8)$$

Each learner position is modified in the teacher phase as Eq. (9). T_F is the teacher factor which is either 1 or 2.

$$x_{i,\text{new}} = x_{i,\text{old}} + x_{\text{teacher}} - T_F x_{\text{mean}} \quad (9)$$

The modified vectors from teacher phase become the input to the learner phase. Learner phase involves the interaction of learner i with learner j , and the i th learner vector is modified as in Eq. (10).

$$\begin{aligned}
x_{i,\text{new}} &= x_{i,\text{old}} + \text{rand}(x_{i,\text{old}} - x_{j,\text{old}}) && \text{if } f(x_{i,\text{old}}) \text{ better than } f(x_{j,\text{old}}) \\
x_{i,\text{new}} &= x_{i,\text{old}} + \text{rand}(x_{j,\text{old}} - x_{i,\text{old}}) && \text{if } f(x_{j,\text{old}}) \text{ better than } f(x_{i,\text{old}})
\end{aligned} \tag{10}$$

3.2 *LebTLBO*

3.2.1 Defining the Learning Enthusiasm

It is assumed that learners with high grades have good LE, while learners with low grades have poor LE. Therefore, the learners are sorted in the order of best to worst based on grades Eq. (11). For a problem with minimization of fitness value involving NL learners, let

$$f(x_1) \leq f(x_2) \leq f(x_3) \leq \dots f(x_{NL}) \tag{11}$$

The LE value for the k^{th} learner is then obtained as Eq. (12).

$$LE_k = LE_{\min} + (LE_{\max} - LE_{\min}) \frac{NL - K}{NL} \tag{12}$$

$k = 1, 2, \dots, NL$; $LE_{\max} = 1$ and $LE_{\min} = [0.1, 0.5]$.

3.2.2 Diversity Enhanced Teaching Strategy

The teacher is considered the best performer of the class. For every learner x_k , a random number $r_k \in [0, 1]$ is generated. If ($r_k < LE_k$), then it is considered that the learner x_k will learn from the teacher; otherwise, the learner will disregard the knowledge from teacher. A diversity enhanced teaching strategy is then used for the learner who is considered to learn from the teacher. The updated vector is given by (13).

$$x_{k,\text{new}}^d = \begin{cases} x_{k,\text{old}}^d + \text{rand}_2(x_{\text{teacher}}^d - T_F \times x_{\text{mean}}^d) & \text{if } \text{rand}_1 < 0.5 \\ x_{r_1}^d + F(x_{r_2}^d - x_{r_3}^d) & \text{if } \text{rand}_1 > 0.5 \end{cases} \tag{13}$$

where r_2 and r_3 are integers selected randomly from $(1, 2, \dots, NL)$, $d \in \{0, 1 \dots D\}$, rand_1 and rand_2 are two uniformly distributed random numbers in the range $[0, 1]$, and F is a scale factor in $[0, 1]$. The updation of the current vector employs a mix of basic TLBO and the differential evolution mutation operator unlike basic TLBO which uses the same differential vector $x_{\text{teacher}} - T_F \times x_{\text{mean}}$ to steer every learner

to the level of teacher. If the vector $x_{k,\text{new}}$ is better than $x_{k,\text{old}}$, it is accepted, else the value of $x_{k,\text{old}}$ is retained.

3.2.3 LE-Based Learner Phase

The learner phase is similar to basic TLBO but with the introduction of learning enthusiasm. The learning enthusiasm values are defined using (12) after sorting the learners in the order of grades. As in the teaching phase, a random value r_k is generated, and if r_k is lesser than LE_k , then it is considered that the k th learner can learn from other learners. The updated vector for x_k after interaction with j th learner is given by Eq. (14).

$$\begin{aligned} x_{k,\text{new}} &= x_{k,\text{old}} + \text{rand}(x_{k,\text{old}} - x_{j,\text{old}}) \quad \text{if } f(x_{k,\text{old}}) \text{ better than } f(x_{j,\text{old}}) \\ x_{k,\text{new}} &= x_{k,\text{old}} + \text{rand}(x_{j,\text{old}} - x_{k,\text{old}}) \quad \text{if } f(x_{j,\text{old}}) \text{ better than } f(x_{k,\text{old}}) \end{aligned} \quad (14)$$

where rand is a uniformly distributed random vector within the interval $[0, 1]$. If the vector $x_{k,\text{new}}$ is better than $x_{k,\text{old}}$ it is accepted, else the value of $x_{k,\text{old}}$ is retained.

3.2.4 Poor Student Tutoring Phase

Students with lower grades have very few opportunities to improve their grades during the teaching and learning phases than students with higher grades. The third phase of poor student tutoring helps to resolve this. According to their grades, students are ranked from best to worst. The learners who are in the bottom ten percent of the ranking are considered as poor learners. For each poor learner, a random learner x_{top} at the top fifty percent is selected. The updated vector can be represented as (15).

$$x_{k,\text{new}} = x_{k,\text{old}} + \text{rand}(x_{\text{top}} - x_{k,\text{old}}) \quad (15)$$

If the vector $x_{k,\text{new}}$ is better than $x_{k,\text{old}}$, it is accepted, else the value of $x_{k,\text{old}}$ is retained.

3.2.5 Application of LebTLBO to the Problem

The application of the LebTLBO algorithm to the problem is depicted using a flowchart as shown in Fig. 1.

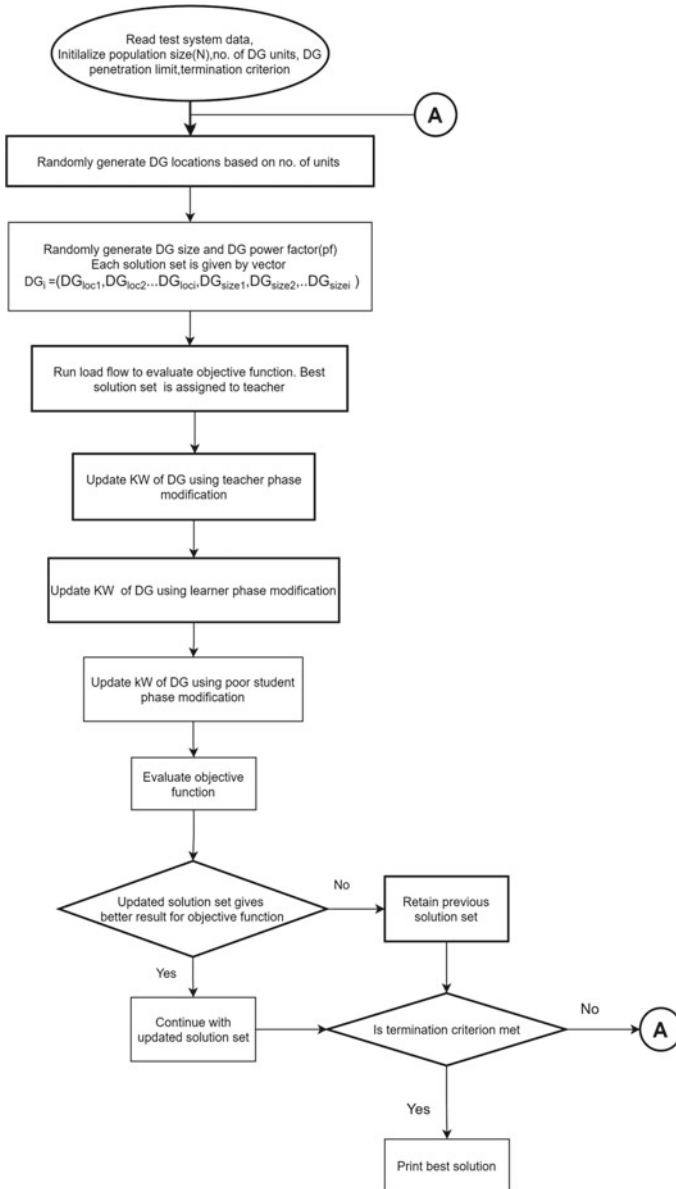


Fig. 1 LebTLBO algorithm implementation for OPDG application

4 Results and Discussions

IEEE 33 bus standard test system with connected load of 3715 kW + 2300 kVAR [22] has been used for simulations. The PV systems are assumed to have continuous ratings. Optimal sizes and locations are determined considering unity power factor for the PV inverters. With the same PV deployment, the optimal inverter power factor is determined with the single objective function of minimizing losses (case 1) and with the multi-objective function of minimizing distribution losses, maximizing voltage stability, and minimizing energy cost (case 2). The power factor variation for the PV inverters is assumed from 0.8 leading to unity power factor to find the optimal operational pf of PV systems. Table 1 summarizes the base case results.

Optimal PV deployment to minimize distribution losses

The objective function is formulated as in Eq. (1) to minimize distribution losses. Table 2 summarizes the findings. As compared to the base case, the distribution losses are reduced by 67.15% when compared to base case losses. Comparative analysis of results with existing solutions shows the effectiveness of the proposed algorithm. The proposed method is able to give lower losses as compared to existing methods. The lower losses are attained at a lower penetration level. To highlight the competitiveness of LebTLBO, the obtained performance characteristics are compared with that of other heuristic algorithms like PSO, CSA, and basic TLBO in solving the stated OPDG problem. This is depicted in Fig. 2.

Optimal PV inverter power factor to minimize distribution losses (Case 1)

The DG deployment is considered same as in Table 2. The variations in load profile and generation have not been considered here. The results obtained for the 33 bus test system, listed in Table 3, show that distribution losses reduce and the voltage

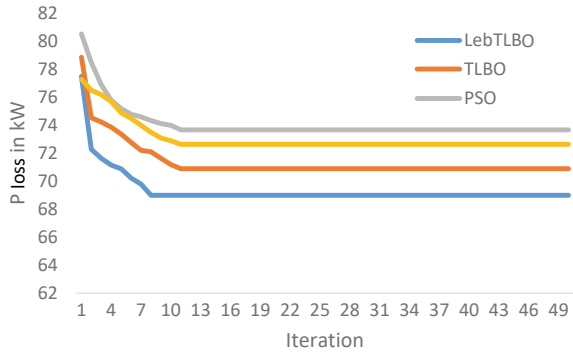
Table 1 Base case test system results

Test system	Losses (kW)	VSI	Minimum bus voltage (pu)/bus no.
IEEE 33 bus system	210.9	0.6486	0.9038 (18)

Table 2 Comparative analysis of simulation results

Method	kVA and locations of PV systems	Losses (kW)	Total (kVA)
Proposed method	460.68(8), 538.75(13), 868.65(24), 1040(30)	69	2908
ACO-ABC [23]	754.7(14), 1099.9(24), 1071.4(30)	71.4	2926
HGWO[7]	802(13), 1090(24), 1054(30)	72.784	2946
PSO[5]	770(14), 1090(24), 1070(30)	72.79	2930
EA-OPF [24]	802(13), 1091(24), 1054(30)	72.79	2947

Fig. 2 Convergence characteristics



stability of the system improves when the PV inverters operate at non-unity power factors. The distribution losses have been reduced to 15.2 kW from the base case loss of 210.07 kW. The comparative analysis with existing results in literature as given in Table 3 shows that the PV deployment results given by the proposed method are able to give lower losses. As the power factor constraint requires the inverter power factor to be above 0.8, the compared results have been chosen where the power factor of DG is maintained above 0.8. The proposed method has achieved lower losses as compared to other methods, and as the results show, this has been achieved with a lower DG penetration level. The simulation findings demonstrate the efficacy of the proposed method.

Table 3 Optimal PV inverter power factor (case 1)

Test system	Method	kVA ratings, pf, and locations of DG sources	Losses (kW)	Total (kVA)
33 bus	Proposed method [LebTLBO]	(460.68,0.89, 8), (538.75,0.88,13), (868.65,0.9,24), (1040,0.8,30)	15.2	2908
	LSFSA [25]	(1382.9,0.866,6), (551.73,0.866,18), (1062.9,0.866,30)	26.72	2994
	IA [26]	(1098,6,0.82), (768,0.82,14), (1098,0.82,30)	22.26	2964
	HSA-PABC [27]	(1014,0.85,12), (960,0.85,25), (1363,0.85,30)	15.91	2880

Table 4 Optimal PV power factor (case 2)

Test system	kVA ratings, pf, and locations of DG sources	Losses (kW)	Minimum voltage in the system (pu)	VSI
33 bus	(460.68,0.9, 8), (538.75,0.9,13), (868.65,0.91,24), (1040,0.8,30)	15.5	0.9791 (18)	0.9188

Table 5 Comparative analysis of case1 and case 2 for PV inverter control

Case	Losses (kW)	VSI	Minimum bus voltage (pu)	Power from grid (kW)	Energy cost (Rs/h)
Case 1	15.2	0.9193	0.9792	1232.2	6162.5
Case 2	15.5	0.9188	0.9791	1208.5	6042.5

Optimal PV inverter control to minimize distribution losses, maximize voltage stability, and minimize energy costs (Case 2)

The objective function is in Eq. (7). The rate at which power is bought from grid is 5/kWh in Indian rupees [28]. The variations in load profile and generation have not been considered here. The results obtained are presented in Table 4, and a comparative analysis with case 1 results is presented in Table 5. The losses are marginally higher by 0.3 kW in case 2 as against case 1. But the higher power factor improves the active power delivery from the DG by 24 kW. This reduces the power to be purchased from the substation. Considering the marginally higher losses, the effective reduction in purchased power per hour from the substation with connected load demand is 23.7 kW. This results in a considerable saving of 10.3 lakhs per year to the distribution company.

5 Conclusion

In this study, LebTLBO algorithm is used to develop a new approach for determining the best allocation and operating power factors of DisCo-owned DG sources in a radial distribution network. With the proposed method, the optimal deployment of PV systems and optimal PV inverter power factor have been determined with the objectives of minimizing distribution losses, improving voltage stability, and minimizing energy cost. Comparative analysis with existing results has been presented to show the effectiveness of the LebTLBO algorithm. The multi-objective function improves the voltage stability and reduces the system losses. The improved utilization of the PV system capacities reduces the incurred energy cost. The total operational cost for the DisCos therefore reduces. In this work, variations in network load profiles

and DG powers are ignored, which can result in changes in voltage stability and even economic issues. This can be considered an extension.


References

1. Prakash DB, Lakshminarayana C (2016) Multiple DG placements in distribution system for power loss reduction using PSO algorithm. *Procedia Technol* 25:785–792
2. Shaaban MF, Atwa YM, El-Saadany EF (2013) DG allocation for benefit maximization in distribution networks. *IEEE Trans Power Syst* 28:639–649
3. Gnanasekaran N, Chandramohan S, Sathish Kumar P, Mohamed Imran A (2016) Optimal placement of capacitors in radial distribution system using shark smell optimization algorithm. *Ain Shams Eng J* 7(2):907–916
4. Hung DQ, Mithulananthan N (2013) Multiple distributed generator placement in primary distribution networks for loss reduction. *IEEE Trans Industr Electron* 60(4):1700–1708
5. Sa'ed JA, Amer MA, Bodair A, Baransi A, Favuzza S, Zizzo GA (2019) Simplified analytical approach for optimal planning of distributed generation in electrical distribution networks. *Appl Sci* 9(24):5446
6. Kansal S, Kumar V, Tyagi B (2016) Hybrid approach for optimal placement of multiple DGs of multiple types in distribution networks. *Int J Electr Power Energy Syst* 75:226–235
7. Sanjay R, Jayabarathi T, Raghunathan T, Ramesh V, Mithulananthan N (2017) Optimal allocation of distributed generation using hybrid grey wolf optimizer. *IEEE Access* 5:14807–14818
8. Moravej Z, Ardejani PE, Imani A (2018) Optimum placement and sizing of DG units based on improving voltage stability using multi-objective evolutionary algorithm. *J Renew Sustain Energy* 10:055304
9. Huy PD, Ramachandaramurthy VK, Yong JY, Tan KM, Ekanayake JB (2020) Optimal placement, sizing and power factor of distributed generation: a comprehensive study spanning from the planning stage to the operation stage. *Energy* 195:117011
10. Suresh MCV, Belwin EJ (2018) Optimal DG placement for benefit maximization in distribution networks by using Dragonfly algorithm. *Renew Wind Water Solar* 5(4):1–8
11. Memarzadeh G, Keynia F (2020) A new index-based method for optimal DG placement in distribution networks. *Eng Rep* 2:e12243
12. Moradi MH, Zeinalzadeh A, Mohammadi Y, Abedini M (2014) An efficient hybrid method for solving the optimal sitting and sizing problem of DG and shunt capacitor banks simultaneously based on imperialist competitive algorithm and genetic algorithm. *Electr Power Energy Syst* 54:101–111
13. Samala RK, Mercy Rosalina K (2021) Optimal allocation of multiple photo-voltaic and/or wind-turbine based distributed generations in radial distribution system using hybrid technique with fuzzy logic controller. *J Electr Eng Technol* 16:101–113
14. Rama Prabha D, Jayabarathi T (2016) Optimal placement and sizing of multiple distributed generating units in distribution networks by invasive weed optimization algorithm. *Ain Shams Eng J* 7(2):683–694
15. Rao RV, Savsani VJ, Vakharia P (2011) Teaching-learning based optimization: a novel method for constrained mechanical design optimization problems. *Comput Aid Des* 43(3):303–315
16. Mohanty B, Tripathy S (2016) A teaching learning based optimization technique for optimal location and size of DG in distribution network. *J Electr Syst Inf Technol* 3:33–44
17. Bhattacharyya B, Babu R (2016) Teaching learning based optimization algorithm for reactive power planning. *Int J Electr Power Energy Syst* 81:248–253
18. Chen X, Xu B, Yu K, Wenli Du (2018) Teaching-learning-based optimization with learning enthusiasm mechanism and its applications in chemical engineering. *J Appl Math* 19:1806947
19. Jäger-Waldau A (2017) PV status report. Publications office of the European Union, Luxembourg

20. Haque MH (1996) Efficient load flow method for distribution systems with radial or mesh configuration. *IEE Proc Gener Transm Distrib* 143(1):33–38
21. Chakravorty M, Das D (2001) Voltage stability analysis of radial distribution networks. *Int J Electr Power Energy Syst* 23(2):129–135
22. Kashem MA, Ganapathy V, Jasmon GB, Buhari MI (2000) A novel method for loss minimization in distribution networks. In: *Proceedings of the international conference on electric utility deregulation and restructuring and power technologies*, City University, London, pp 251–256
23. Kefayat M, Lashkar Ara A, Nabavi Niaki SA (2015) A hybrid of ant colony optimization and artificial bee colony algorithm for probabilistic optimal placement and sizing of distributed energy resources. *Energy Convers Manage* 92:149–161
24. Mahmoud K, Yorino N, Ahmed A (2016) Optimal distributed generation allocation in distribution systems for loss minimization. *IEEE Trans Power Syst* 31(2):960–969
25. Injeti SK, Prema Kumar N (2013) A novel approach to identify optimal access point and capacity of multiple DGs in a small, medium and large scale radial distribution systems. *Int J Electr Power Energy Syst* 45(1):142–151
26. Hung DQ, Mithulananthan N (2013) Multiple distributed generator placement in primary distribution networks for loss reduction. *IEEE Trans Ind Electron* 60(4):1700–1708
27. Muthukumar K, Jayalalitha S (2016) Optimal placement and sizing of distributed generators and shunt capacitors for power loss minimization in radial distribution networks using hybrid heuristic search optimization technique. *Int J Elect Power Energy Syst* 78:299–319
28. Kansal S, Tyagi B, Kumar V (2017) Cost–benefit analysis for optimal distributed generation placement in distribution systems. *Int J Ambient Energy* 38(1):45–54

Machine Learning Framework for Prediction of Parkinson's Disease in Cloud Environment



K. Aditya Shastry , V. Sushma, Naman Bansal, Ujjwal Saxena,
Shrey Srivastava, and Suvang Samal

1 Introduction

The severity of “Parkinson’s disease” worsens over time, making it one of the most complicated conditions that affects a person’s motor functions. This disease affects around 1% of the population over sixty. The prevalence is approximately 250 per 10,000 persons. Average onset age is between 55 and 65 years [1]. The problem of Parkinson’s Disease prediction using ML methods is novel. The early diagnosis of the phases of Parkinson’s disease can be very beneficial for treating the illness [2]. The use of computing resources in healthcare departments is increasing all the time, and it is becoming the norm to electronically record patient data that was formerly recorded on paper-based forms. As a result, a substantial number of electronic health records are now more accessible. ML and data mining techniques can be used to improve the quality and productivity of medical and healthcare facilities, as well as predict the likelihood of Parkinson’s disease [3].

Over the last two decades, the rate of innovation in the field of machine learning has risen dramatically. From machine learning models that can classify every object in a photograph to disease detection, progress can be seen in a range of sectors [4]. Machine learning has led to a number of successes, including the detection of diseases in patients, the development of AI chatbots, and the enhancement of speech recognition in natural language processing [5]. The objective of this effort was to build a system for forecasting Parkinson’s disease that takes voice data as input, transmits it to the cloud for speech analysis, and then uses a machine learning model to come up with a prediction.

K. A. Shastry (✉) · V. Sushma · N. Bansal · U. Saxena · S. Srivastava · S. Samal
Nitte Meenakshi Institute of Technology, Bengaluru 560064, India
e-mail: adityashastry.k@nmit.ac.in

V. Sushma
e-mail: sushma.v@nmit.ac.in

2 Related Work

Various research on the diagnosis of Parkinson's disease have been done in recent years, employing models such as neural networks, decision trees, and regression, thanks to advancements in the field of machine learning and natural language processing. This section discusses certain relevant research works in the area of Parkinson detection using ML techniques.

Pramanik and Sarker [6] conducted one such research on the detection of "Parkinson's" utilizing vocal information from patients. The information employed in this research was provided by the "Department of Neurology in Cerrahpasa, Faculty of Medicine, Istanbul University". The data set includes information from "188" people with Parkinson's disease (in which 81 are women and are 107 men). The patients in this data set range in age from 33 to 87, with an average age of 65.1. The "data set" also includes information of "64" healthful people ("41" female and "23" male), with a median age of 61.1. In [7], the researchers intended to discriminate Parkinson Disease subjects from the people who were not afflicted by the disease. The study solicited the help of 40 Parkinson's disease sufferers and 40 healthy people. This investigation's methodology includes a brief questionnaire and three recordings from each participant. A total of 44 acoustic features were examined in each recording. i.e., 44-dimensional vector per voice recording. These extracted traits are classified into many groups based on their ability to predict whether or not a person will be impacted by Parkinson's disease.

The authors of [8] built a cloud-based system for calculating, storing, and monitoring voice and tremor samples taken by cell phones to identify Parkinson's disease. They discovered that k -nearest neighbors (k-NN) outperformed support vector machine (SVM) and naive Bayes in terms of accuracy (NB). To identify Parkinson's patient samples from healthy people, [9] employed a classification system based on convolutional neural networks (CNN), artificial neural networks (ANN), and hidden Markov models (HMM). The authors found that the ANN-based Parkinson detection system outperforms the HMM and CNN-based Parkinson detection systems.

3 Dataset

The voice dataset was collected from the UCI ML repository [11] and used in our model. The given voice data was then processed using Parselmouth, a speech analysis package. It splits the voice into numerous parameters, and we use our model to forecast the outcome based on these parameters.

Table 1 explains the parameters of the voice dataset [12].

Table 1 Parameters of the voice dataset for parkinson detection

Feature	Description
Pitch	It is a perceptual feature of sound that determines whether a sound is lower or higher
Local jitter	It signifies the mean “absolute difference” among two intervals, divided by the mean interval. It also has a threshold limit of 1.04c/o for diagnosing diseases
Local absolute jitter	It is the average “absolute difference” among two successive intervals
ppq5 jitter	The “five-point Period Perturbation Quotient” (ppq5) is the mean “absolute difference” among an interval and the mean of its four nearest neighbors, divided by the mean interval. Because this figure was based on jitter measurements influenced by noise, MDVP labels this parameter “PPQ” and uses “0.840”c/o as a pathology “threshold”
Jitter rap	It represents the disturbance’s average, i.e., the mean “absolute difference” among one period and the period’s mean, as well as the period’s average with its two neighbors
Local shimmer	It is the mean “absolute difference” among the “amplitudes” of two successive intervals
Local DB shimmer	This is the difference in the amplitude and frequency of consecutive periods’ average absolute base-10 logarithm, magnified by 20. “MDVP” refers to this parameter as “ShdB” and establishes a diagnostic threshold of 0.350 dB
AQ3 shimmer	The “Amplitude Perturbation Quotient” is a ternary measure. It is the mean “absolute difference” among an interval’s amplitude and the mean of its neighbors’ amplitudes, divided by the mean amplitude
AQ5 shimmer	The “Amplitude Perturbation Quotient” is a “five-point” metric. It denotes the mean “absolute difference” among an interval’s amplitude and the mean of its four nearest neighbors’ amplitudes, divided by the mean amplitude
AQ11 shimmer	The “Amplitude Perturbation Quotient” is a metric of 11 points. It is the mean “absolute difference” among the amplitude of an interval and the mean amplitudes of its 10 nearest neighbors, divided by the mean amplitude. “MDVP” refers to this statistic as “APQ”, and it has a diagnostic limit of “3.070%”
Target attribute	A score of 0 suggests that the individual does not have Parkinson’s disease, whereas a value of 1 shows that the individual has

4 Proposed Work

This section describes the proposed work. Figure 1 depicts the application’s framework.

The disease-related data is stored in the cloud. The Parkinson’s input parameters are used to determine whether the individual has Parkinson’s disease. The prediction model is then deployed on the cloud, enabling doctors all over the globe to access the findings.

The data flow diagram for the application is shown in Fig. 2.

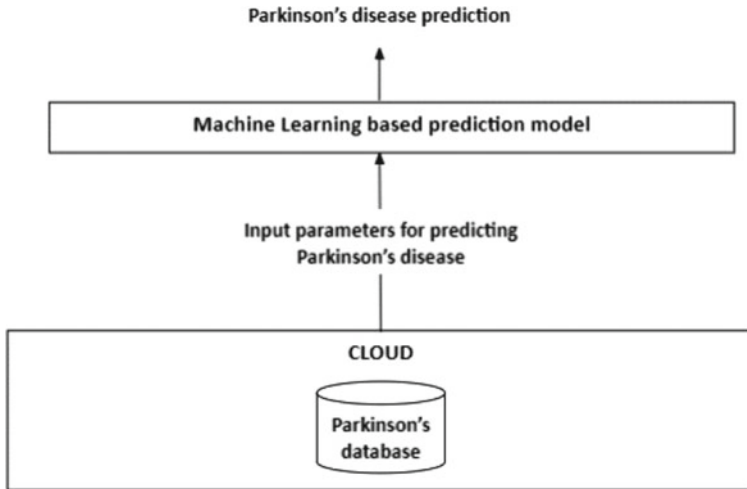


Fig. 1 Framework of the proposed work

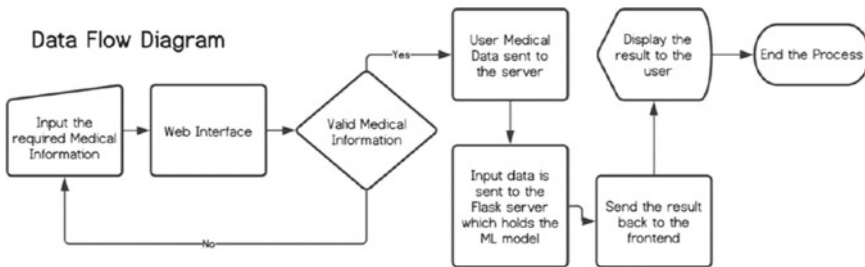


Fig. 2 Flowchart of the proposed application

We have voice recordings of users who have the condition and those who do not. We use that to train the model. The user who wants to test for Parkinson’s disease uploads a voice recording to a server that holds the speech processing library and machine learning model. The result is then retrieved and communicated to the user, who is informed whether or not he is infected with the disease.

In this work, the decision tree classifier and the random forest classifier were compared. The algorithms are explained in Sects. 4.1 and 4.2.

4.1 Decision Tree Classifier

“Decision Tree” is a “supervised learning” technique that may be utilized to resolve both classification and “regression” problems, nevertheless it is generally utilized to

resolve classification issues. It is a tree-structured classifier, with core nodes representing data set features, branches representing choice rules, and every “leaf node” representing the outcome. The decision and the leaf nodes signify the two connections of a decision tree. Decision nodes make decisions and have many branches, whereas leaf nodes represent the output of these decisions and have no longer branches. The decisions or tests are made based on the data gathering features [13].

Algorithm 1 shows the decision tree classifier used for Parkinson’s disease prediction.

Algorithm 1: Decision Tree Classifier

```

INPUT: S, where S = set of classified instances
OUTPUT: Decision Tree
initialization;
while not all partitions processed do
    maxGain = 0;
    splitA = null;
    e = Entropy(Attributes)
    for all attributes a in S do
        gain = InformationGain(a, e)
        if gain > maxGain then
            maxGain = gain
            splitA = a
        end
    end
    Partition(S, splitA)
end

```

4.2 Random Forest Classifier

“Random forests” also called “random decision forests” are a category of “ensemble learning” techniques for “classification, regression,” and additional jobs that operate by constructing a collection of “decision trees” at the time of training. The “random forest’s” outcome for categorization tasks is the category of the given trees. We return the average estimate of the specific trees for regression tasks. Decision trees tend to overfit their training set, which is corrected by random forests. “Random forests” perform better than “decision trees” in majority of the instances, nevertheless they are less precise than “gradient” enhanced trees. The performance of “random forest” can be adversely impacted by the quality of the data [14].

Algorithm 2 shows the random forest classifier used for Parkinson disease prediction.

Algorithm 2: Random Forest Classifier

INPUT: S where $S =$ training set $(x_1, y_1), \dots, (x_n, y_n)$, F where $F =$ features, B where $B =$ number of trees in forest

FUNCTION (*Random Forest*(S, F))

$H = \phi$

for i from 1 to B do

- | $S^{(i)} =$ A bootstrap sample from S
- | $h_i =$ RandomizedTreeLearn($S^{(i)}; F$)
- | $H = H \cup \{h_i\}$

end

return H

FUNCTION (*RandomizedTreeLearn*(S, F))

for each node: do

- | $f =$ very small subset of F
- | Split on best feature in f

end

return the learned tree

Figure 3 demonstrates the working of the “random forest” classifier for detecting Parkinson.

As shown in Fig. 3, the random forest works by sampling the recordings and bagging the properties of the Parkinson voice data. Then, for each of the samples and bagged features, unique decision trees are generated. The outputs from each of the trees are then concatenated, and the final class is predicted using majority voting.

For regression problems, mean regression is utilised, whereas for classification tasks, majority voting is utilised. Because this was a classification task, the final class was determined by majority voting.

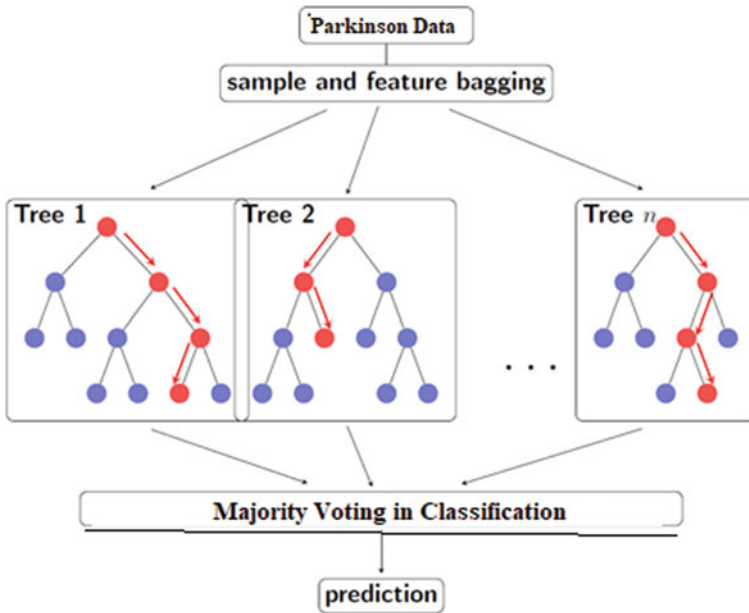


Fig. 3 Random Forest working

The proposed system involved running two different algorithms on a data set, namely:

- Decision tree classifier
- Random forest

We experimented with different attributes and identified 14 attributes that would constitute the ML model. We then used the Parselmouth library to extract those attributes from the input audio file and transmit them to the backend API which is linked to the Heroku-hosted ML model. This gives us our result which we display on our website built in React.js. We used the axios library to make a post request and submit our form data that includes the voice input of the user. We then wait for the response to be received from the api, after which it is stored in a data variable using state management in react, that is, further displayed to the user.

5 Experimental Setup and Results

“Python 3.7”, “NumPy 1.19.5”, “Pandas 1.2.4”, Pycaret 2.3.1, and sklearn were applied in this research. The hardware used for this work were 4 GB of “DDR4 RAM”, 512 GB of “Solid-State Drive (SSD)”, Intel Xeon Processor, NVIDIA Tesla K80/any CUDA compatible GPU with a minimum of 3.0 compute capability.

We used over 100 recordings of Parkinson’s patients and non-patients to train our model. Parkinson’s To achieve maximum accuracy, the training is based on a range of characteristics included in the audio data. Parselmouth is an audio processing library that we used. Praat is a voice processing software. Parselmouth not only gain access to “Praat’s C/C++ code” but also provides faster access to the system’s information. It also has a user interface that appears like any other Python library. The extracted values from Parselmouth were then converted to a csv file for analysis in Jupyter Notebook or Google Colab.

For analysis, we request that the user provide a file in.wav format. We use our trained ML model to process it and offer the outcome. The model processes the user’s voice input and then delivers the result back to the API, which is housed on a cloud server (Heroku), where it is finally shown to the user. The data is sent to the cloud server using a backend API implemented in Flask. We chose Heroku as the cloud server because of the several advantages it offers developers, including a large range of services, affordable price packages, and an optional free service for less production-intensive applications. It enables us to link our app to a ready-to-use backend API and, as a result, handle communication between the frontend and backend.

Figure 4 shows the screenshot of the screen displayed to the user when he first enters the website.

The website commences the prediction process using the ML model after the user uploads a voice file, and the user is presented the analysis screen in Fig. 5.

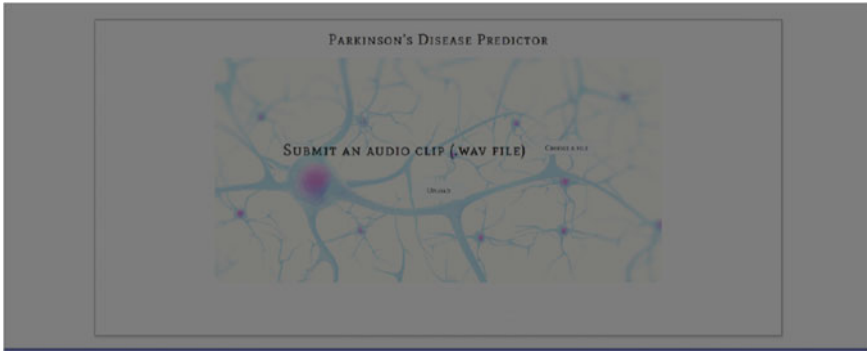


Fig. 4 Application homepage



Fig. 5 Loading while the voice data is analyzed in our application

Figure 5 depicts the results produced from the voice input provided by the user after successful prediction and analysis. The result is extracted from the API hosted on a cloud platform that manages communication between the frontend and backend.

Figure 6 illustrates the outcome of the application for the input voice file.

A decision tree classifier was the first model we implemented. With that model, we were able to achieve a 62c/o accuracy. After that, we implemented a random forest classifier-based model. With 30 trees, we were able to achieve a 90c/o accuracy rate.

The number of trees in a “random forest” classifier indicates the total number of trees in the forest. Each tree receives the same input parameters and votes yes or no separately (in binary classification). The final result is then computed using the average of the votes. Because a large number of trees can cause bias in the findings, this value needs to be fine-tuned. It is evident that we got a good boost in accuracy by switching our model from decision tree classifier to a random forest classifier. This is because a “random forest” comprises of several individual trees, and every tree is built on a random instance of training information. In several cases, this yields substantially higher accuracy than a single decision tree.

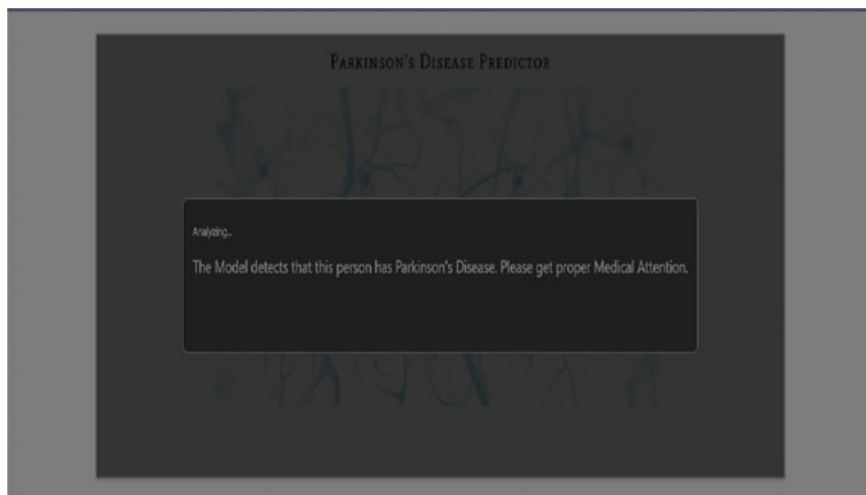


Fig. 6 Output of the application

The graph of the number of trees in a random forest versus the accuracies obtained for each number of trees is shown in Fig. 7.

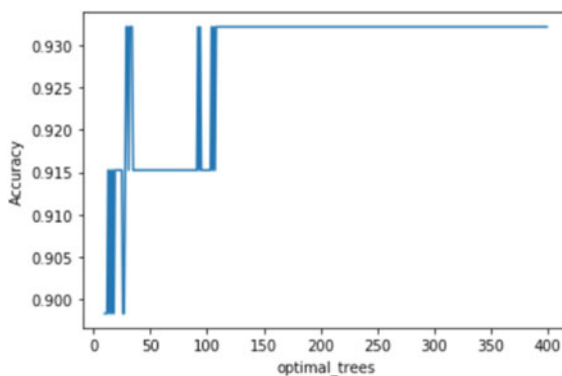
The number of trees were varied from 1 to 400. The best accuracy of 93% was obtained for 30 trees. Figure 8 shows the comparison of decision tree and random forest classifiers for predicting the Parkinson disease on voice data.

The system was subjected to the following forms of testing:

- Unit Testing:

Unit testing is the first level of testing and usually comes at the beginning of testing any application. This type of testing utilizes individual units or components of the software to be tested. We isolated the major small units of the code implemented and tested the functions, loop, or whether a statement in the program is working fine or

Fig. 7 Accuracy versus number of trees



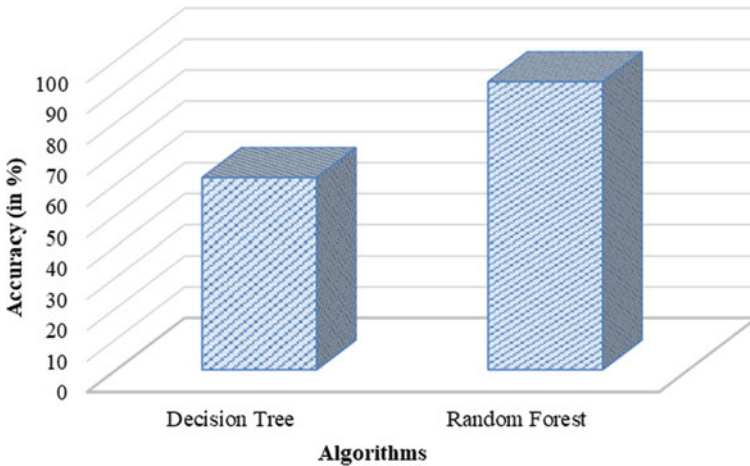


Fig. 8 Result comparison of ML algorithms for Parkinson prediction

not. This helped us optimize the code further and use proper code across the entire script. This allowed us to ultimately run two different algorithms on the model to acquire a prediction based on 14 different input parameters.

- **Integration Testing:**

It is the subsequent stage of assessment and usually occurs once the “unit testing” process is over. The aim of “integration testing” is to detect and uncover deficiencies during the interaction among modules. It utilizes different units for assessment, and these units are further combined and analyzed in an “integration testing”. Here, we tested all the libraries that have been integrated into the project to verify whether the libraries are able to communicate and provide the necessary desired output to our code. We primarily utilised the `parselMouth` library for voice synthesis, and this testing helped us to determine whether or not the several parameters supplied to the library produced the appropriate result. We then created an API using Flask and used it to detect Parkinson’s based on the input audio `_le` with the help of the `parselMouth` library. And lastly, we deployed the application to the heroku cloud platform to further check its integration with the front-end.

- **Functional Testing:**

Functional testing is the third level of testing and usually comes after the integration testing process is over. The purpose of functional testing is to verify the functionality of the entire application. This helps us to decide if the behavior of the application is as expected according to our needs. Here, we tested the functionality of the project by providing an audio file as input from the front-end website to the API that further connected our application to the back end and supplied the audio input to detect whether the person has Parkinson’s or not based on the accuracy provided from

running it against the algorithms in the model. If the person was detected to have Parkinson's, then the website displays the same and vice versa.

6 Conclusion and Future Work

The ML models used in this work for predicting Parkinson's disease turned out to be effective up to an extent in showing results for people who might be having Parkinson's disease. Upon analysis, it was found that random forest showed the highest accuracy compared to all the other models used. The results proved that the machine learning model can be used in detection of the disease, as well as to create an awareness among people. The ML model was then deployed on cloud environment for better accessibility to the end users. In the future, the model can be trained with a much greater amount of data to achieve higher performance. The work can then be effectively used as a preliminary check for a person wanting a diagnosis for Parkinson's disease.

References

1. Mei J, Desrosiers C, Frasnelli J (2021) Machine learning for the diagnosis of Parkinson's disease: a review of literature. *Front Aging Neurosci* 13(184). <https://doi.org/10.3389/fnagi.2021.633752>
2. Radhakrishnan DM, Goyal V (Mar–Apr 2018) Parkinson's disease: a review. *Neurol India*. 66(Supplement):S26–S35. <https://doi.org/10.4103/0028-3886.226451>. PMID: 29503325.
3. Poewe W, Seppi K, Tanner C et al (2017) Parkinson disease. *Nat Rev Dis Primers* 3:17013. <https://doi.org/10.1038/nrdp.2017.13>
4. Armstrong MJ, Okun MS (2020) Diagnosis and treatment of Parkinson disease: a review. *JAMA* 323(6):548–560. <https://doi.org/10.1001/jama.2019.22360>
5. Dhall D, Kaur R, Juneja M (2020) Machine learning: a review of the algorithms and its applications. In: Singh P, Kar A, Singh Y, Kolekar M, Tanwar S (eds) *Proceedings of ICRIC 2019. Lecture notes in electrical engineering*, vol 597. Springer, Cham. https://doi.org/10.1007/978-3-030-29407-6_5
6. Pramanik A, Sarker A (2021) Parkinson's disease detection from voice and speech data using machine learning. In: Uddin MS, Bansal JC (eds) *Proceedings of international joint conference on advances in computational intelligence. Algorithms for intelligent systems*. Springer, Singapore. https://doi.org/10.1007/978-981-16-0586-4_36
7. Naranjo L, Pérez CJ, Martín J, Campos-Roca Y (2017) A two-stage variable selection and classification approach for Parkinson's disease detection by using voice recording replications. *Comput Methods Programs Biomed* 142:147–156. <https://doi.org/10.1016/j.cmpb.2017.02.019> Epub 2017 Feb 22 PMID: 28325442
8. Sajal MSR, Ehsan MT, Vaidyanathan R et al (2020) Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis. *Brain Inf* 7:12. <https://doi.org/10.1186/s40708-020-00113-1>
9. Radha N, Sachin Madhavan RM, Sameera holy S (2021) Parkinson's disease detection using machine learning techniques. *Rev Argent de Clínica Psicológica* XXX:543–552. <https://doi.org/10.24205/03276716.2020.4055>

10. Jaichandran R, Leelavathy S, Usha Kiruthika S, Krishna G, Mathew MJ, Baiju J (2022) Machine learning technique based Parkinson's disease detection from spiral and voice inputs. *Eur J Mol Clin Med* 7(4):2815–2820
11. Parkinsons data set, UCI, Machine learning repository. <https://archive.ics.uci.edu/ml/datasets/parkinsons>
12. Teixeira J, Gonçalves A (2014) Accuracy of jitter and shimmer measurements. *Procedia Technol* 16:1190–1199. <https://doi.org/10.1016/j.protcy.2014.10.134>
13. Pramanik M, Pradhan R, Nandy P, Bhoi AK, Barsocchi P (2021) Machine learning methods with decision forests for Parkinson's detection. *Appl Sci* 11:581. <https://doi.org/10.3390/app11020581>
14. Açıcı K, Erdaş Ç, Aşuroğlu T, Toprak MK, Erdem H, Oğul H (2017) A random forest method to detect Parkinson's disease via gait analysis, 609–619. https://doi.org/10.1007/978-3-319-65172-9_51

A Comparative Analysis to Measure Scholastic Success of Students Using Data Science Methods



Saleem Malik, K. Jothimani, and U. J. Ujwal

1 Introduction

Understudy scholastic execution is investigated broadly to handle scholarly underachievement, expanded college dropout rates and graduation delays, among other industrious difficulties [1]. In basic terms, understudy execution alludes to the degree of accomplishing present moment and long haul objectives in schooling [2]. In any case, academicians measure understudy accomplishment from alternate points of view, going from understudies' last grades, grade point normal (GPA), to future occupation possibilities [3]. The writing offers an abundance of computational endeavors endeavoring to improve understudy execution in schools and colleges, most prominently those determined by information mining and learning examination methods [4]. Notwithstanding, disarray actually wins with respect to the viability of the current canny strategies and models.

1.1 Identifying Research Gap

These days, there are tons of examination and studies with the purpose of pathway with the appearance about anticipating understudies' conduct, amid additional connected subjects about revenue within the instructive region. Lynn and Emanuel [5] released a review paper in which the author discussed machine learning algorithms such as naive bayes, support vector machines, decision trees, neural network and k -nearest neighbor. We expanded our work in this paper to include a few more algorithms, such as clustering, association rule mining and regression. Noticing the worth

S. Malik (✉) · K. Jothimani · U. J. Ujwal
CSE Department, KVGCE, Sullia 574327, India
e-mail: baronsaleem@gmail.com

Table 1 SPIDER protocol adopted for this survey

Characteristics	Scope
Sample	Studies predicting student performance using the learning outcomes
Phenomenon of interest	List of intelligent models and technique
Design	Comparison across the identified models and techniques
Evaluation	Quality and accuracy of the approaches. Set of performance predictors of learning outcomes
Research type	Qualitative, quantitative and mixed methods

and conceivable serious expenses related with establishments of higher learning, we see the reasonable need to investigate the exercises impacting understudy achievement or inability to take care of the issue of understudy wearing down and stay away from the extreme expenses and results that it brings. Lynn and Emanuel [5] included a review of papers published between 2010 and 2020. Our research compares and contrasts different machine learning algorithms and methods used in papers published between 2000 and 2020. Consideration and rejection paradigms for collecting literature were used in Tables 1 and 2, respectively. Accordingly, the fundamental objective of this investigation is to introduce an inside and out outline of the various methods and calculations suggested that have been applied to this subject. This exploration plan to anticipate understudy execution from information gathered on understudy's movement signs on an Internet learning and understudy's segment data. We need to discover the precision of some arrangement models for anticipating understudy scholarly execution [6].

1.2 Significant Objectives of This Examination

- Finding the best characterization method on understudy informational index.
- Summarizing most far and wide used approaches for forecasting student concert.
- Among the extensively used methods, which is the most appropriate method for forecasting student concert.
- Deeply comprehend the insightful methodologies created to conjecture understudy learning results, which address the understudy scholarly execution.
- Compare the presentation of existing models on various perspectives, including their exactness, qualities and shortcomings.

Table 2 In this survey, consideration and rejection paradigms for collecting literature were used

Paradigms	Consideration	Rejection
Period	Proclaimed after 2000–2020	Proclaimed before 2000
Subject of research	Studies that use a direct link to the learning outcomes to predict student success	Studies that do not use a direct link to the learning outcomes to predict student success
Kind of record and publication locus	<ul style="list-style-type: none"> • Refereed, invited or any other conference papers presented at an academic or professional conference • Scholarly books, monographs and chapters • Journal publications • Course materials 	<ul style="list-style-type: none"> • Not refereed, invited or any other conference papers presented at an academic or professional conference
Accessibility and availability	Open access and full articles available	Open access and full articles available
Vocabulary	Written in English	Written in other language
Appropriateness	<ul style="list-style-type: none"> • Experiential papers • Participating in learning conditions • Fundamental design is investigating instructive setting 	<ul style="list-style-type: none"> • Unapplied papers • Not participating in other conditions than learning • Fundamental design is not investigating instructive setting

2 Research Design Search and Screening Strategy

This examination has followed the suggestion given by [7, 8] to deciding example, phenomenon of interest, design, evaluation and research type (SPIDER) as demonstrated in Table 1. The objective here is to distinguish the holes or focal issues with a specific spotlight on student’s execution investigation and forecast related writing. This nonpartisan review is definitely not a comprehensive one; rather, it addresses the important writing urgent to the exploration addresses outlined [8].

Once the taxonomy is defined, we have also received the regular methodology for looking through significant writing too. In this methodology, we search through the Google Scholar which assists us in arranging a caution with search strings, “educational data mining and student performance,” “learning analytics and student performance,” “student performance and teaching quality,” “student performance and domain knowledge,” “Predicting students’ performance,” “Predicting algorithm students,” “Recommender systems prediction students,” “Artificial neural network prediction students,” “Algorithms analytics students and Students analytics prediction.” Because of this setup, Google Scholar routinely sends a rundown of as of late distributed pertinent papers to our email. As such, we identified and searched seven major online bibliographic databases, which contain engineering and science publications. These databases include the ACM digital library, IEEE Xplore, Google Scholar, Science Direct, Scopus, Springer and Web

of Science. These are the common databases searched by software engineering reviews and are expected to incorporate the studies investigating the predictive modeling of student outcomes. Other electronic databases, such as DBLP and CiteSeer, were excluded from the search since their results are inclusive within the previous seven databases. Consideration and rejection standards for gathering writing in this survey are shown in Table 2 [6, 8]. Comparable survey is carried out by [9] with few conditions. Figure 1 demonstrates the strategy received for leading this study. Analogous methodology was used by [8] with few taxonomy, search strings and methods. The above half and half inquiry approach recovers different articles which are separated efficiently to get the rundown of chosen articles. The suggestion of the SPIDER Statement [7, 10] assists us in the decision. As a basic development, subtitle to accept examines identified with understudy execution investigation or forecast is scrutinized [8]. This interaction helps us in distinguishing particular examinations as an expected possibility for this review. We read the theoretical and catchphrase referenced in every single one of them for additional screening, and it sifts through immaterial investigations. In the subsequent stage, we have perused the full text and attempted to distinguish whether the commitment of the investigation is urgent to the examination inquiries of this overview. We additionally dispose of those investigations which are not pleasing and contain hazy system, dataset or commitment. In the following stage of the subjective blend, seeker articles are added to EndNote [11, 12], a reference of the board device. Here, important feature of EndNote is to assist in gathering of related articles. Moreover, general note portion of EndNote is used to keep a track on the key revelations, gigantic responsibilities and limitations of each article which eventually urges us to assessment.

This research played out a deliberate survey where the pertinent scholastic works anticipating understudy execution utilizing learning results were recognized, chosen and fundamentally assessed utilizing a few standards, as introduced in the outcome area. To smooth out our commitments, we detailed three key research questions as follows:

- How is understudy scholarly execution estimated utilizing learning results?
- What methods are formulated to conjecture understudy scholastic execution utilizing learning results?
- Which is the best prevailing method of understudy performance utilizing learning results?

3 Methods for Predicting Student's Triumph that Are Far-Reaching

In present data mining prediction strategies, prescient displaying remains generally utilized during foreseeing understudy execution. To fabricate the prescient demonstrating, there are a few assignments utilized, which are order, relapse and arrangement. The most mainstream assignment to anticipate understudy execution is

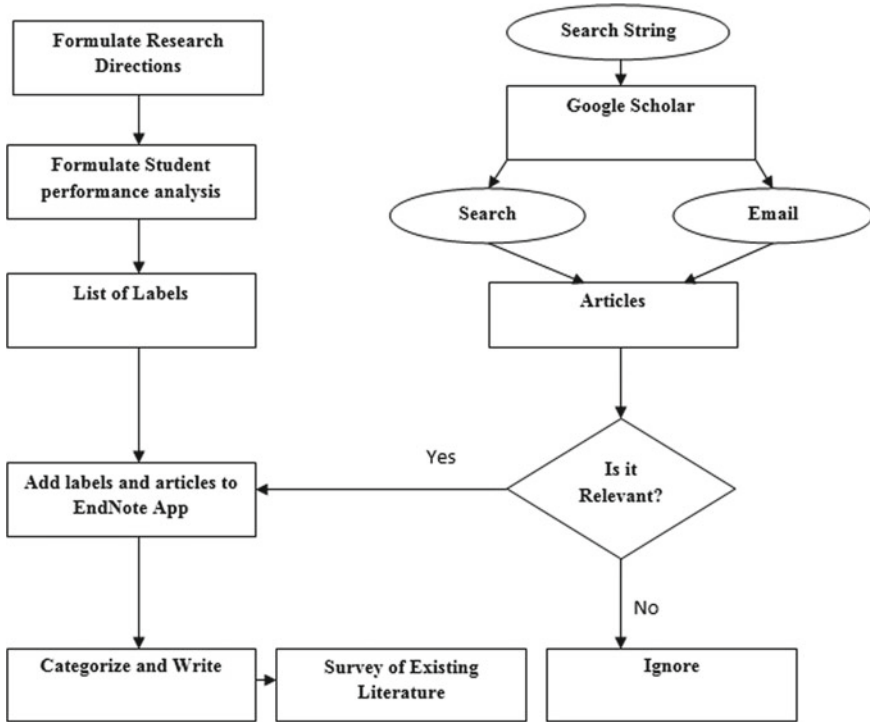


Fig. 1 Methodology adopted for this survey

grouping. There are a few calculations under characterization chore with the purpose have been put into anticipate understudy execution. Amid calculations utilized are decision tree (DT), support vector machine (SVM), neural networks (NN), *K*-nearest neighbor (KNN) and naive Bayes (NB). Then, each particular utilized data mining strategies assembled along calculations during anticipating understudy execution prospective depicted in the course of following segment.

3.1 Decision Trees

Decision trees are the most popular order model. The decision tree starts in the midst of a root center, from which customers can decide. Customers can circularly part per center point from this center point dependent on the decision tree learning computation. The closing stage consequence is a decision tree, with apiece branch tending to a possible choice circumstance and its result [13]. The vast majority of scientists have utilized this method on account of its effortlessness and understandability to reveal little or huge information structure and anticipate the worth [14, 10, 15]. Romero and Ventura [7] said that the choice tree models are handily perceived in light of their

thinking cycle and can be straightforwardly changed over in the direction of deck based on IF-THEN principles [16]. As demonstrated in table, there are around eight papers that have utilized decision tree as their strategy to assess understudy execution. Instances of past examinations utilizing decision tree strategy are foreseeing abandon highlights about understudy information intended for scholarly execution [14], anticipating third semester execution of MCA understudies [17] in addition to furthermore foreseeing specific reasonable vocation for an understudy through their personal conduct standards [18]. The understudy's execution assessment depends on highlights extricated from logged information in a training online framework. The instances of dataset are understudy last grades [19], last CGPA [3] and marks acquired specifically colloquium [16]. Such datasets be considered also broke down to discover the primary credits or factors that may influence the understudy execution [15, 20]. At that point, the appropriate information mining calculation will be explored to foresee understudy execution [21]. Chatterjee and Hadi [22] have thought about the characterization strategies for foreseeing understudy execution in their examination [12]. Helal et al. [23] researched the exactness of grouping models to anticipate student's movement in tertiary training [24].

3.2 *Neural Network*

Another well-known technique used in instructive data mining is neural networks. The advantage of using a neural network is that it can identify all possible relations between indicators [24]. Even in complex nonlinear connections between dependent and free variables, neural network could perform a complete identification devoid of any uncertainty [25]. As a result, among the various forecasting techniques, the neural network strategy is chosen as the best. Seven papers were distributed using the neural network technique as part of the meta-investigation analysis. Certain papers bestow an artificial neural network model for predicting the execution of understudies [25, 26]. Neural network breaks down the credits as follows: confirmation data [27], understudies' mentality toward self-controlled knowledge and scholarly execution [28]. The respite is the similar articles, but with a decision tree approach, in which analysts compared the two strategies to see which one is the better prediction technique for analyzing understudies' performance. Table summarizes the implications of forecast accuracy.

3.3 *Naive Bayes*

Aforementioned classifier uses probabilities to predict class chipping in, such as the probability that a given example belongs to a particular class. There have been a few Bayes calculations developed, with Bayesian and naive Bayes being the two most common. The naive Bayes calculation assumes that the effect of a quality on a

given class is independent of the estimates of different people ascribes [29]. Naive Bayes calculation is likewise a possibility for specialists to make a forecast. In the midst of thirty papers, seven papers were presented that have utilized naive Bayes calculations to appraise understudy execution. Specific target about every one of each seven papers holds up toward tracking down specific best forecast method in foreseeing understudy execution via building examinations [10, 12, 21, 30]. Their exploration showed that naive Bayes has utilized all of characteristics contained in the information. At that point, it examined every single one of them to show the significance and independency of each ascribes [10].

3.4 K-Nearest Neighbor

The K -nearest neighbor classifier tackles a completely different approach to grouping. They do not make any unmistakable global models, but they do test them locally and verifiably. The basic idea is to organize another item by evaluating the class estimations of the K most indistinguishable data focuses. In K -nearest neighbor classifiers, the solitary learning task is to select two important boundaries: the number of neighbor's k and the distance metric d [31]. As shown in Table, each of the three papers examined in this study revealed that K -nearest neighbor provided the best exhibition with the greatest precision. According to [32], the K -nearest neighbor strategy needed less effort to distinguish between a slow student, a normal student, a fantastic student and a superb student [19, 12]. K -nearest neighbor provides a fair level of precision within determining a specific example for a student's movement within advanced schooling [24].

3.5 Support Vector Machine

While class confines remain nonlinear but there is insufficient knowledge on the road to be trained composite nonlinear models, support vector machines (SVM) are the best technique. SVM focuses solely on class boundaries; focuses that can be grouped in any way are skipped. The objective is to find the "thickest hyperplane," which divides the groups [31]. SVM is a managed learning technique with the aim of pattern recognition. Three articles have used SVM as a tool to predict the understudy's presentation. Hamalainen and Vinni [32] chose support vector machine as their forecasting tool because it was well suited to small datasets [7]. SVM has a good speculation potential and is faster than other techniques, according to [33]. Polyzou and Karypis [31] Then, there is [32]'s analysis. Helal et al. [23] showed that the SVM technique yielded the highest expectation precision in identifying understudies on the verge of fizzling [24]. The following table depicts the aftereffects of anticipation.

3.6 Association Rule Mining

Preeminent association rule mining is a specific one, notable and famous information digging strategies utilized broadly for instructive purposes [34]. It is advantageous surely for understanding the educational parts of realizing which thusly assist the scholastic executive with outlining arrangements [35, 36]. Presently, amount of studies subsist which utilize the affiliation rule digging for investigating understudy execution [24, 34, 35]. To acquire a bunch of significant guidelines, it is crucial to pre-decide the insignificant help and certainty. Be that as it may, it is hard for an instructor to choose these two info boundaries ahead of time. Moreover, the quantity of got rules might be too high at times, and the greater part of them are non-fascinating and with low understandability. A joined proportion of aggregate intriguing quality may likewise be successful in this unique situation [35].

3.7 Regression

A condition between the reliant variable and one or different autonomous factors characterizes the relationship in relapse examination. In conjecture backdrop, it is likewise utilized intended for assessing a certain significance about personage indicators and comprehends specific relationship between capricious [37]. The inescapable use of relapse is obvious in instructive information mining writing. It is in fact every now and again utilized in EDM reads for building up or discrediting the effect of educating quality. Many exploration works have investigated the relapse examination for understudy execution examination [38–40].

3.8 Clustering

It is an idea of separating information into gatherings of comparable articles where each gathering, or bunch, comprises of items that are like each other and unlike the object of different gatherings. Apportioning and progressive strategies are the two general classifications of grouping, while k -implies and agglomerative various leveled methods are extremely famous among them. Customary grouping approaches are not reasonable when both quantitative and subjective ascribes exist together [41, 42]. Be that as it may, the majority of the instructive dataset contains both of these kinds. The vast majority of the grouping approach moreover requires an information boundary which decides the quantity of coming about bunch. By the by, EDM scientists have impressively utilized this solo strategy in understudy execution investigation writing [11, 43].

4 Prediction and Accuracy

At the point when researchers utilize a computation to isolate the parts of a dataset with two conditions (for instance, positive and negative), they can create a two-class chaos structure, which tends to the quantity of segments that were effectively anticipated and the number that was mistakenly characterized [4, 6, 11]. True positives (TP) are those examples of positive information that the estimation effectively distinguished as obvious, while false negatives are those mistakenly marked as false negative (FN). Then again, true negatives (TN) are negative parts that are precisely named all things considered, while false positives are those that are mistakenly expected as false positives (FP).

A collector working trademark (CWT) twist [58, 56] or an accuracy review (AR) twist [6, 56] can be made utilizing disarray structures. Specialists can now, finally, be trusted. To assess the introduction of the gathering, compute the area under curve (AUC) of the CWT twist or the AR twist. Previously, specialists envisioned a couple of disorder lattice rates [13, 18, 56]. There are just two disorder framework orders in some of them.

1. Sensitivity (Eq. 1) is depicted as the level of fruitful understudies who were accurately named “effective” among all effective students [56]. It centers on bringing down FN levels.
2. For all non-fruitful understudies, explicitness (Eq. 2) is the extent of non-fruitful understudies who are wrongly delegated effective students [56]. To perceive negative results.
3. Precision (Eq. 3) and the extent of effective understudies accurately delegated “fruitful” for all understudies anticipated as “fruitful” by the algorithm [56]. It is centered on diminishing FP.
4. For all non-effective understudies, the negative prescient worth (Eq. 4) addresses the extent of non-effective understudies who are wrongly delegated fruitful students [56]. To center bringing down FP.

These four rates are referred to as simple confusion matrix rates [56].

1. Level of true positives:

$$\frac{TP}{TP + FN} \quad (1)$$

2. Level of negative rates:

$$\frac{TN}{TN + FP} \quad (2)$$

3. High probability of success:

$$\frac{TP}{TP + FP} \quad (3)$$

4. Low probability of success:

$$\frac{TN}{TN + FN} \quad (4)$$

Exactness assessment will be utilized to assess every estimation (Eq. 5). We utilized four variables: true positive (TP), true negative (TN), false positive (FP) and false negative (FN) to decide the level of effectively perceived name (FN). To decide it, we first add the measures of information that are adequately portrayed by the classifier, apportioned by the absolute number of information focuses organized, as illustrated in condition under.

5. Accuracy

$$\frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

Since a couple of models are ordinarily assembled, it is important to assess them and pick the most fitting. The confusion network is regularly utilized while assessing the display of plan calculations [59]. To assess the model, particular execution steps are fused; essentially, all extents of execution are reliant on the disorder cross section and the numbers in it [46, 56]. To make a more exact result, these activities are surveyed together.

5 Findings and Observations

Student concert forecasting contains recognizing an undisclosed mark connected with student [5]. On the other hand, several attributes can influence students' concert, building aforementioned chore difficult toward accomplishment. Each essential could consist of economic condition of a student, demographic distinctiveness and students' psychological profiles, precedent academic experiences, dissimilar cultural backgrounds and communications among fellow students [5]. Despite the above, on behalf of every institution on the way to carry out a right performance forecasting, extremely, they have got to first recognize the menace attributes that might influence forecasting results. Every educational organization should constantly response the significant query ahead of captivating forecasting steps; "What are the important risk factors or variables for predicting student performance?" These menace attributes might contain in use student performance forecasting methods, in addition to the datasets measured to obtain forecasting outcome. The datasets might consist of attributes like parents' education, parents' income, first child, gender, earlier semester grade, attendance, GPA, scholarship, etc. Diverse Institutions mull over poles apart datasets on the way to forecast student concert. On the other hand, different datasets effort sounds through anecdotal methods pertaining to forecasting. For that reason, forecasted results could contrast depending on methods utilized. Even though wavering

attributes with the aim of concern forecasting results be evident, various factors might be fragile as well as intricate toward to recognize and classify exclusive of applying a more refined analysis. Consequently, utilizing recent data mining methods like clustering, regression and neural networks might precisely forecast student concert (pass/fail) contrast with new methods. The results from accurate forecast are capable to assist institutions in the direction to attain excellence in education.

6 Result

The consequences of the usually utilized understudy execution forecast strategies above from 2000 to 2020 were dissected and used to plot a chart of how they contrast in their general expectation precision. Comparable methodology is utilized in [48, 49]. The graph is appeared in Fig. 2. Past examinations’ techniques for anticipating understudy achievement and their results were contrasted and the objective of figuring out which strategy had the most noteworthy precision, and the outcomes were plotted on the chart in Fig. 2. The expectation exactness of the customary expectation techniques used to conjecture understudy achievement that the creators explored in this report, all assembled by their calculations from 2000 to 2020, is appeared in Table 3. In contrast with any remaining strategies, the clustering approach had a high expectation exactness of 83% as appeared in Fig. 2. The neural network was the second generally precise, with a score of 82%. The regression and SVM techniques were then utilized, the two of which gave a comparative degree of precision 81%. KNN had the most reduced exactness in anticipating understudy results 74%.

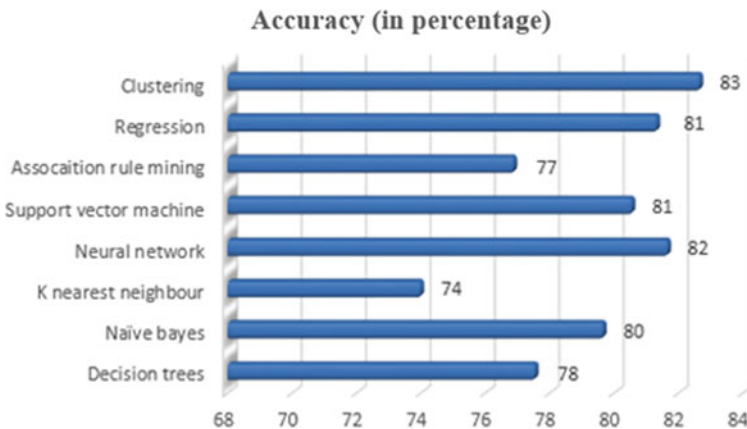


Fig. 2 Methods for predicting student success with high accuracy

Table 3 Results of widely used approaches' prediction accuracy

Decision trees	Accuracy (%)	83.34	82.05	78.08	90.75	70.00	88.45
	Reference	[37]	[23]	[43]	[43]	[44]	[45]
Neural network	Accuracy (%)	71.14	77.84	80.00	76.84	75.00	69.67
	Reference	[46]	[47]	[48]	[49]	[50]	[51]
Naïve Bayes	Accuracy (%)	75.00	71.14	84.14	80.00	83.00	82.13
	Reference	[12]	[30]	[10]	[21]	[29]	[52]
<i>K</i> -nearest neighbor	Accuracy (%)	71.04	72.19	65.90	68.98	67.90	73.92
	Reference	[52]	[21]	[53]	[42]	[54]	[55]
Support vector machines	Accuracy (%)	80.14	73.17	83.12	67.15	65.15	70.13
	Reference	[32]	[56]	[31]	[9]	[24]	[31]
Association rule mining	Accuracy (%)	80.00	98.70	90.10	73.30	75.00	76.45
	Reference	[24]	[35]	[36]	[38]	[34]	[57]
Regression	Accuracy (%)	86.00	52.32	62.50	94.42	72.00	76.65
	Reference	[38]	[37]	[39]	[40]	[49]	[50]
Clustering	Accuracy (%)	98.56	78.08	83.56	92.00	87.84	96.00
	Reference	[11]	[43]	[41]	[37]	[23]	[43]

7 Final Thoughts and Way Forward

Understudy execution is a basic factor that should be completely analyzed if the objective of preparing in higher instructive foundations and at all degrees of schooling is to be met. This is because of the way that anticipating understudy achievement helps institutional pioneers in building up their instructional constructions. This examination pointed toward looking into the generally utilized arrangement methods for foreseeing understudy execution. Among the broadly utilized strategies to anticipate understudy execution, the clustering strategy end up being the best technique for foreseeing understudy execution contrasted with association rule mining, naive Bayes regression, support vector machines and *k*-nearest neighbor. In light of its convenience and capacity to uncover little or enormous data structures and anticipate values, it gives high exactness. Overall, the findings of this audit will assist instructors in intentionally monitoring students' success by using the least demanding and most precise technique to predict understudy execution. The creators accept that utilizing the best forecast technique assists instructors with gathering understudies' exhibition, which permits early intercessions that may acquire an increment brilliant scholastic execution rate, in this way advancing training with great achievers. Makers will utilize the data from this review as a beginning stage for other comparative examinations in the educational information mining area later on. It is likewise pivotal to sort out the most ideal approach to build the accuracy of different methods.

References

1. Gedeon TD, Turner S (1993) Explaining student grades predicted by a neural network. In: Proceedings of 1993 international joint conference on neural networks, IJCNN'93-Nagoya, vol 1. IEEE, pp 609–612
2. Aghabozorgi S, Mahroei H, Dutt A, Wah TY, Herawan T (2014) An approachable analytical study on big educational data mining. In: International conference on computational science and its applications, Springer, pp 721–737
3. Asif R, Merceron A, Ali SA, Haider NG (2017) Analyzing undergraduate students' performance using educational data mining. *Comput Educ* 113:177–194
4. Baker RS (2014) Educational data mining: an advance for intelligent systems in education. *IEEE Intell Syst* 29(3):78–82
5. Lynn ND, Emanuel AWR (2021) Using data mining techniques to predict students' performance: a review. In: IOP Conference series: materials science and engineering
6. Khanna L, Singh SN, Alam M (2016) Educational data mining and its role in determining factors affecting student's academic performance: a systematic review. In: 2016 1st India international conference on information processing (IICIP), IEEE, pp 1–7
7. Romero C, Ventura S (2010) Educational data mining: a review of the state of the art. *IEEE Trans Syst Man Cybern Part C Appl Rev* 40(6):601–618
8. Khan A, Ghosh SK (2020) Student performance analysis and prediction in classroom learning: a review of educational data mining studies. *Educ Inf Technol*
9. Lemay DJ, Baik C, Doleck T (2021) Comparison of learning analytics and educational data mining: a topic modeling approach. *Comput Educ Artif Intell*
10. Wook M, Yusof ZM, Nazri MZA (2016) Educational data mining acceptance among undergraduate students. *Educ Inf Technol* 22(3):1195–1216
11. Pena-Ayala A (2014) Educational data mining: a survey and a data mining-based analysis of recent works. *Expert Syst Appl* 41(4):1432–1462
12. Koedinger KR, D'Mello S, McLaughlin EA, Pardos ZA, Rose CP (2015) Data mining and education. *Wiley Interdisc Rev Cogn Sci* 6(4):333–353
13. Kumar DA, Selvam RP, Kumar KS (2018) Review on prediction algorithms in educational data mining. *Int J Pure Appl Math* 118(8):531–537
14. Ogor EN (2007) Student academic performance monitoring and evaluation using data mining techniques. In: Electronics, robotics and automotive mechanics conference, IEEE, pp 354–359
15. Dutt A, Ismail MA, Herawan T (2017) A systematic review on educational data mining. *IEEE Access* 5:15991–16005
16. Sweeney M, Rangwala H, Lester J, Johri A (2016) Next-term student performance prediction: a recommender systems approach. *J Educ Data Min* 8(1):22–51
17. Wang Y, Ostrow K, Adjei S, Heffernan N (2016) The opportunity count model: a flexible approach to modeling student performance. In: Proceedings of the third (2016) ACM conference on learning@ Scale, ACM, pp 113–116
18. Hu X, Cheong CWL, Ding W, Woo M (2017) A systematic review of studies on predicting student learning outcomes using learning analytics. In: Proceedings of the seventh international learning analytics & knowledge conference, ACM, pp 528–529
19. Zollanvari A, Kizilirmak RC, Kho YH, Hernandez-Torrano D (2017) Predicting students' GPA and developing intervention strategies based on self-regulatory learning behaviors. *IEEE Access* 5:23792–23802
20. Bendikson L, Hattie J, Robinson V (2011) Identifying the comparative academic performance of secondary schools. *J Educ Adm* 49(4):433–449
21. Al-Obeidat F, Tubaishat A, Dillon A, Shah B (2017) Analyzing students' performance using multicriteria classification. *Clust Comput* 21(1):623–632
22. Chatterjee S, Hadi AS (2015) Regression analysis by example. Wiley, New York
23. Helal S, Li J, Liu L, Ebrahimie E, Dawson S, Murray DJ (2018) Identifying key factors of student academic performance by subgroup discovery. *Int J Data Sci Analytics* 7(3):227–245

24. Jishan ST, Rashu RI, Haque N, Rahman RM (2015) Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique. *Decis Analytics* 2(1):1
25. Natek S, Zwilling M (2014) Student data mining solution-knowledge management system related to higher education institutions. *Expert Syst Appl* 41(14):6400–6407
26. Christian TM, Ayub M (2014) Exploration of classification using NBTree for predicting students' performance. In: 2014 international conference on data and software engineering (ICODSE), IEEE, pp 1–6
27. Quadri MMN, Kalyankar NV (2010) Drop out feature of student data for academic performance using decision tree techniques. *Global J Comput Sci Technol* 10(2)
28. Backenkohler M, Wolf V (2017). Student performance prediction and optimal course selection: an "MDP approach. In: International conference on software engineering and formal methods, Springer, pp 40–47
29. O'Connell KA, Westl E, Crosslin M, Berry TL, Grover JP (2018) Student ability best predicts final grade in a college algebra course. *J Learn Analytics* 5(3):167–181
30. Kumar M, Singh AJ, Handa D (2017) Literature survey on student's performance prediction in education using data mining techniques. *Int J Edu Manage Eng* 6:40–49
31. Chaturvedi R, Ezeife CI (2017) Predicting student performance in an ITS using task-driven features. In: 2017 IEEE international conference on computer and information technology (CIT), IEEE, pp 168–175
32. Hamalainen W, Vinni M (2006) Comparison of machine learning methods for intelligent tutoring systems. In: *Intelligent tutoring systems*, Springer, pp 525–534
33. Polyzou A, Karypis G (2019) Feature extraction for next-term prediction of poor student performance. *IEEE Trans Learn Technol*
34. Damasevicius R (2010) Analysis of academic results for informatics course improvement using association rule mining. In: *Information systems development*, Springer, Berlin, pp 357–363
35. Angeli C, Howard S, Ma J, Yang J, Kirschner PA (2017) Data mining in educational technology classroom research: can it make a contribution? *Comput Educ* 113:226–242
36. Adjei SA, Botelho AF, Heffernan NT (2016) Predicting student performance on post-requisite skills using prerequisite skill data: an alternative method for refining prerequisite skill structures. In: *Proceedings of the sixth international conference on learning analytics & knowledge*, ACM, pp 469–473
37. Goos M, Salomons A (2016) Measuring teaching quality in higher education: assessing selection bias in course evaluations. *Res High Educ* 58(4):341–364
38. Chen W, Brinton CG, Cao D, Mason-singh A, Lu C, Chiang M (2018) Early detection prediction of learning outcomes in online short-courses via learning behaviors. *IEEE Trans Learn Technol* 12(1):44–58
39. Ustunluoglu E (2016) Teaching quality matters in higher education: a case study from Turkey and Slovakia. *Teachers Teaching* 23(3):367–382
40. Kesavaraj G, Sukumaran S (2019) A study on classification techniques in data mining. In: 2019 11th International conference on computing, communications and networking technologies (ICCCNT), IEEE, pp 1–7
41. She HC, Cheng MT, Li TW, Wang CY, Chiu HT, Lee PZ et al (2012) Web-based undergraduate chemistry problem-solving: the interplay of task performance, domain knowledge and web-searching strategies. *Comput Educ* 59(2):750–761
42. Osmanbegovic E, Suljic M Data mining approach for predicting student performance. *Econ Rev* 10(1)
43. Meier Y, Xu J, Atan O, van der Schaar M (2016) Predicting grades. *IEEE Trans Signal Process* 64(4):959–972
44. Quille K, Bergin S (2018) Programming: predicting student success early in CS1. A re-validation and replication study. In: *Proceedings of the 23rd annual ACM conference on innovation and technology in computer science education*, ACM, pp 15–20
45. Yu L, Lee C, Pan H, Chou C, Chao P, Chen Z et al (2018) Improving early prediction of academic failure using sentiment analysis on self evaluated comments. *J Comput Assist Learn* 34(4):358–365

46. Chanlekha H, Niramitranon J (2019). Student performance prediction model for early-identification of at-risk students in traditional classroom settings. In: Proceedings of the 10th international conference on management of digital ecosystems—MEDES '19, ACM, pp 239–245
47. Yaacob WFW, Nasir SAM, Yaacob WFW, Sobri NM (2019) Supervised data mining approach for predicting student performance. *Indonesian J Electr Eng Comput Sci* 16(3):1584–1592. ISSN: 2502-4752. <https://doi.org/10.11591/ijeecs.v16.i3.pp1584-1592>
48. Salal YK, Abdullaev SM, Kumar M (2019) Educational data mining: student performance prediction in academic. *Int J Eng Adv Technol (IJEAT)* 8(4C). ISSN: 2249-8958
49. Khan A, Ghosh SK (2020) Student performance analysis and prediction in compounded schooling: a review of educational data mining studies. *Educ Inf Technol* 26:205–240. <https://doi.org/10.1007/s10639-020-10230-3>
50. Bin Mat U, Buniyamin N, Arsad PM, Kassim R (2013) An overview of using academic analytics to predict and improve students' achievement: a proposed proactive intelligent intervention. In: Engineering education (ICEED) 2013 IEEE 5th conference on, IEEE, pp 126–130
51. Ibrahim, Z, Rusli D (2007) Predicting students academic performance: comparing artificial neural network, decision tree and linear regression. In: 21st Annual SAS Malaysia forum
52. Romero C, Ventura S (2010) Educational data mining: a review of the state of the art. *Trans Sys Man Cyber Part C* 40(6):601–618. <https://doi.org/10.1109/TSMCC.2010.2053532>. doi:10.1109/TSMCC.2010.2053532
53. Quadri MM, Kalyankar N Drop out feature of student data for academic performance using decision tree techniques. *Global J Comput Sci Technol* 10(2)
54. Sukumar Letchuman MW Mac Roper, Pragmatic cost estimation for web applications
55. Angeline DMD (2013) Association rule generation for student performance analysis using Apriori algorithm. *SIJ Trans Comput Sci Eng Appl (CSEA)* 1(1):12–16
56. Chicco D, Tötsch N, Jurman G (2021) The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Min*
57. Kamley S, Jaloree S, Thakur RS (2016) A review and performance prediction of students' using association rule mining based approach. *Data Min Knowl Eng* 8(8):252–259
58. Livieris IE, Drakopoulou K, Mikropoulos TA, Tampakas V, Pintelas P (2018) An ensemble-based semi-supervised approach for predicting students' performance. In: Research on e-learning and ICT in education, Springer, pp 25–42
59. Mimis M, El Hajji M, Es-saady Y, Oueld Guejdi A, Douzi H, Mammass D (2019) A framework for smart academic guidance using educational data mining. *Educ Inf Technol* 24(2):1379–1393

Gesture-Controlled Speech Assist Device for the Verbally Disabled



Shreeram V. Kulkarni, Shruti Gatade, Vasudha Hegde, and G. Manohar

1 Introduction

Sign language is a method of communication that makes use of the user's movements. Communication between deaf and hearing persons has a substantial disadvantage when contrast to communication among blind and ancient visual people [1]. The blind communicate openly in languages, whereas the deaf has their own set of symbols that they refer to as language. As per an assessment undertaken by the Government of India in 2011, India's census of dumb people is estimated to be out with 20.02 lakhs, with 56 and 46% of the population suffering from speech and hearing abnormalities, correspondingly. According to survey data, 1.33 billion people encounter communication issues in everyday situations, with sign language being utilized to express messages [2]. Every typical human being's primary mode of communication is speech. However, those who are speech impaired uses sign languages. The majority of people are unable to comprehend sign language. As a result, it becomes difficult for a person with speech impairment to convey his or her opinions and beliefs. This creates a barrier to communication between the deaf and the rest of society. A device or tool that can convert hand motions into auditory words is required to tackle this problem. Several devices have been developed, however, all of them have limitations with mobility, size, and cost. The primary goal of the hand sign translation system is to recognize and communicate via hand gestures [3].

The verbally-disabled people often find it difficult to convey what they want to say. To overcome this problem, they make use of sign language. Sign language is a language that uses manual communication to convey meaning. They employ hand

S. V. Kulkarni (✉) · S. Gatade · V. Hegde
NitteMeenakshi Institute of Technology, Yelahanka, Bengaluru 560064, India
e-mail: Shreeram.kulkarni@nmit.ac.in

G. Manohar
DXC Technology, Bengaluru 560100, India

gestures, movement of fingers, and orientation to convey a speaker's idea. A gesture-controlled device is one which has human gesture recognition capabilities [4, 5]. These gestures are used to perform specific predefined functions. These gestures can be in various physical forms. The most common one is by the use of hand gestures. A gesture-controlled speech assistance device is one which converts hand gestures into speech in the form of text displayed on a screen. These messages are in a language such as English to enable the speaker to convey their message to another person. In our project, the glove maps the orientation of the hand and fingers with the help of flex sensors and an accelerometer [6, 7]. The data is then transmitted to the Arduino Nano microcontroller, where data is analyzed according to the program written [8]. This system is modeled specifically to help convert sign language to speech. Thus, with the help of this, the barrier faced by verbally-disabled people in communicating with the society can be reduced to a great extent [9, 10].

2 System Block Diagram

The overall system block diagram is shown in Fig. 1. System facilitates two-way interaction between the disabled and the general population. Transmitter and receiver are the two components of the system. The flex sensor plays the major role [3, 10]. The glove is fitted with flex sensors along the length of each finger and the thumb.

The flex sensors give output in the form of voltage variation that varies with degree of bend. This flex sensor output is given to the ADC channels of microcontroller. It processes the signals and performs analog-to-digital signal conversion. Further, the processed data is sent in a wireless manner to the receiver section. In this section, the

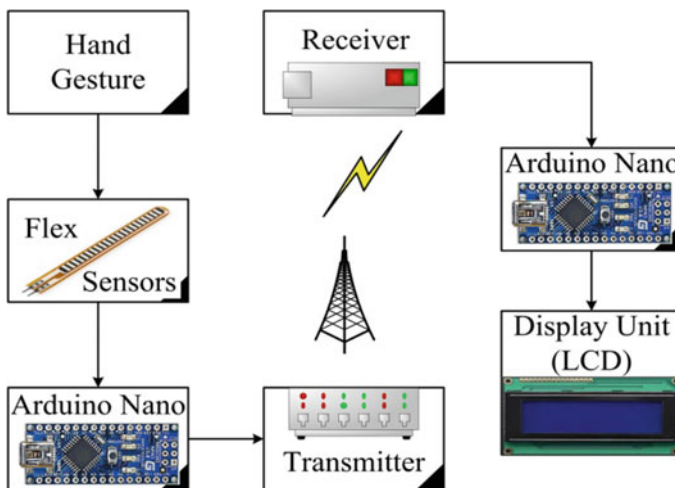


Fig. 1 Overall working block diagram

gesture is recognized, and the corresponding output is displayed on LCD [11–13]. Before the data is sent wirelessly, Arduino Nano is programmed in such a manner that it processes the hand gestures, the resistance values which is measured by the flex sensors attached to each finger [14]. These measured values of resistance are sent as analog input to the microcontroller which compares it with.

A flex sensor or bend sensor is a sensor that measures the amount of deflection or bending. By combining the flex sensor with a static resistor to create a voltage divider, a variable voltage that can be read by a microcontroller’s analog-to-digital converter is produced. A user makes a gesture using his/her hand. Due to this, there is a change in resistance of flex sensors attached to the glove. This causes a change in voltage drop across the flex sensors [15]. The glove was equipped with flex sensors, a gyroscope, and an accelerometer. These sensors detect hand motions with varied degrees of resistance and according to finger angle movement. The Filipino Sign Language Alphabet, as illustrated in Fig. 2 [16], was utilized as the foundation for assigning a message to each letter.

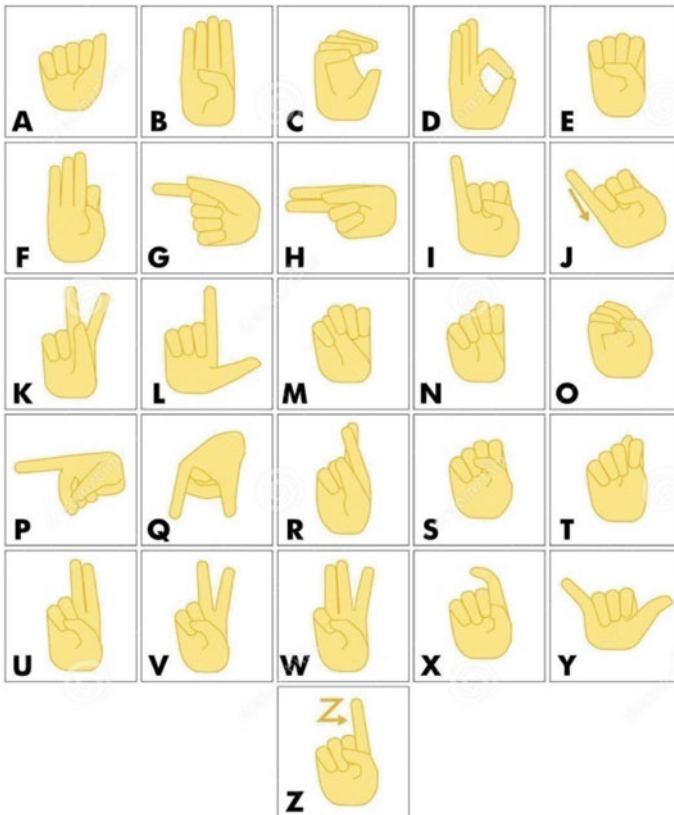


Fig. 2 American sign language illustrations and vectors standards [16]

The flex-powered glove makes it simple to communicate the actions depicted in the figure. Image processing is employed in today's gesture detection systems for the recognition of made gestures. These systems use Web cameras, morphology, gray scaling and thresholding, feature extraction, and binarization to name a few techniques. Because the processes required include flex sensing, microcontroller computations, and synchronization between the microcontroller and output devices such as a smaller LCD display, the recommended system requires significantly less processing. As a result, the cost of the equipment reduces the computing time and complexity [10, 13].

3 Working and Hardware Implementations of the System

Figure 3 depicts the device's design, while Fig. 4 depicts the arrangement of the various electronic components utilized in the device's design, such as the main controller, sensors, and output modules. The output of the flex sensor is given as analog input to the Arduino Nano. The Arduino Nano is a small, complete, and breadboard-friendly microcontroller board based on the ATmega328P. The input from the flex sensors is then converted into discrete digital values by the Arduino Nano. Arduino will compare actions assigned to different gestures against different values of input. Corresponding output is sent via 433 MHz radio frequency (RF) module which is a (usually) small electronic device used to transmit and/or receive radio signals between two devices. In an embedded system, it is often desirable to communicate with another device wirelessly.

Signal sent by input side 433 MHz RF module transmitter is received by the RF module receiver. This value is processed in the Arduino, where the corresponding phrases are displayed using 16 * 2 LCD display unit with required conversions phrases assigned to the corresponding values. If the condition matches, then the phrases are sent wirelessly to the output side microcontroller. If the condition fails, i.e., there is no match for the certain set of input values from the flex sensors, in that case, LCD will not display anything. In this manner, different gestures are recognized, and phrases are continuously sent to the LCD display.

On the output side, the receiver and liquid crystal display are interfaced with Arduino Nano and are mounted on a stripboard as shown in Fig. 4. A 9 V battery is mounted on the stripboard for power supply to Arduino. The flowchart of software implementation is shown in Fig. 5.

Messages assigned to gesture and flowchart

```
const char *m1 = "    HI HOW ARE YOU?"; const char *m2 =
"PEACE"; const char *m3 = "WHAT?"; const char *m4 =
"SUPER"; const char *m5 = "    SHOULD DO SOMETHING ABOUT
IT ";
```

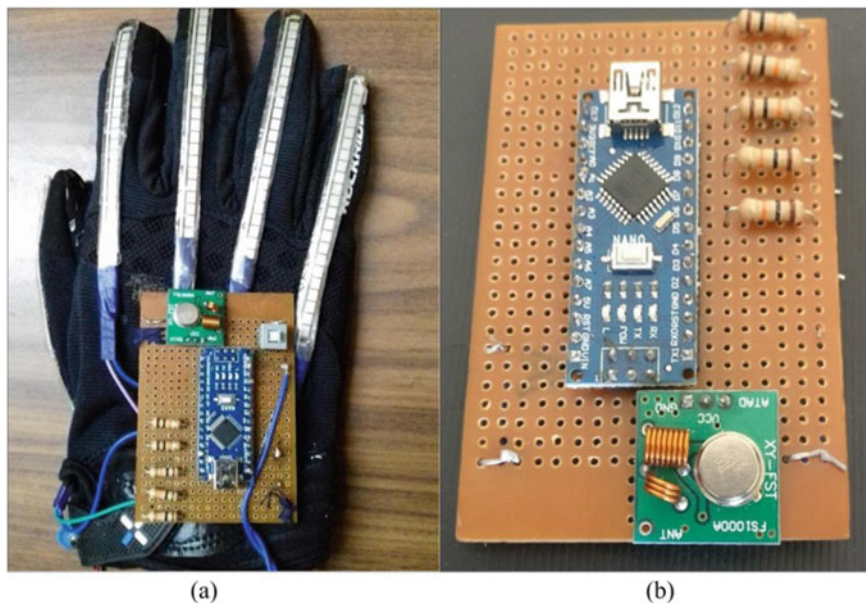
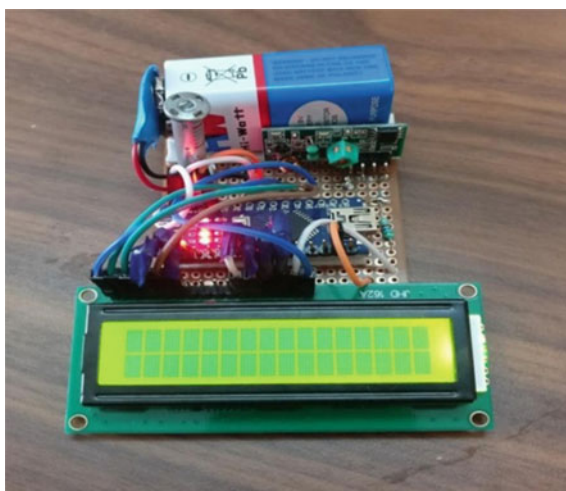



Fig. 3 Final model of the glove **a** top view, **b** rear view of the stripboard

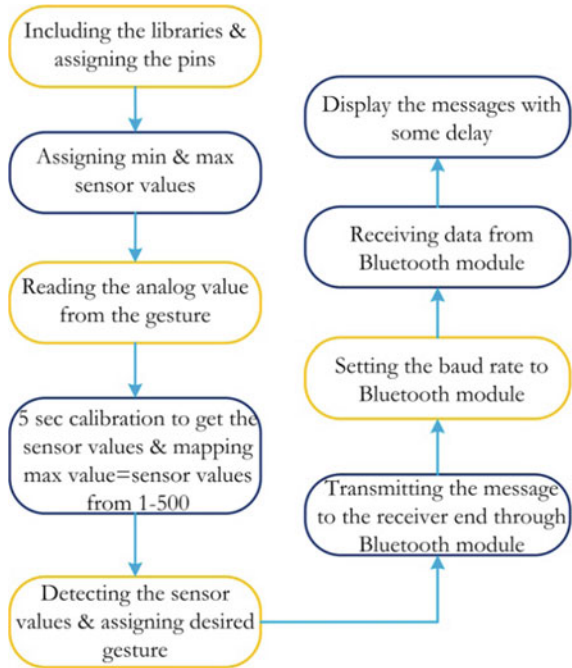
Fig. 4 Output view of Arduino Nano with liquid crystal display



4 Results and Discussion

To detect finger movements, flex sensors were used in the design. It is made up of five sensors that have been arranged in a hand glove to make them more comfortable

Fig. 5 Software implementation flowchart



to use. When bending, the resistance value of the flex sensors changes. A constant-value resistor connects one side of the voltage divider to the other. The Arduino detects the voltage difference as the sensors bend and orders the servos to move proportionally. On the robotic hand’s side, a 12 V external power supply is utilized. The gesture-controlled speech assistance device is successfully built. The necessary hardware connections to the device are checked and tested. The Arduino Nano both on input and output side is configured in their platforms to achieve desired task of gesture recognition and output display, respectively. All the flex sensors are tested and mounted on the glove using synthetic glue. The LCD display is checked and tested with required connection to the microcontroller. The RF transmitter/receiver pair is installed, and communication between input side and output side is checked. The recognition of gestures by movement of fingers is achieved as shown in Fig. 6.

The transfer of data in the form of phrases is achieved between the input and output side. The display of phrases on output side LCD is controlled by input side glove gestures. At output side, phrases are displayed on the LCD which has a functioning backlight to facilitate reading in poor lighting conditions shown in Fig. 7. The key advantages of the device are: It is wireless, portable, light in weight, easy to handle, multilingual.

Fig. 6 Some of the gestures are shown below along with their corresponding phrases **a** HI HOW ARE YOU? **b** PEACE, **c** WHAT? **d** SUPER, **e** SHOULD DO SOMETHING ABOUT IT

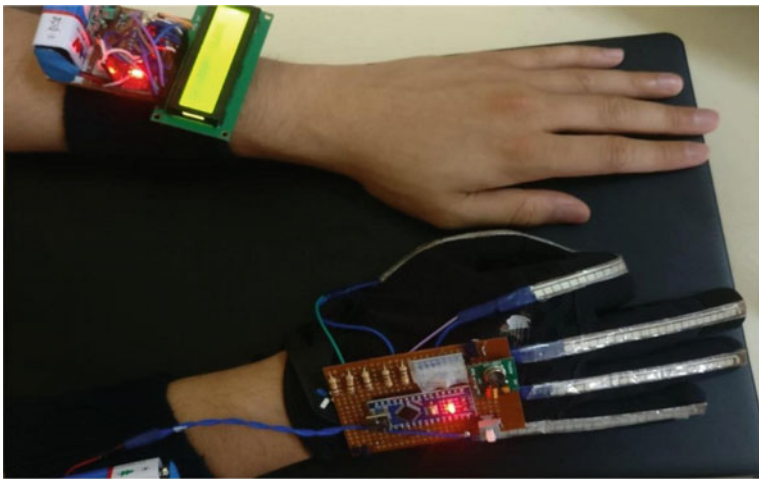
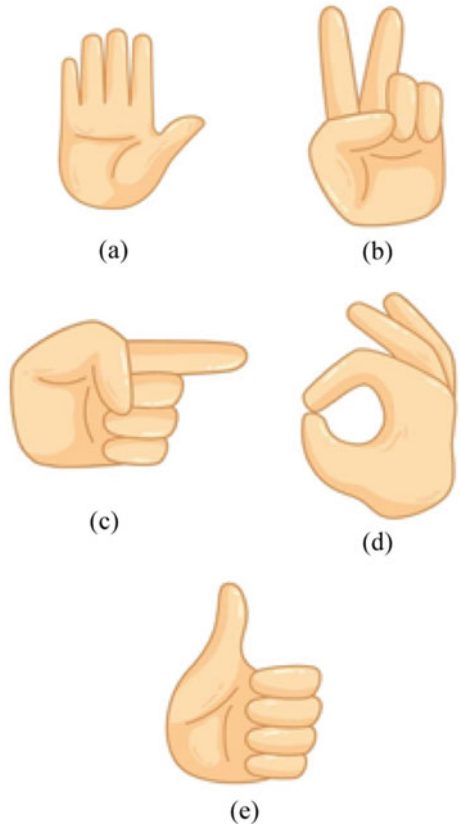


Fig. 7 Gesture-controlled speech assistance device for the verbally disabled

5 Conclusion

For the vocally disabled, the gesture-controlled voice aid technology provides a first step into communication. This device could be part of a larger network of devices that collaborate to break down communication barriers. We can improve the accuracy of gesture detection while also expanding the number of gestures that can be stored by adding more sensors such as the accelerometer and gyroscope. This glove can also be used in conjunction with speakers to create a computerized assistant that acts as a virtual voice and reads out messages from the user. The use of such gloves is currently limited, but various efforts are in the works to make them more economical and simple to use. With modern technologies such as artificial intelligence and machine learning entering the scene, incorporating these into a speech aid device opens up a world of possibilities. To put it another way, the glove can turn into a complete 360° helper for the user by connecting to the Internet and executing various IOT-based operations, such as turning on and off electronic gadgets in a smart home using gestures. To expand its capabilities, this glove can be linked to a smartphone via apps. Brain mapping and artificial intelligence are two other technologies being developed now that will have a significant impact. The ultimate objective of this technology will be achieved when disabled persons may fully communicate with the rest of the world by using such gadgets to entirely overcome their disabilities. This would also provide disabled people access to a wider range of job opportunities that they may not have access to now. A highly precise, cost-effective, and self-contained glove for deaf and dumb persons were devised to serve as a communication bridge. Their sign language gestures can be converted into speech using the glove. Smart Glove is a gesture-to-phrase translator.

References

1. Oo HM, Tun KT, Thant ML (2019) Deaf sign language using automatic hand gesture robot based on microcontroller system. *Int J Trend Sci Res Dev* 3:2132–2136
2. Telluri P, Manam S, Somarouthu S, Oli JM, Ramesh C (2020, July) Low cost flex powered gesture detection system and its applications. In: 2020 Second international conference on inventive research in computing applications (ICIRCA), IEEE, pp 1128–1131
3. Nagpal A, Singha K, Gouri R, Noor A, Bagwari A (2020, September) Hand sign translation to audio message and text message: a device. In: 2020 12th International conference on computational intelligence and communication networks (CICN), IEEE, pp 243–245
4. Flores MBH, Siloy CMB, Oppus C, Agustin L (2014, November) User-oriented finger-gesture glove controller with hand movement virtualization using flex sensors and a digital accelerometer. In: 2014 International conference on humanoid, nanotechnology, information technology, communication and control, environment and management (HNICEM), IEEE, pp 1–4
5. Dhepekar P, Adhav YG (2016, September) Wireless robotic hand for remote operations using flex sensor. In: 2016 International conference on automatic control and dynamic optimization techniques (ICACDOT), IEEE, pp 114–118
6. Padmanabhan V, Sornalatha M (2014) Hand gesture recognition and voice conversion system for dumb people. *Int J Sci Eng Res* 5(5):427

7. Manikandan K, Patidar A, Walia P, Roy AB (2018) Hand gesture detection and conversion to speech and text. arXiv preprint [arXiv:1811.11997](https://arxiv.org/abs/1811.11997)
8. Yamunarani T, Kanimozhi G (2018) Hand gesture recognition system for disabled people using arduino. *Int J Adv Res Innovative Ideas Educ (IJARIIE)* 4(2):3894–3900
9. Nagpal A, Singha K, Gouri R, Noor A, Bagwari A, Qamarra S (2020, October) Helping hand device for speech impaired people. In: 2020 Global conference on wireless and optical technologies (GCWOT), IEEE, pp 1–4
10. Poornima N, Yaji A, Achuth M, Dsilva AM, Chethana SR (2021, May) Review on text and speech conversion techniques based on hand gesture. In: 2021 5th International conference on intelligent computing and control systems (ICICCS), IEEE, pp 1682–1689
11. Speak H (2014) Why is communication important to human life. Retrieved From
12. Sunitha KA, Saraswathi PA, Aarthi M, Jayapriya K, Lingam S (2016) Deaf mute communication interpreter-a review. *Int J Appl Eng Res* 11:290–296
13. Sturman DJ, Zeltzer D (1994) A survey of glove-based input. *IEEE Comput Graphics Appl* 14(1):30–39
14. Kunjumon J, Megalingam RK (2019, November) Hand gesture recognition system for translating indian sign language into text and speech. In: 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), IEEE, pp 14–18
15. Verdadero MS, Cruz JCD (2019) An assistive hand glove for hearing and speech impaired persons. In: 2019 IEEE 11th International conference on humanoid, nanotechnology, information technology, communication and control, environment, and management (HNICEM), IEEE, pp 1–6
16. <https://www.dreamstime.com/illustration/american-sign-language.html>

Highly Classified with Two Factor Authentication Encrypted Secured Mail



N. H. Prasad, B. N. Lakshmi Narayan, and S. GovindaKishora

1 Introduction

Nowadays, security is that the most concerning aspect in any IT-related transaction. Considering this thing in mind, we studied the present system of “mail server” services. One in all the most drawbacks of the present system is that it is only one level or unary international intelligence agency, i.e., specific username and password which give mail limited security only. The matter is solved by giving them extra security that is by giving secondary to the mail. Here, we have also applied the concept of priority queuing of the mail where the inbox may be sorted within the user-specified format consistent with the users given priority like date, alphabetically Email ID, and secured mail [1]. The objective of our project is to produce second-level security to the present email system. This makes use of the secondary key to every mail which is vital. Each mail that is sent by the sender is going to be having a key of its own which can be sent to the transportable of the receiver. We have got also tried to form the email system a reliable and user-friendly communication mode [2].

2 Literature Review

Email, every so often referred to as e mail, can be a PC-based totally method of sending messages from one a person to a exclusive. These messages usually carry person portions of textual content which you will be capable of ship to a one-of-a-kind man or woman whether or now not the opposite consumer is not always logged in (i.e.,

N. H. Prasad · B. N. L. Narayan · S. GovindaKishora (✉)

Department of Masters of Computer Applications, Nitte Meenakshi Institute of Technology, Bengaluru, India

e-mail: govindakishor3@gmail.com

the use of the computer) on the time you send your message. This system is similar to sending and receiving a letter. At the start, email messages have been constrained to easy text, but now, many structures can cope with more complicated formats, like graphics and phrase-processed files. While mail is obtained on a ADPS, it is normally saved in an electronic mailbox for the recipient to examine later. Electronic mailboxes are usually special files on a computer that can be accessed using diverse commands. Each person normally has a non-public mailbox.

3 Proposed System

Machine reminiscence like RAM shops all of the active logs, disk buffers, and associated records. Moreover, it shops all the transactions which might be being presently done. If such keep crashes unexpectedly, it would get rid of all the logs and active copies of the database. It makes recuperation almost impossible, as the entirety, it really is required to get better the info is lost. Following strategies can also be adopted just in case of lack of computer storage [3]—we will have checkpoints at more than one ranges to keep plenty of the contents of the database periodically. A nation of the lively database within the risky reminiscence might be periodically dumped onto strong garage, which can also contain logs and active transactions and buffer blocks. While the machine recovers from a failure, it can repair the most recent sell off. It could maintain a re do-listing and an undo-listing as checkpoints. It could recover the device by using consulting undo-redo lists to revive the kingdom of all transactions up to the closing checkpoint [4].

In all mail system, they follow the procedure to test the mail using the existing security tool before sending to mail server. The email header will be taken and same will be matched with label in security tool, if it matched further process continues, if it not the mail will reported to higher authority to take further action [5].

Buffers are allocated for every inbox mail, and details regarding are noted such as sender id, date, and time. If the centralized servers are attacked, the confidential message and the private information will be leaked to the attacker [6]. Buffers then search for packet ids, if the details stored at storage garage matches the PID, then it will be allowed to read.

Due to the untrustworthy nodes participating in the blockchain network, the blockchain-based system needs to adopt a suitable consensus algorithm to achieve the consistency of the decentralized distributed databases with the node competition [11].

4 Architecture

See (Fig. 1).

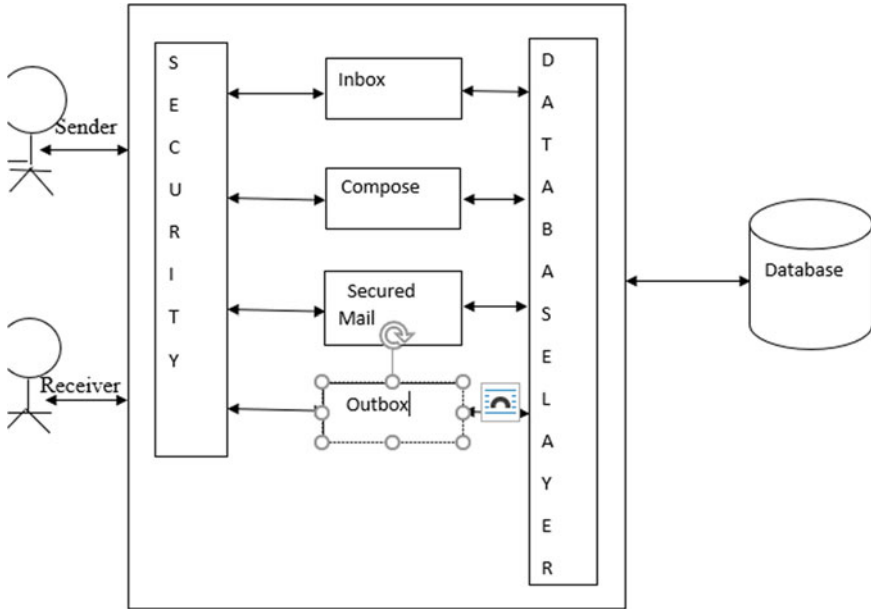


Fig. 1 System architecture

5 Methodology

Secured mail is that the email system that is developed to present higher security for confidential emails. The user styles of this mail system are

- The sender
- The receiver

To provide an affordable level of privacy, all the routes within the email pathway and every one connections between them must be secured. Here, the second level of mail security means that we are encrypting the mail employing a unique key as a method of security within the second stage to shield the mail contents whether or not the user loses ownership of his/her account [7]. Here, we are implementing the secured mail system which provides higher security to the mails. The sender sends the confidential mail, and it is received by the receiver when the receiver clicks to open the closed mail he are asked for the key, at the identical time, he gets a message with the OTP, this key needs to be entered and checks for validity, then the receiver can easily read the mail. This secrets unique and it gets a refresh [8] (Table 1).

Interface uses HTTP and GET method to send a account ID and encrypted password to server. Login pages make use of HTTP where hacker can obtain password by eavesdropping in the network. Even though login page 163 uses HTTP, but the account credentials are all sent through secure socket layer to server, in this stage, hacker can identify the account ID followed by password [9].

Table 1 Encrypted password in different email systems

Name	HTTP/HTTPS	POST/GET	Password is encrypted?
Gmail	HTTPS	POST	Yes
Yahoo	HTTPS	POST	Yes
Hotmail	HTTPS	POST	Yes
Sohu: interface(1)	HTTP	GET	Yes
Sohu: interface(2)	HTTPS	GET	No (hashed)
Sohu: interface(3)	HTTPS	POST	Yes
163	HTTP	POST	Yes

The details are taken from two sources, such as email header and content. Header contains account ID and password with IP address, and content contains packet-related information [10].

6 Results

See Figs. 2, 3, 4 and 5.

7 Conclusion

To provide an affordable privacy, all routers within the electronic mail pathway and connection between them have to be secured. Here, the title “secured mail system” means that we are encrypting mail using OTP as a method of security within the second stage, to safeguard the mail contents whether or not the user loses ownership of his/her account.

Here, we have implemented the secured mail system which supplies higher security to mails. The sender sends the confidential mails, and it is received by the user when the receiver clicks to open the mail, then it will elicit the OTP, at the identical time, he gets a message with the OTP. This OTP must be entered and checks for validity, then the receiver can easily read the mail. This secrets unique and it gets regenerated.

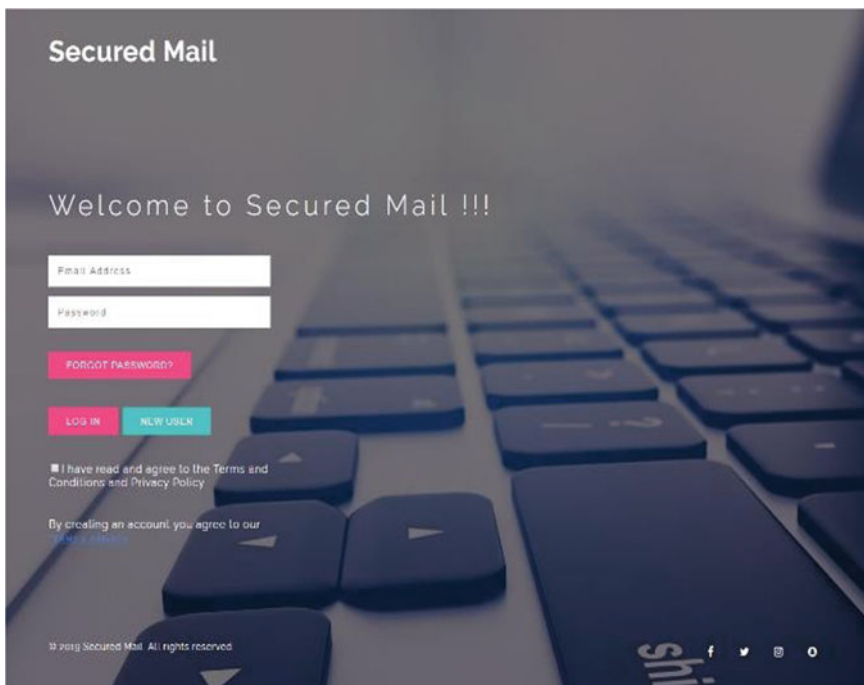


Fig. 2 Login page

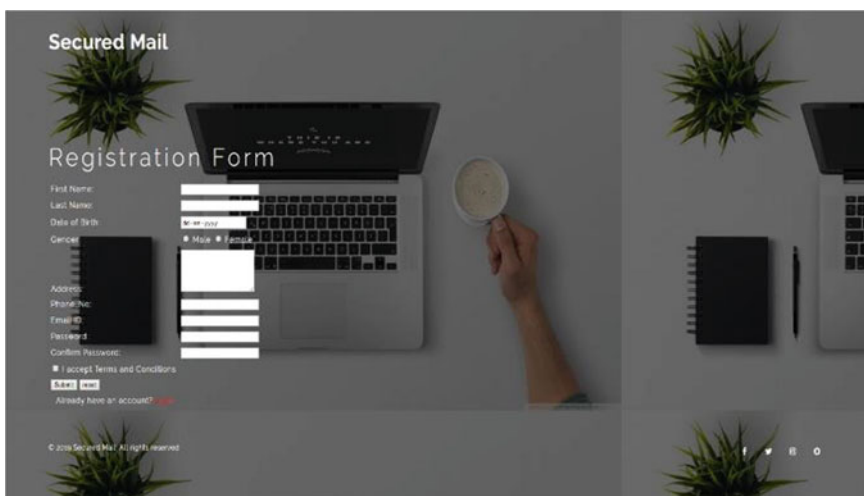


Fig. 3 Registration page

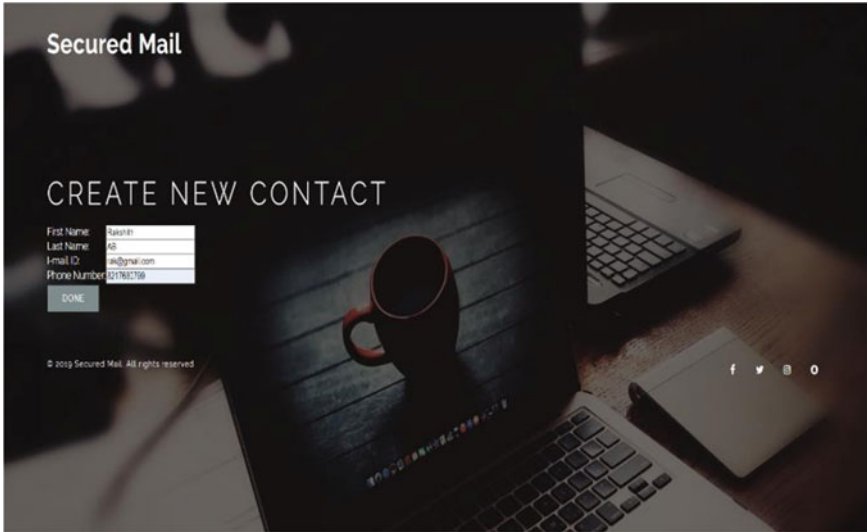


Fig. 4 Contact page

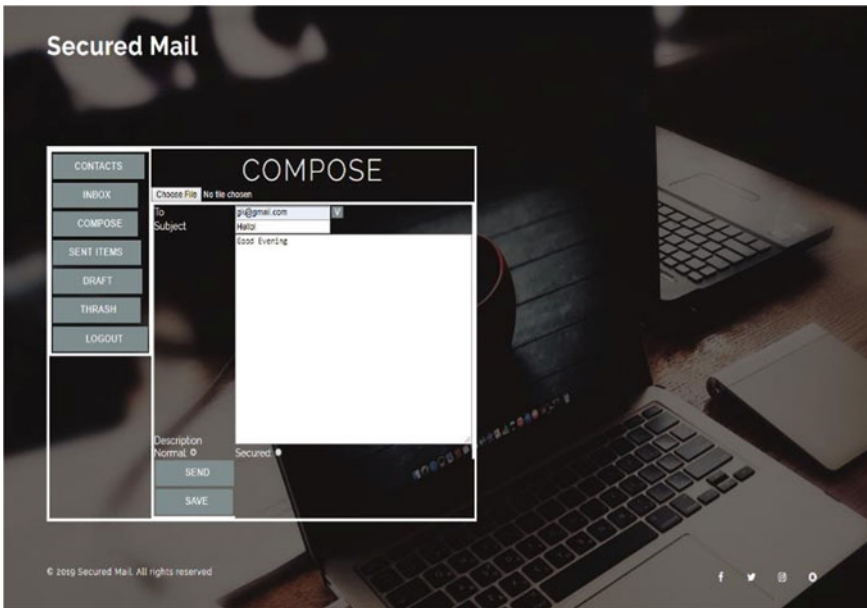


Fig. 5 Compose mail

8 Future Enhancements

- Development can be made, to add videos or send videos
- Provide cloud storage facility
- Hear songs by giving a mail Saavan facility

References

1. Benarous L, Kadri B, Bouridane A (2017) A survey on cyber security evolution and threats: biometric authentication solutions. In: Biometric security and privacy, Springer, Berlin, Germany, pp 371–411
2. Boyd C, Mathuria A (2013) Protocols for Authentication and Key Establishment. Springer, Berlin, Germany
3. Mohsin J, Han L, Hammoudeh M, Hegarty R (2017) Two factor versus multi-factor, an authentication battle in mobile cloud computing environments. In: Proceedings of the international conference on future networks and distributed systems, Cambridge, UK, 19–20 July 2017; ACM, New York, NY, USA, p 39
4. Pathan ASK (2016) Security of Self-Organizing Networks: MANET, WSN, WMN, VANET; CRC Press: Boca Raton. FL, USA
5. Borran F, Schiper A (2010) A leader-free byzantine consensus algorithm. In: International conference on distributed computing and networking, Springer
6. Li T, Mehta A, Yang P (2017) Security analysis of email systems. In: 2017 IEEE 4th International conference on cyber security and cloud computing (CSCloud). IEEE Xplore: 24 July 2017. <https://doi.org/10.1109/CSCloud.2017.20>
7. Balloon AM (2001) From wax seals to hypertext: electronic signatures, contract formation, and a new model for consumer protection in internet transactions. Emory Law J 50:905
8. Danny T (2017) MFA (Multi-Factor Authentication) with Biometrics. Available online: <https://www.bayometric.com/mfa-multi-factor-authentication-biometrics/>. Accessed online 4 Jan 2018
9. Huang JW, Chiang, CW, Chang JW (2018) Email security level classification of imbalanced data using artificial neural network: the real case in a world-leading enterprise. In: Engineering applications of artificial intelligence, vol 75. October 2018
10. Bao X (2020) A decentralized secure mailbox system based on blockchain. In: 2020 International conference on computer communication and network security (CCNS) IEEE Xplore: 02 Nov 2020. <https://doi.org/10.1109/CCNS50731.2020.00038>

Efficacious Intrusion Detection on Cloud Using Improved BES and HYBRID SKINET-EKNN



C. U. Om Kumar , Ponsy R. K. Sathia Bhaman , and Prasad

1 Introduction

Though various approaches have been proposed for solving the challenges of DDoS attack, the best solution in terms of crucial paradigms like false positive ratio (FPR) and detection rate (DR) with utmost accuracy and precision is yet to come by. In this study, it is proposed to detect and curtail the propagation of DDoS attacks in a cloud environment by enhanced IDS. It consist of an EPIA and IBES which are two novel additions that strengthen the existing IDS. A hybrid classifier is also introduced for the classification of malicious attacks.

The use of cooperative IDS not only helps in the accurate detection of cloud intrusion attacks but also significantly increases the security of the system. Self-explanatory as it is, Intrusion detection system detects anomalies in the network traffic. On account of this functional attribute IDS has become a crucial component of security information system model meant to protect the Internet of Things (IoT) network from cyber-attacks.

Until recently, the single intrusion detection system was considered effective as the flow of traffic was limited. But with emerging developments in IoT happening

C. U. O. Kumar (✉)

School of Computer Science & Engineering, Vellore Institute of Technology - Chennai Campus, Chennai 600127, India

e-mail: omkumar.cu@vit.ac.in

P. R. K. S. Bhaman

Department of Computer Technology, MIT Campus-Anna University, Chennai 600044, India

e-mail: Indiaponsy@mitindia.edu

Prasad

Department of Computer Applications, NITTE Meenakshi Institute of Technology, Bangalore 560064, India

in quicker strides, the nature of attacks is also getting complex. In these circumstances, single IDS is considered incapable of arresting such advanced attacks owing to constraints like incomplete knowledge of the implications of such attack patterns. A new approach using IDS was suggested by [1] that helped in selecting IoT data features. Before the feature selection, the lowest confidence level packets are eliminated through confidence level of EPIA algorithm and the flow properties are inspected through Shannon property. Then, optimal features are obtained from the filtered packets using IBES optimization. Finally, classification is performed by HSKiNET-EKNN.

Many innovative approaches such as swarm intelligence, data mining and machine learning have been introduced to prevent the network from different attacks. Author's in [2] used beta mixture model (BMM) for training the classifier using raw dataset to detect attacks. Signature-based detection method was done using individual machine learning approaches by adopting K -nearest neighbour (KNN). The objectives of the study.

1. To strengthen the existing packet inspection algorithm by integrating it with Shannon Entropy for the inspection of Flow properties.
2. To optimize the feature selection process by adding the maiden improved bald eagle search algorithm for the maiden time.
3. To achieve a better rate of accuracy by adding a maiden hybrid classifier algorithm called HSKiNET: EKNN for the maiden time.

2 Review of Literature

An intrusion detection system is a software that analyses the network traffic for malicious activity. The wide range of IDS are further classified as NIDS, HIDS, PIDS, APIDS and HIDS. Anomaly-based and signature-based detection are the techniques most commonly used for detection in all variants of IDS.

Ficco et al. [3] has deployed hierarchical IDS in which the obtained data is transmitted to the security engines over the physical server where the data correlation is performed to determine the distributed attacks. A collaborative IDS framework was proposed by [4]. The unknown attacks were detected using a fusion of Decision Tree classifier and SVM and the known attacks are determined by Snort. For collecting information regarding the attacks from VM at both virtualization and network levels, the VMs were monitored using the proposed technique of [5]. The malicious network packet detection model handles high network traffic at the hypervisor level. The drawback of this work was that memory introspection was not performed in VMM necessary for the detailed investigation of attacks.

Arjunan and Modi in [6] predicted the types of attacks by combining both signature and anomaly techniques. This work was a maiden venture for detecting both known and unknown attacks. By means of alert correlation, the distributed attack is detected using the correlation module existing in the centralized server. Many machine learning techniques like ET, RF, DT and naive Bayes were used to improve

the accuracy rate and reduce the FPR. Alerts from each classifier are fed into Dempster–Shafer theory for further improving detection accuracy. The IDS evaluation has maximum encouraging outcomes, but many detection defects like maximum FPR for unbalanced samples and minimum detection rate for unknown attacks are yet to be solved.

Authors in [7] proposed anomaly-based intrusion detection system for cloud providers. The conventional network based IDS inspects the traffic by firing rules through their detection engine. This work deploys IDS sensors across VMs and hypervisor that help in improving the Detection Accuracy in the cloud environment. The limitation of this work is that the aforementioned sensor deployment is tested offline which avoids many real time challenges faced in real cloud testbed through VM, Containers and Dockers. Anomaly detection by Homayoun et al. [8] developed BotShark through Stacked Autoencoders which extracts relevant features from a huge enormous input. The reduced subset of features is passed to softmax classifier to generate probabilities for detecting the most probable malicious and benign traffic. Though this technique achieved a FPR of 0.13 it requires a pre-training phase to understand the actual profile and network features.

Patil et al. in [9] devised protocol specific multi-threaded network intrusion detection system (PM-NIDS) for the cloud environment. The model captures DoS/DDoS attacks by sniffing packet protocols. They segregate the packets into different queues based on the protocol so that they can be handled in parallel thereby reducing packet loss. Protocol specific classifiers like DT, RF and ONER were used for generating alerts. Better performance was achieved, but still the handling of different protocols remains a critical issue. Li et al. [10] devised an IDS model that detects malicious traffic using a RNN-RBM hybrid model. From the encountered traffic RBM retrieves features to build a feature vector whereas the RNN identifies Flow features. The identified Flow features are then fed to softmax classifier to find the probable ones.

Chergui and Bousti [11] designed a Non-monotonic Ontological Contextual-based strategy to minimize the FPRs of IDS. They address issues like raising false alerts, vulnerability and network updation. In this context if the system handles the issue of vulnerability, then the exploited sequence is categorized neither as true alert nor false alert. The class-imbalance issue or improving IDS performance was carried out by [12]. Cost-sensitive stacked autoencoder (CSSAE) generates costs for each class. Furthermore, features are learnt from minority and majority classes based on their derived cost values. The cost matrix is built by adjusting the neurons' corresponding cost through the cost function.

3 Efficacious Intrusion Detection System

A cloud's function is to collect and store packets from distant cloud users through routers in cloudlets. In this process, it becomes vulnerable to attacks from botnets. The CC in system model displayed in Fig. 1 observes all files on cloudlets from clients and produces a threshold level for each cloudlet so as to avoid imbalance in

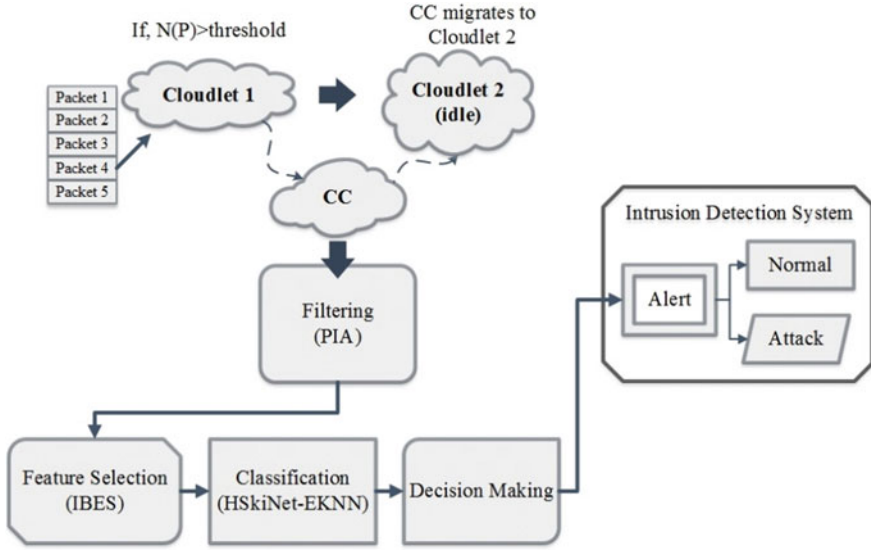


Fig. 1 Overview of the proposed system model

packet’s traffic. It transfers the files to idle cloudlets when there is a surge in traffic flow. In this study, two novel additions, namely EPIA and IBES are introduced to streamline the inspection of packet flow traffic through Feature Selection.

3.1 Enhanced Packet Inspection Algorithm

In this model, the PIA is enhanced using Shannon entropy method, where packets flow and arrival time are analysed using confidence level and then the packet flow properties are analysed using Shannon entropy method. At first, every packet is analysed according to the packet flow and arrival time. It foresees a surge in flow based on features like packet count, arrival time and checks the confidence level.

The confidence level (CL) is evaluated according to single and pair attributes Balamurugan and Saravanan in [13].

(i) Single Attribute’s Confidence

$$CL(X_i = X_{i,j}) = \frac{N(X_i = X_{i,j})}{N_n} \quad i = 1, 2, \dots, n \quad \text{and} \quad j = 1, 2, 3 \dots m \tag{1}$$

(ii) Pair Attribute’s Confidence

$$CL = (X_{i1} = x_{i1,j1}, X_{i2} = x_{i2,j2}) = \frac{N(X_{i1} = x_{i1,j1}, X_{i2} = x_{i2,j2})}{N_n} \quad (2)$$

Then, every packet's confidence level is evaluated; if it is low, then the packet is eliminated, or else it is admitted. Then, the packets are checked for flow properties. Feature-based IDS has been introduced not only for controlling traffic volume but also to explore the network traffic's flow properties. General Internet Protocol (IP) attributes such as port numbers, Source and Destination IP address are utilised for this purpose. The Shannon entropy is applied in the context of Intrusion Detection. For instance, assume the Probability distribution as

$$P = \langle p_1, p_2, \dots, p_n \rangle \text{ using } 0 \leq p_i \leq 1$$

where $\sum_{i=1}^n p_i = 1$, the entropy is defined as,

$$E_s = - \sum_{i=1}^n p_i \log p_i \quad (3)$$

The relationship between Shannon entropy and probability is checked. If the entropy value is close to zero, it is denoted as low probability; if it is close to one, it is represented as high probability having same attributions as entropy result.

After checking the arrival time, the algorithm is initiated and based on the arrival time, the packet flow is classified. Finally, the confidence level for every packet is checked, and based on this, the packets are either accepted or eliminated. Using Shannon entropy method, the IP flow properties are investigated and based on probability, the number of packets are reduced drastically. Then the filtered packets undergo feature selection procedure for obtaining optimal features. The feature selection procedure is processed through IBES optimization which is explained in the following section.

3.2 Improved Bald Eagle Search

The structure of basic bald eagle search is improved to increase convergence speed, reliability and solution accuracy. A new parameter, opposite-based learning (OBL) is added for enhancing the efficiency. The new method known as IBES is tested in feature selection approach. The population diversity of BES in the search space is improved by OBL. In IBES, the algorithm follows an effective fitness function to eliminate irrelevant and redundant features.

The collection of feature subsets FS is expressed as:

$$FS = (FS_1, FS_2, \dots, FS_{2^n-1}) \quad n = \text{number of features} \quad (4)$$

The number of subsets is exhaustive. Various strategies are developed for getting an enumerative solution. The Metaheuristic IBES algorithm is incorporated in the wrapper based approach, which is a maiden venture used in this study for enhancing the Intrusion Detection System and has not been implemented so far. BES is a strategy originally proposed by Alsattar et al. in [14] which works on the assumption of Eagle's strategy of searching and obtaining prey using optimal hunting decision. IBES an enhanced approach of BES adopts the OBL function for enhancing the population diversity and makes use of an effective fitness function for the removal of redundant and irrelevant features.

An improved version of BES algorithm is adopted for feature selection as the conventional BES algorithm lags in convergence speed. Opposition-based learning (OBL) proposed by Tubishat et al. in [15] is adopted as it improves BES's ability for generating solutions. In addition, an effective fitness function is used for removal of redundant and irrelevant features. After the BES algorithm has generated its initial population, the OBL approach finds each of the initial solution's opposite solution.

3.2.1 Opposite Based Population

Actual interval $\gamma \in [lb,ub]$ is defined as γ , where lb and ub are considered as the lower bound and upper bound values correspondingly in the present j dimension. Here \forall is the opposite number of γ evaluated using the equation

$$\forall = lb + ub - \gamma \quad (5)$$

Then, the features are compared with the fitness function in each stage based on the objective of maximizing accuracy, detection rate and minimizing false positive rate for getting the best feature. Thus, based on the fitness, OBL takes the best n solution from the set of initial and opposite solutions. Based on the classification accuracy, the IBES calculates the fitness value for each possible solution. Furthermore, while testing the dataset used by IBES to evaluate the SKiNET classification accuracy, the training dataset is used to train the SKiNET classifier. Thus, IBES chooses the best solution X^* from the selected solutions.

$$\mathbf{FitnessFunction} : \text{fitness}_{\text{IBES}}(\forall) = DR + [1 - FAR] + F \left[1 - \frac{\sum_{i=1}^N F_i}{N} \right] \quad (6)$$

3.3 Hybrid Classification Process

The network system consists of an input, a hidden and an output layer. In the event of receiving an external input information the active neurons will rush to respond to the input. Ultimately, the winning neurons will be excited to display the classification results while the loosing neurons will remain inactive. The SKiNET proposed by Banbury et al. in [16] calculates the minimum distance probability based on the Euclidean distance and classifies using maximum probability method. EKNN predicts the appropriate feature class label by using kernel similarity function. In this study, a hybrid classification algorithm is obtained by integrating SKiNET-EKNN. Here, VMM executes the hybrid classification algorithm by which several DDoS attacks are detected, especially in an IoT cloud environment. The conventional KNN algorithm follows Euclidean distance to find out the similarity amongst the neighbours. Here to enhance the KNN performance, Kernel-Based Similarity Measure is introduced instead of Euclidean distance measure. Since Euclidean distance cannot classify the linearly inseparable complex datasets, kernel function is adopted for plotting both data points as high-dimensional feature space to create the data at ease and find the closed one. Initially, in SKiNET, the feature values are normalized, and the weight value is randomly assigned. Then, Euclidean distance is calculated for every feature and weight. Finally, minimum distance value is determined using the maximum probability method.

3.4 Enhanced KNN: Kernel-Based Similarity Measure

The testing results of distance values are compared with SKiNET value, here K value is fixed and based on the K value all the distance values are ranked. From the ranked results, majority vote result is taken as the prediction of classification. Since it is challenging to cluster Euclidean distances on linearly inseparable, complex datasets, a kernel function is followed to create high-dimensional feature space by mapping two data points for easing the clustering of data. $S = [\vec{u}_i]$ $i = 1 to N$ represents the dataset wherein $[\rightarrow_{u_i}] \in \mathbb{R}^d$, is a nonlinear kernel function which is used for mapping the raw data obtained from the input space \mathbb{R}^d to the high Dimensional Feature space H , as shown below:

$$\varphi : \mathbb{R}^2 \rightarrow \Omega, \rightarrow_{u_i} \varphi(\rightarrow_{u_i}) \quad (7)$$

Hereafter, the following equations denote the kernel distance between two data points, $[\rightarrow_{u_i}]$ and wherein $[\rightarrow_{u_j}]$

$$\|\varphi(\rightarrow_{u_i}) - \varphi(\rightarrow_{u_j})\|^2 = ((\varphi(\rightarrow_{u_i}) - \varphi(\rightarrow_{u_j}))(\varphi(\rightarrow_{u_i}) - \varphi(\rightarrow_{u_j}))) \quad (8)$$

$$\varphi^T(\rightarrow_{u_i}) \cdot \varphi(\rightarrow_{u_j}) - 2\varphi^T(\rightarrow_{u_i}) \cdot \varphi(\rightarrow_{u_j}) + \varphi^T(\rightarrow_{u_i}) \cdot \varphi(\rightarrow_{u_j}) \quad (9)$$

$$= K(\rightarrow_{u_i}, \rightarrow_{u_j}) - 2K(\rightarrow_{u_i}, \rightarrow_{u_j}) + K(\rightarrow_{u_i}, \rightarrow_{u_j}) \quad (10)$$

The Gaussian kernel is expressed as shown in the equation:

$$K(\rightarrow_{u_i}, \rightarrow_{u_j}) = \exp\left(\frac{\|\rightarrow_{u_i} - \rightarrow_{u_j}\|^2}{2\sigma^2}\right), \quad \sigma > 0 \quad (11)$$

The distance between two data points are evaluated thus:

$$d_{i,j} = \|\varphi(\rightarrow_{u_i}) - \varphi(\rightarrow_{u_j})\| = \sqrt{2(1 - K(\rightarrow_{u_i}, \rightarrow_{u_j}))} \quad (12)$$

In the hybrid SKiNET-EKNN, the Euclidean distance of SKiNET is replaced by the EKNN. Here, the kernel-based similarity is used instead of Euclidean distance, and the subsequent steps are processed based on the SKiNET. Finally, the predicted classification output is obtained and displayed. Equation (12) is substituted in Eq. (13) as shown below.

$$d_{i,j} = 2\sqrt{2(1 - K(\rightarrow_{u_i}, \rightarrow_{u_j}))} \quad i = 1, 2, 3, \dots, N \quad j = 1, 2, 3, \dots, 100 \quad (13)$$

The proposed classification approach contributes to enhancing the accuracy of hybrid SKiNET-EKNN thus making it more efficacious.

4 Experimental Analysis

The performance of the proposed system is evaluated by implementation in JAVA through cloudsim platform. The simulation and evaluation over cloudSim are benchmarked with standardized well-known datasets such as NSL-KDD by [17], UNSW-NB15 by [18] and N-BaIoT by [19].

4.1 NSL-KDD 99 Dataset

NSL-KDD is an extended version of the original KDD cup 99 dataset. This dataset consists of redundant data that causes imbalance in learning leading to biased results (Table 1).

Thus, in conclusion, the performance of IBES strategy approach in accuracy is better with a peaking value of 99.5%. In terms of the next parameter, FPR also the

Table 1 Overall performance of NSL-KDD 99 dataset

Sl. No.	Approach type	ACC (%)	DR (%)	FPR (%)
1	IBES-HSKiNET_EKNN[proposed]	99.5	99.12	0.3
2	Karami in [20]	98.3	97.05	0.87
3	Gu et al. in [21]	99.41	99.09	0.31
4	Sahar et al. in [22]	99.38	99.38	0.7
5	Gowrison et al. in [23]	99.34	99.27	0.59

current proposed approach peaks again with 0.3%. Considering DR also the current approach comes next to Muhammed et al. with a rate of 99.12% which is a mere 0.26% less than the highest performance of 99.38% recorded by Muhammed et al.

4.2 UNSW-NB 15 Dataset

The UNSW-NB15 dataset comprises synthetic attack characteristics and replicates real-time events. The Tcpdump tool, Bro-IDS tool and Argus were used to create 25,400,443 raw network packets that generated 49 features with class labels. Table 2 juxtaposes the performance efficiency of the proposed approach against that of existing approaches.

As an overall inference, it is seen that the IBES strategy adopted in the study has registered the highest ACC rate of 99.6%. Taking into consideration, the performance of second parameter, namely DR, here again the IBES strategy has yielded a performance rate of 99.02%. In terms of the third and final parameter, i.e., FPR too, the IBES approach has topped all other approaches recording the least FPR of 1%. In a nutshell, the IBES approach has emerged a clear winner by topping in all the three parameters.

Table 2 Overall performance of UNSW-NB15 dataset

Sl. No.	Approach type	ACC (%)	DR (%)	FPR (%)
1	IBES- HSKiNET_EKNN[proposed]	99.60	99.02	1
2	Karami in [20]	92.80	91.30	5.1
3	Yang et al. in [17]	90.20	96.22	17.15
4	Kumar et al. in [18]	84.83	90.32	2.01
5	Muna et al. in [24]	99.54	98.93	1.38
6	Patil et al. in [9]	97.09	96.70	2.03

Table 3 Categorization of attacks

Category of attack	ACC (%)	DR (%)	FPR (%)
Normal	99.91	100	0.9
Combo	99.82	99.9	0.81
Junk	99.4	99.7	2.41
TCP	99.91	99.9	0.39
UDP	99.61	99.9	3.2
SCAN_MIRAI	99.73	99.8	0.81
SYN	99.57	99.71	1.62
UDP_Mirai	99.82	99.9	0.81
UDP_Plain	99.56	99.7	1.62

Table 4 Overall performance of datasets

Dataset used	ACC (%)	DR (%)	FPR (%)
NSL KDD-99	99.5	99.12	0.3
UNSW NB15	99.6	99.02	1
N-BaIoT	99.7	99.8	0.2

4.3 NBaIoT Dataset

BASHLITE and Mirai are two of the foremost general IoT botnet dataset families. This is the new IoT Botnet dataset that is valued as a benchmark dataset for the proposed IDS. Besides the three aspects, namely ACC, DR and FPR. Table 3 presents the performance of each intrusion category in terms of accuracy, detection rate and false positive rate for N-BaIoT dataset.

In summation from Table 4, the current approach using N-BaIoT dataset has outperformed the other two. It is observed that the IBES optimization approach has yielded a better performance. The use of EPIA for filtering packets on the basis of confidence level and Shannon entropy for achieving a reduction or easing congestion in traffic flow is worthy of implementation; The use of IBES optimization algorithm for selecting optimal features and the adoption of HSkNET_EKNN hybrid classification have all contributed for achieving better outcomes. This stands as a testimony for the validity of the proposed improved bald eagle search approach in the substantial detection and confrontation of flash attack in a cloud environment.

5 Conclusion

This study proposes the adoption of a new IDS for effectively predicting the malware attacks in the VM. Initially, EPIA filtered the low-risk malicious packets and it used Shannon entropy for identifying the packet flow properties. Then, the filtered packets

are sent for selection of optimal features. The OBL with added fitness function enhanced the algorithm and improved the convergence time yielding commendable results. Finally, the accuracy in classification has been improved by HSKiNET-EKNN. Here, SKiNET takes the minimum distance probability value and predicts the results using EKNN. Also, the KNN is enhanced by kernel-based similarity function. The simulated performance of proposed approach has been tested on three benchmarked datasets such as NSL-KDD, UNSW-NB15 and N-BaIoT. The proposed model has achieved a maximum ACC of 99.7%, DR of 99.80% and low FPR of 0.2% pointing out that the enhanced IDS as a detection and confrontation strategy for curbing DDoS attack in a cloud environment is efficacious.

References

1. Li D, Deng L, Lee M, Wang H (2019) IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning. *Int J Inf Manage* 49:533–545
2. Moustafa N, Creech G, Slay J (2018) Anomaly detection system using beta mixture models and outlier detection. *Progress in Computing, Springer, Analytics and Networking*, pp 125–135
3. Ficco M, Tasquier L, Aversa R (2013) Intrusion detection in cloud computing. In: P2P, parallel, grid, cloud and internet computing, pp 276–283
4. Singh DP, Borisaniya B, Modi C (2016) Collaborative ids framework for cloud. *Int J Network Secur* 18(4):699–709
5. Mishra P, Pilli ES, Varadharajan Y, Tupakula U (2017) Out-VM monitoring for malicious network packet detection in cloud. In: *ISEA asia security and privacy IEEE*, pp 1–10
6. Arjunan K, Modi CN (2017) An enhanced intrusion detection framework for securing network layer of cloud computing. In: *ISEA asia security and privacy IEEE*, pp 1–10
7. Rezvani M (2018) Assessment methodology for anomaly-based intrusion detection in cloud computing. *J AI Data Min* 6(2):387–397
8. Homayoun S, Ahmadzadeh M, Hashemi S, Dehghantanha A, Khayami R (2018) BoTShark: a deep learning approach for botnet traffic detection. In: *Cyber threat intelligence, Springer*, pp 137–153
9. Patil R, Dudeja H, Gawade S, Modi C (2018) Protocol specific multi-threaded network intrusion detection system (PM-NIDS) for DoS/DDoS attack detection in cloud. In: *2018 9th International conference on computing, communication and networking technologies IEEE*, pp 1–7
10. Li C, Wang J, Ye X (2018) Using a recurrent neural network and restricted Boltzmann machines for malicious traffic detection. *Neuro Quantology* 16(5):823–831
11. Chergui N, Boustia N (2019) Contextual-based approach to reduce false positives. *IET Inf Secur* 14(1):89–98
12. Telikani A, Gandomi AH (2019) Cost-sensitive stacked auto-encoders for intrusion detection in the Internet of Things. *Internet of Things* 1–25
13. Balamurugan V, Saravanan R (2019) Enhanced intrusion detection and prevention system on cloud environment using hybrid classification and OTS generation. *Clust Comput* 22(6):13027–13039
14. Alsattar HA, Zaidan AA, Zaidan BB (2020) Novel meta-heuristic bald eagle search optimisation algorithm. *Artif Intell Rev* 53:2237–2264
15. Tubishat M, Idris N, Shuib L, Abushariah MAM, Mirjalili S (2020) Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection. *Expert Syst Appl* 145:113122

16. Banbury C, Mason R, Styles I, Eisenstein N, Clancy M, Belli A, Logan A, Oppenheimer PG (2019) Development of the self optimising Kohonen index network (SKiNET) for Raman spectroscopy based detection of anatomical eye tissue. *Sci Rep* 9(1):1–9
17. Yang Y, Zheng K, Bin WU, Yang Y, Wang X (2020) Network intrusion detection based on supervised adversarial variational auto-encoder with regularization. *IEEE Access* 8:42169–42184
18. Kumar V, Sinha D, Das AK, Pandey SC, Goswami RT (2019) An integrated rule based intrusion detection system: analysis on-NB15 data set and the real time online dataset. *Cluster Comput* 1–22
19. Meidan Y, Bohadana M, Mathov Y, Mirsky Y, Shabtai A, Breitenbacher D, Elovici Y (2018) N-BaIoT—network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Comput* 17(3):12–22
20. Karami A (2018) An anomaly-based intrusion detection system in presence of benign outliers with visualization capabilities. *Expert Syst Appl* 108:36–60
21. Gu J, Wang L, Wang H, Wang S (2019) A novel approach to intrusion detection using SVM ensemble with feature augmentation. *Comput Secur* 86:53–62
22. Sahar NM, Sari S, Taujuddin NSAM (2020) Intrusion-detection system based on hybrid models. In: *IOP conference series: materials science and engineering*, vol. 917 (no 1), IOP Publishing, p 012059
23. Gowrison G, Ramar K, Muneeswaran K, Revathi T (2013) Minimal complexity attack classification intrusion detection system. *Appl Soft Comput* 13(2):921–927
24. Muna AH, Moustafa N, Sitnikova E (2018) Identification of malicious activities in industrial internet of things based on deep learning models. *J Inf Secur Appl*, 41: 1–11

An Amalgamated and Personalized System for the Prognosis and Detecting the Presence of Parkinson's Disease at Its Early Onset



K. Harshitha, T. R. Vinay , K. Keerti, and M. Shreya

1 Introduction

There are no single specific tests that can be used to detect Parkinson's disease.

To rule out the presence of other illnesses, various procedures such as positron emission tomography (PET) scan, brain ultrasonography, and magnetic resonance imaging (MRI) can be performed. However, they are not very useful in the detection of Parkinson's disease.

Parkinson's disease takes time to diagnose. Following the diagnosis, regular meetings with neurologists are required to assess the patient's status and symptoms over time and accurately diagnose this disease. According to a study, the clinical diagnosis of Parkinson's disease is accurate 80.6% of the time [1].

With the invasion of technology also, there are various deep neural network (DNN), machine learning (ML), and artificial neural network (ANN) models available but they are extremely generalized. No two people experience this disease in the same way. Hence, it has been decided to come up with an amalgamated and personalized model that takes into consideration various parameters like medical history, voice analysis, and handwriting analysis which will potentially help in detecting this disease.

K. Harshitha (✉) · T. R. Vinay · K. Keerti · M. Shreya
Department of CSE, Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: harshithak711@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_7

2 Related Work

A literature survey of some latest research papers gave an insight into how to go about the project and the areas that require focus to overcome the shortcomings of the previous work.

The following significant insights can be drawn from the research paper [2]:

- Provide the long-term monitoring and detection of movement-related and other non-movement-related symptoms.
- There will be continuous monitoring.
- Smartphones are widely used these days and making use of them as smart devices with multiple sensors is more feasible.
- Since all minor symptoms are to be captured, monitoring must be done for a long time. The users must trust the system and be up to it.
- More data is required.

According to the research paper [3], the following major insights can be derived:

- From the speech recordings, feature extraction, feature subset selection, and classifying features were done. The performance of the system is assessed.
- In this voice-based detection system,
 - The most recent and largest publicly available dataset is used.
 - A small number of features are considered.
- The key lies in selecting features in a small number, which are relevant and unrelated.
- No other symptoms are explored.

The authors of the research paper [4] propose the following:

- Two vocal recordings are taken using a smartphone (SP), microphone, and acoustic cardioid (AC) channels. This data is classified using support vector machine (SVM) and K-nearest neighbors (KNN) algorithms.
- The features of voice data such as vocal frequency when the patient pronounces certain words are taken as inputs.
- Validation is checked.
- No other symptoms are explored.
- In this model, the language used in the input must be the same as dataset languages.

The research paper [5] suggests the following:

- The following types of algorithms are used in this system:
 - Four feature selection algorithms.
 - Six classification algorithms multilayer perceptron (MLP), Naive Bayes, K-nearest neighbors (KNN), and support vector machine (SVM).
 - Two validation algorithms.

- Some new dysphonia measures were used to extract 132 features of speech signals. The dataset comprises only certain vowels.
- Feature selection methods are studied for feasibility and then applied to the samples. This is to find more relevant features as this will increase the performance of the classifiers.
- The input words are restricted to certain words only.
- No other symptoms are explored.

The authors of the research paper [6] intend to put forth the following:

- Parkinson’s disease (PD) is detected by analyzing gait features using a deep learning approach.
- Data is collected via wireless sensors. This can be used to know the severity level of the disease progression.
- Sensors are placed under the feet of the patient, which will measure the changing weight when the patient is in motion.
- No other symptoms are explored.

The research paper [7] proposes the following methodology:

- Speech is one of the factors which help in the detection of Parkinson’s disease as it affects various factors of speech.
- Speech signals are recorded, pre-processed, conversion into intrinsic mode functions (IMF), feature extraction, and classification using support vector machine (SVM) and random forest (RF) are performed.
- The voice signals are converted to IMFs by making use of empirical mode decomposition (EMD) technique.
- The following insights can be derived:
 - Initial four IMFs give vocal tract information of the patient.
 - Other IMFs give vocal fold vibration information.
- Only the voice dataset is used. Other symptoms are not explored.

Research paper [8] suggests the following:

- Thirteen people with Parkinson’s disease were chosen for the study and were required to wear a flexible sensor and a smartwatch on that hand that is most affected.
- A physician rated the severity of tremor and uncontrolled voluntary movements in each hand as the participants completed various motor tasks. Then, machine learning models were employed on acquired data to distinguish between them and their performance was compared while utilizing multiple types of sensors.
- Data collection is very flexible as wearable sensors and smartwatch is used on the affected hand.
- Sensors have limited battery life and memory capacity.

The following is propounded by research paper [9]:

- A new hybrid intelligent system is developed which is used to perform voice signal analysis to detect Parkinson's disease.
- The suggested method employs genetic algorithms (GA) and linear discriminant analysis (LDA) for dimensionality reduction and neural network optimization, respectively.
- The proposed system has improved performance and lesser complexity than the existing systems.
- Other symptoms are not taken into consideration.

Research paper [10] takes the following into consideration:

- Patients' habits like smoking, alcohol, etc., are taken into consideration, and their risk factor is calculated.
- Positive predictive values (PPVs) are calculated using Bayes' theorem. The dataset is separated into 70% development and 30% validation samples using a random sampling of individuals.
- Symptoms appear in patients several years before a diagnosis is made. Early detection aids in more effective therapy, which enhances the standard of living.
- The proposed model is not completely accurate because it is influenced by medical history and genetic factors.

The authors of the research paper [11] intend to put forth the following technique:

- The suggested methodology analyzes the voice data of the patients to predict the severity of the disease.
- To divide the patient's voice dataset into "severe" and "non-severe" categories, deep learning is utilized.
- An input layer, hidden layers, and output layer make up a deep neural network. Two neurons in the output layer corresponding to the two classes: "severe" and "non-severe." The results were reported in binary form, with 1 indicating Parkinson's disease and 0 indicating healthy.
- No other symptoms are explored.

The research paper [12] proposes the following methodology:

- The main goal is early detection of Parkinson's disease using voice analysis
- Among the various features of the database like shimmer, jitter, frequency, and generic algorithms are employed to choose a set of features among these, such that maximum accuracy can be achieved.
- AdaBoost and bagging algorithms have been used as classifiers to detect Parkinson's.
- Compared with existing methods, the proposed system uses fewer features for classification and achieves higher accuracy. No other symptoms are explored as only voice is taken into consideration.

The following can be perceived from the research paper [13]:

- The voice dataset collected was given as inputs measured in the form of frequency, shimmer, jitter, etc.
- Data from the retrieved dataset is classified using supervised (SVM and KNN) and unsupervised learning methods.
- Output was given in the binary form (1 being Parkinson's, 0 being healthy).
- A relative study and accuracy of different classification methods are carried out.
- Only voice is taken into consideration. Other symptoms are not explored.

The authors of the research paper [14] propose the following:

- Voice dataset is taken and fed into the velocities of articulators model. Normal and novel voices are compared and results are given.
- The Interspeech 2015 Computational Paralinguistic Challenge provided the database.
- The features of this dataset consist of many aspects like vocal tract dynamics, vocal tract stability, loudness, phonation to effectively aid the detection of Parkinson's disease.
- This study majorly focuses on finding features with a neurological connection which may be the factor affected in Parkinson's disease patients. It aims to expand the existing dataset with more features.
- Only voice is taken into consideration. No other symptoms were explored.

The following method is suggested by the research paper [15]:

- The aim of the study is to develop a suitable test for neurological illnesses like Parkinson's disease.
- A set of movement patterns that have not been detected earlier in Parkinson's patients are studied, which is unlike other normal testing methods.
- Passive marker-based motion analyzers are used to record movement patterns.
- Based on the recordings, a standard measurement setup and evaluation algorithms must be employed to help in deriving more insights.
- No other symptoms are explored.

The system suggested by the authors of the research paper [16] is:

- Voice data from various patients are collected and fed to the hybrid system which performs data preprocessing, feature selection or reduction, and classification.
- Clustering techniques are used to distinguish features like difficulty in speaking, stutter which are typically observed in Parkinson's.
- It is observed that the accuracy of Parkinson's disease prediction is 100% when a hybrid method is employed.
- No other symptoms are explored.

2.1 Summary of Literature Survey

Some of the evident and major deliverables from the literature survey are:

- Only the criteria of voice are taken into consideration, which may or may not be reliable in most cases.
- Other major symptoms are almost unexplored.
- These tests are not personalized. Only general outcomes are given.
- The patient’s health history is not taken into consideration.

3 Proposed Methodology

The model proposed is not exclusive to any one symptom of Parkinson’s. Since the goal is to make it as personalized as possible, factors of personal health conditions along with their voice and handwriting are taken into consideration and an overall result is obtained which will help the user in taking appropriate measures to alter their lifestyle to help not aggravate this disease.

To achieve this, various machine learning models are built which are ingenious. The inputs given by the user are quantified to the required type and fed to the model which will give the best/most accurate results. Furthermore, the results obtained from the models are processed and the most suitable and final output is given with three possibilities:

1. There are high chances the person has developed PD (Parkinson’s disease).
2. There are chances the person may develop PD in the future.
3. The person is healthy (Fig. 1).

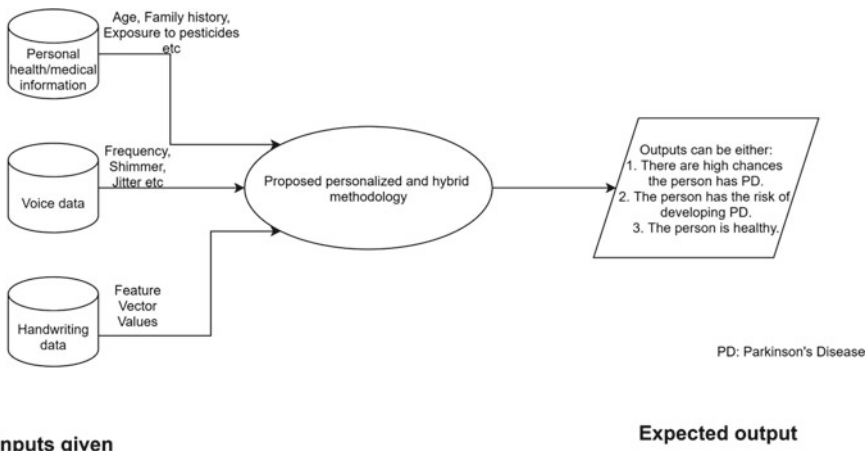


Fig. 1 PHASE 0: data flow diagram

4 Technological Concepts Used

4.1 Health/Medical History Analysis:

- Based on studies from various medical journals, some of the major factors that may lead to the development of Parkinson’s disease were compiled [17, 18].
- Questions are framed with certain weights as follows:
 - I. Is your age over 60?
Yes-1 No-0.
 - II. Your gender?
Male-1 Female-0.
 - III. Do you have a parent(s) or sibling(s) who is/are affected by Parkinson’s?
Yes-1 No-0.
 - IV. Have you ever been exposed to chemicals like pesticides and herbicides; Agent Orange (an herbicide and defoliant chemical); or worked with heavy metals, detergents, or solvents?
Yes-1 No-0.
 - V. Have you ever received severe head trauma?
Yes-1 No-0.
 - VI. Do you consume medications such as antipsychotics for treating severe paranoia and schizophrenia?
Yes-1 No-0.
 - VII. Have you been exposed to nicotine for longer durations of time?
Yes-0 No-1.
 - VIII. Have you been consuming coffee/caffeine products for a long duration of time?
Yes-0 No-1.
 - IX. Do you lead an active lifestyle with regular exercise?
Yes-0 No-1.
 - X. Have you been consuming statins to reduce cholesterol levels?
Yes-0 No-1.
- 1 indicates that the probability of developing Parkinson’s disease is high while 0 indicates the opposite.
- Set the threshold value to 5 which is the midpoint of the sum weightage of each question.
- If the patients’ score is above 5, the probability of developing the disease is very high. If less than 4, the probability is less.
- A score around the midpoint, i.e., 4, 5, or 6, may indicate that the person is at risk of developing this disease later on.
- The output from this module is known as “Result 1” (Fig. 2).



Fig. 2 Scale indicating the probability of developing Parkinson's disease

4.2 Voice Analysis

Feature extraction from voice

- The dataset from UCI machine learning repository is used [19].
- The features will be extracted which are in alignment with the datasets used. These features are extracted using the multi-dimensional voice program (MDVP).
- Further, each row of the dataset contains a name and status. Status can be 0 (no Parkinson's disease) or 1 (Parkinson's disease).
- The extracted features can be merged into a dataset by grouping them into a single table.

K-Nearest Neighbors (KNN) Algorithm

- This algorithm works by learning a well-labeled set of data and produces the most appropriate output when unlabeled data is fed as input.
- The outputs will be in the form of 1 or 0 indicating a yes or no.
- This algorithm works based on "similar things are close to each other." A deeper understanding can be obtained by plotting the data.
- The given Parkinson's voice dataset can be fitted in a K-nearest neighbor model
- After fitting and training, the test data prediction gives us an accuracy of 91.83673469387756%.
- Output from this module is known as "Result 2."

4.3 Handwriting Analysis

Histogram of Oriented Gradients (HOG) Descriptor

- It is a feature descriptor that is utilized to extract features from images. It is commonly used in computer vision for object detection.
- It mainly concentrates on the structure or shape of an object, as well as changes in direction of the object's edges.
- It is used to quantify the spiral images and return the feature vectors.

- The complete input image is separated into several parts, where the magnitude and direction are calculated for each part and these are utilized to determine the change in direction of spiral images.
- As a result, it is used to distinguish between the shape of the spiral in PD and non-PD patients.

Random Forest (RF) Classifier

- This is a technique that builds a series of decision trees from a randomly selected subset of the training data.
- The votes from several decision trees are then combined to determine the final result or class of the object.
- The feature vectors from all images retrieved using the HOG descriptor are used to train a random forest model.
- It is used to classify testing data as healthy or Parkinson's, and the accuracy of the model is calculated.
- Finally, the spiral images will be classified as healthy (0) or Parkinson's (1).
- Output from this module is known as "Result 3."

4.4 Final Result

- Assign the intermediate result values to variables.
- Based on their values, we can make the best possible predictions as follows:
 - If (Result 1 < 4) and (Result 2 and Result 3 = 0), then the person is healthy.
 - If (Result 1 > = 7 and Result 1 < = 10) and (Result 2 or Result 3 = 1), then there are high chances the person has developed PD (Parkinson's disease).
 - If (Result 1 = 5 or 4 or 6) and (Result 2 and Result 3 = 0), then there are chances the person may develop PD in the future (Fig. 3).

5 Results

The expected results from the possible inputs and the verification based on actual results are as in Table 1.

6 Conclusion

In conclusion, this novel, and amalgamated prediction system is expected to give the best possible output regarding the judgment of the presence or development of this disease. Compared to the other models that are present, this model takes into consideration the health and lifestyle conditions of the patients along with other

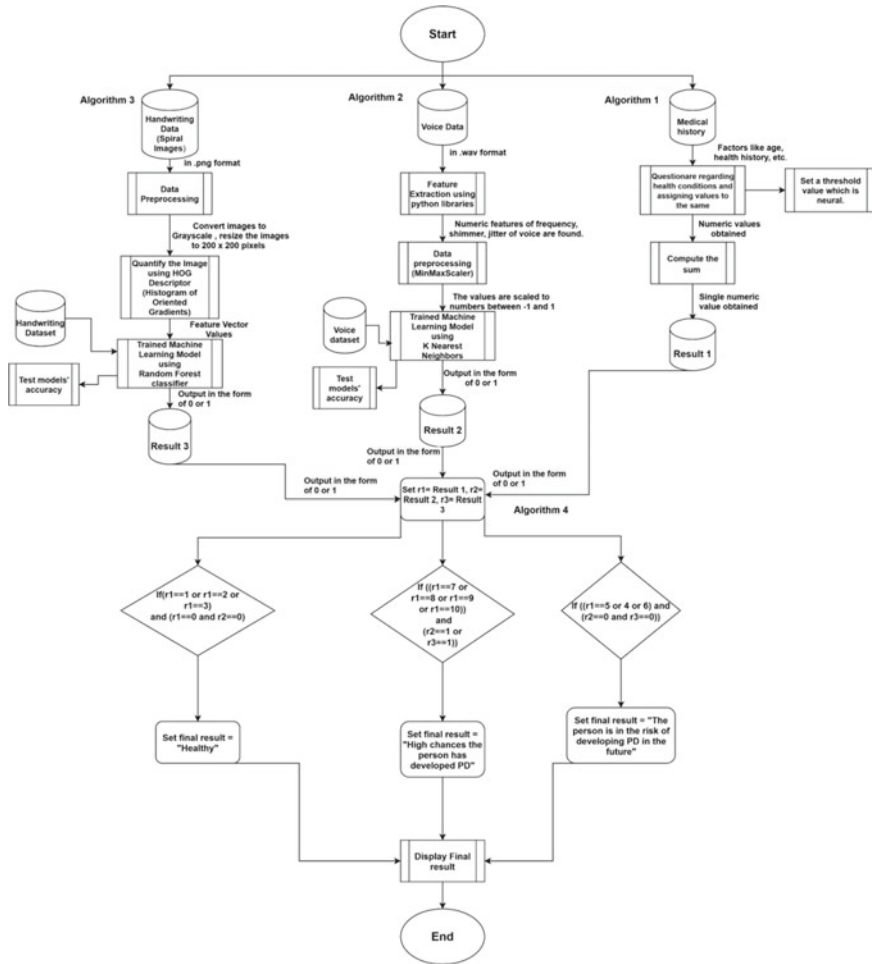





Fig. 3 PHASE 1: data flow diagram

detection tests. This satisfies the goal of making it as personalized as possible as no two people experience the disease in the same way.



Further, we can incorporate more detection tests which will help in making the system even more reliable.

Table 1 Result validation table

Sample Input	Health/Medical History (Parameter 1)	Voice (After feature extraction) (Parameter 2)	Handwriting (Parameter 3)	Actual Results of the parameters	Observed Results of parameters 2 and 3	Validation (Actual Result vs Observed Result of parameters 2 and 3)
1.	<p>Have you been consuming coffee/caffeine products for a long duration of time? No-1</p> <p>Sum=1</p> <p>(The choices chosen for the remaining questions are 0)</p>	<p>119.99200, 157.30200, 74.99700, 0.00784, 0.00007, 0.00370, 0.00554, 0.01109, 0.04374, 0.42600, 0.02182, 0.03130, 0.02971, 0.06545, 0.02211, 21.03300, 0.414783, 0.815285, -4.813031, 0.266482, 2.301442, 0.284654</p>		<p>Parameter 1 - 1 Indicates a low risk of developing Parkinson's Disease (PD)</p> <p>Parameter 2 - 1</p> <p>Parameter 3 - 1</p>	<p>Parameter 2 - 1</p> <p>Parameter 3 - 1</p>	<p>Parameter 2 - Success</p> <p>Parameter 3 - Success</p>
2.	<p>Is your age over 60? Yes-1</p> <p>Have you been exposed to nicotine for longer durations of time? No-1</p> <p>Sum=2</p>	<p>122.40000, 148.65000, 113.81900, 0.00968, 0.00008, 0.00465, 0.00696, 0.01394, 0.06134, 0.62600, 0.03134, 0.04518, 0.04368, 0.09403, 0.01929, 19.08500, 0.458359, 0.819521, -4.075192,0 .335590, 2.486855, 0.368674</p>		<p>Parameter 1 - 2 Indicates a low risk of developing PD</p> <p>Parameter 2 - 1</p> <p>Parameter 3 - 1</p>	<p>Parameter 2 - 1</p> <p>Parameter 3 - 1</p>	<p>Parameter 2 - Success</p> <p>Parameter 3 - Success</p>
3.	<p>Do you consume medications such as antipsychotics for treating severe paranoia and schizophrenia? Yes-1</p> <p>Have you been exposed to nicotine for longer durations of time? Yes-0</p>	<p>116.68200, 131.11100, 111.55500, 0.01050, 0.00009, 0.00544, 0.00781, 0.01633, 0.05233, 0.48200, 0.02757, 0.03858</p>		<p>Parameter 1 - 3 Indicates a low risk of developing PD</p> <p>Parameter 2 - 1</p> <p>Parameter 3 - 0</p>	<p>Parameter 2 - 1</p> <p>Parameter 3 - 0</p>	<p>Parameter 2 - Success</p> <p>Parameter 3 - Success</p>

(continued)

Table 1 (continued)

	<p>Have you been consuming coffee/caffeine products for a long duration of time? No-1</p> <p>Sum=3</p>	<p>0.03590, 0.08270, 0.01309, 20.65100, 0.429895, 0.825288, -4.443179, 0.311173, 2.342259, 0.332634</p>				
4.	<p>Is your age over 60? Yes-1</p> <p>Do you have a parent(s) or sibling(s) who is/are affected by Parkinson's? Yes-1</p> <p>Have you ever been exposed to chemicals like pesticides and herbicides; Agent Orange (a herbicide and defoliant chemical); or worked with heavy metals, detergents, or solvents? Yes-1</p> <p>Have you ever received severe head trauma? Yes-1</p> <p>Do you consume medications such as antipsychotics for treating severe paranoia and schizophrenia? Yes-1</p> <p>Have you been exposed to nicotine for longer durations of time? No-1</p> <p>Sum=6</p>	<p>174.18800, 230.97800, 94.26100, 0.00459, 0.00003, 0.00263, 0.00259, 0.00790, 0.04087, 0.40500, 0.02336, 0.02498, 0.02745, 0.07008, 0.02764, 19.51700, 0.448439, 0.657899, -6.538586, 0.121952, 2.657476, 0.133050</p>		<p>Parameter 1-6 Indicates a possible risk of developing PD in the future</p> <p>Parameter 2-0</p>	Parameter 2-1	<p>Parameter 2-1 Fail (The cause of failure of producing an accurate output could be because the K Nearest Neighbors algorithm can give an accuracy of 91.83% only. Hence, there is still an 8.17% possibility of failure.)</p> <p>Parameter 3-1 Success</p>
5.	<p>Is your age over 60? Yes-1</p> <p>Do you have a parent(s) or sibling(s) who is/are affected by Parkinson's? Yes-1</p> <p>Have you ever been exposed to chemicals like pesticides and herbicides;</p>	<p>209.51600, 253.01700, 89.48800, 0.00564, 0.00003, 0.00331, 0.00292, 0.00994, 0.02751, 0.26300, 0.01604, 0.01657, 0.01879,</p>		<p>Parameter 1-7 Indicates a possible high risk of developing/pre sense of PD</p> <p>Parameter 2-0</p> <p>Parameter 3-0</p>	Parameter 2-0	<p>Parameter 2-0 Success</p> <p>Parameter 3-1 Fail</p>

(continued)

Table 1 (continued)

Agent Orange (a herbicide and defoliant chemical); or worked with heavy metals, detergents, or solvents? Yes-1	0.04812, 0.01810, 19.14700, 0.431674, 0.683244, -6.195325, 0.129303, 2.784312, 0.168895					(The cause of failure of producing an accurate output could be because the Random Forest Classifier algorithm can give an accuracy of 86.67%)
Have you ever received severe head trauma? Yes-1						
Do you consume medications such as antipsychotics for treating severe paranoia and schizophrenia? Yes-1						
Do you lead an active lifestyle with regular exercise? No-1						
Have you been consuming statins to reduce cholesterol levels? No-1						
Sum=7						

References

1. Mayo Clinic Parkinson’s Disease Diagnosis and Treatment. <https://www.mayoclinic.org/diseases-conditions/parkinsons-disease/diagnosis-treatment/drc-20376062>
2. Zhang H, Song C, Rathore AS, Huang M-C, Zhang Y, Xu W (2021) mHealth technologies towards Parkinson’s disease detection and monitoring in daily life: a comprehensive review. *IEEE Rev Biomed Eng* 14:71–81. <https://doi.org/10.1109/RBME.2020.2991813>
3. Solana-Lavalle G, Galán-Hernández JC, Rosas-Romero R (2020) Automatic Parkinson disease detection at early stages as a pre-diagnosis tool by using classifiers and a small set of vocal features. *Biocybernetics Biomed Eng* 40(1):505–516. ISSN 0208–5216. <https://doi.org/10.1016/j.bbe.2020.01.003>
4. Priya TV, Sivapatham S, Kar A (2020) Parkinson’s disease detection using multiple speech signals. In: 2020 IEEE 4th conference on information and communication technology (CICT), 2020, pp 1–5. <https://doi.org/10.1109/CICT51604.2020.9312113>
5. Cantürk İ, Karabiber F (2016) A machine learning system for the diagnosis of Parkinson’s disease from speech signals and its application to multiple speech signal types. *Arab J Sci Eng* 41:5049–5059. <https://doi.org/10.1007/s13369-016-2206-3>
6. Aversano L, Bernardi ML, Cimitile M, Pecori R (2020) Early detection of Parkinson disease using deep neural networks on gait dynamics. *Int Joint Conf Neural Netw (IJCNN)* 2020:1–8. <https://doi.org/10.1109/IJCNN48605.2020.9207380>
7. Karan B, Sahu SS, Mahto K (2020) Parkinson disease prediction using intrinsic mode function based features from speech signal. *Biocybernetics Biomed Eng* 40(1):249–264. ISSN 0208–5216. <https://doi.org/10.1016/j.bbe.2019.05.005>
8. Shawen Ni, O’Brien M, Venkatesan S, Lonini L, Simuni T, Hamilton J, Ghaffari R, Rogers J, Jayaraman A (2020). Role of data measurement characteristics in the accurate detection of

- Parkinson's disease symptoms using wearable sensors. *J NeuroEng Rehabil.* 17. <https://doi.org/10.1186/s12984-020-00684-4>
9. Ali L, Zhu C, Zhang Z, Liu Y (2019) Automated detection of Parkinson's disease based on multiple types of sustained phonations using linear discriminant analysis and genetically optimized neural network. *IEEE J Trans Eng Health Med* 1–1. <https://doi.org/10.1109/JTEHM.2019.2940900>
 10. Schrag A, Anastasiou Z, Ambler G, Noyce A, Walters K (2019) Predicting diagnosis of Parkinson's disease: a risk algorithm based on primary care presentations. <https://doi.org/10.1002/mds.27616>
 11. Grover S, Bhartia S, Yadav A, Seeja KR (2018) Predicting severity of Parkinson's disease using deep learning. *Procedia Comput Sci* 132:1788–1794. ISSN 1877–0509. <https://doi.org/10.1016/j.procs.2018.05.154>
 12. Fayyazifar N, Samadiani N (2017) Parkinson's disease detection using ensemble techniques and genetic algorithm. *Artif Intell Sig Process Conf (AISP) 2017*:162–165. <https://doi.org/10.1109/AISP.2017.8324074>
 13. Sriram TV, Rao MV, Narayana GVS, Kaladhar DSVGK (2015) Diagnosis of Parkinson disease using machine learning and data mining systems from voice dataset. In: Satapathy S, Biswal B, Udgata S, Mandal J (eds) *Proceedings of the 3rd international conference on frontiers of intelligent computing: theory and applications (FICTA) 2014. Advances in intelligent systems and computing*, vol 327. Springer, Cham. https://doi.org/10.1007/978-3-319-11933-5_17
 14. Williamson JR, Quatieri TF, Helfer BS, Ciccarella G, Mehta DD (2015) Segment-dependent dynamics in predicting Parkinson's disease. In: *Proceedings of InterSpeech*, pp 518–522
 15. Jobbagy A, Furnee H, Harcos P, Tarczy M, Krekule I, Komjathi L (1997) Analysis of movement patterns aids the early detection of Parkinson's disease. In: *Proceedings of the 19th annual international conference of the IEEE engineering in medicine and biology society. Magnificent milestones and emerging opportunities in medical engineering (Cat. No.97CH36136)*, pp 1760–1763 vol 4. <https://doi.org/10.1109/IEMBS.1997.757066>
 16. Hariharan M, Polat K, R Sindhu (2014) A new hybrid intelligent system for accurate detection of Parkinson's disease. *Comput Methods Prog Biomed* 113(3):904–913. ISSN 0169–2607. <https://doi.org/10.1016/j.cmpb.2014.01.004>
 17. Johns Hopkins Medicine, Parkinson's Disease risk factors and causes. <https://www.hopkinsmedicine.org/health/conditions-and-diseases/parkinsons-disease/parkinsons-disease-risk-factors-and-causes>
 18. Medical News Today. <https://www.medicalnewstoday.com/articles/323440>
 19. ICS UCI Machine Learning Databases. <https://archive.ics.uci.edu/ml/machine-learning-databases/parkinsons/>

Design and Implementation of Flyback Converter Topology for Dual DC Outputs



C. H. V. Ramesh, Sudeep Shetty, Shreeram V. Kulkarni, and Rajkiran Ballal

1 Introduction

Due to the increasing power demand, there is a vast increase in power generation. The generation may be from renewable or non-renewable energy sources. To reduce carbon emission and to protect the environment, we must opt for renewable energy (green energy) and the energy obtained might be in the AC or the DC forms. Due to the fact that the conversion of energy from AC–DC, DC–AC, AC–AC, DC–DC, and AC-DC-AC, the converter topologies are evolved [1–3]. There have been many different topologies in the last few years. This article focuses on the flyback converter topology for various charging adapters (mobile, laptop, etc.) applications and SMPS circuits. It is used for many low-power output applications. The output power rating of the flyback converter should not be more than 100 W. Generally suitable for high-voltage and low-power applications. Its most important features are simplicity, low cost, and galvanic insulation [4, 5].

The flyback converter is a kind of converter which is utilized to change over the electrical vitality from DC to DC or AC to DC [6]. It is like Buck–Boost converter; the distinction is transformer which is utilized for storing energy, and it gives separation among input and output in case of flyback converter. But in the case of Buck–Boost, there is no isolation provided among input and output terminals. It is SMPS circuit and used for many low-power output applications. The output power rating of the flyback converter should not be more than 100 W [7–9]. DC or unregulated DC voltage is contributed as input to the circuit, or else AC voltage is rectified and given as input to the flyback converter circuit. With this converter, single or numerous output voltages which are segregated from the input voltages can be gotten [10].

C. H. V. Ramesh (✉) · S. Shetty · S. V. Kulkarni · R. Ballal
NitteMeenakshi Institute of Technology, Bengaluru, India
e-mail: venkataramesh.ch@nmit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_8

The DC-DC converter can be divided into hard switching converters and soft-switching converters. Because recent low efficiency of hard switching converters, soft-switching technology is becoming more and more popular. Separate converter topologies for forward, reverse, push-pull, etc., are usually used for SMPS applications. The switching frequencies of these converters are typically in the kHz range. This reduces the size of the transformer [11, 12]. Flyback converter is the most used SMPS circuit for low-power output applications having the advantage of isolated output. From an energy efficiency standpoint, flyback power supplies are inferior to many other SMPS circuits. But their simple topology makes it popular for low-cost and low-output power range [13]. The [14] proposes ZVS technology for various non-isolated DC-DC converters. There is a limit to the voltage gain that can be achieved using a buck/boost converter or boost converter. Unwanted to operate a boost or buck-boost converter with a very high duty cycle the ratio due to the very high current ripple of the capacitor. And it happens to be the solution is to choose a separate topology to get the high price voltage amplification between the battery and the DC bus.

The topology of a flyback converter is essentially a coupled inductor, a PWM controlled switch on the primary side, and a diode on the secondary side of the coupled inductor with the capacitor connected across the load. Figure 1 shows the basic configuration of a flyback converter. Here, MOSFET is used to get fast dynamic control over duty ratio. The coupled inductor is used for voltage isolation with a series opposition connection.

This paper offers information regarding the classifications of the converters as per the potency and power density, SMPSs are helpful than linear power switches. Most of the advanced communications and laptop systems need SMPSs that have high-power density, high potency, and constant operating frequency. In the last decade, plenty of power convertor topology has been planned for switch mode power provider's applications. A power converter is an electrical or electro robot that is used for changing power from AC-DC or DC-DC [15]. In [16], the information regarding the classification of DC-DC converters is given. The design methodology for the transformer in a flyback converter which automatically completes soft switching of

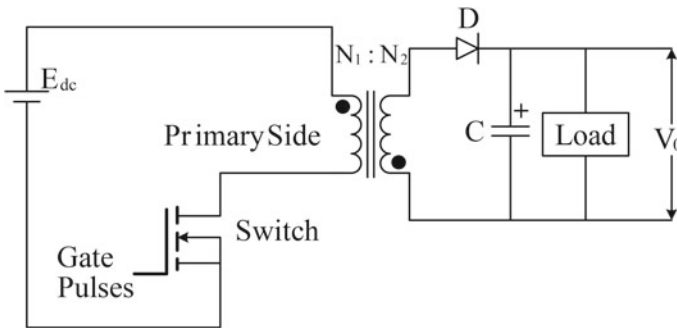


Fig. 1 Flyback converter topology

the main switch given in [17]. No extra circuit or management theme is required to achieve zero current activate and total zero voltage switch (both throughout activate and switch OFF) of the MOSFET. Also, it eliminates the dissipative snubber that is a vital part of the flyback converter. However, this theme is all applicable to fastened input and constant load applications.

2 Flyback Converter: Principle of Working

The flyback converter topology circuitry is shown in Fig. 1. After rectification and filtering of utility AC supply, unregulated DC voltage is obtained to input the converter. MOSFET is used as a quick switching gadget to control the duty ratio to maintain the desired output voltage. Gate pulse to this switching device is given by PWM controller which has 100 kHz of switching frequency. To provide seclusion among input and output, a flyback transformer is used. When switch is in ON condition, transformer will get energized from the input voltage source while the output load is supplied from the capacitor connected at the output side. When switch is in OFF condition, the transformer will transfer the energy to output load and output capacitor. Switch arranged in series with transformer primary winding and its duty ratio is controlled to acquire a desired maximum value. The mode of operation and working principles is explained in the subsequent section and depicted in Figs. 2 and 3.

2.1 Mode of Operation

- The input voltage source is directly connected to the transformer primary winding when the switch is closed. The transformer stores energy as the primary current and magnetic flux increase in the transformer. Due to sign convention, a negative voltage is prompted in the transformer so the diode gets reverse biased. Then, output load is provided from the output capacitor.

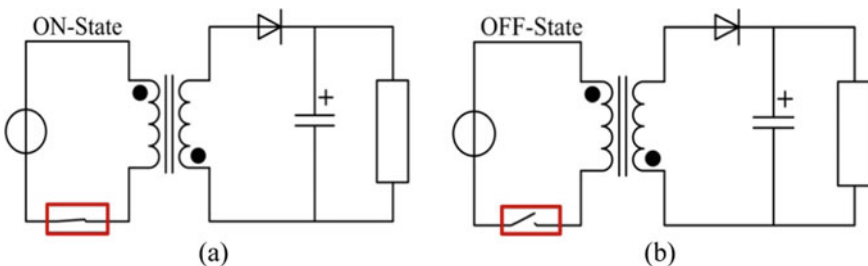


Fig. 2 Mode of operation **a** switch is closed-ON and **b** switch in open-OFF

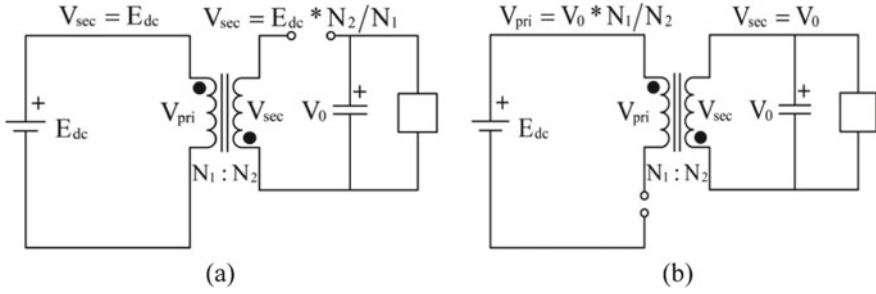


Fig. 3 Mode of operation **a** Mode 1 equivalent circuit diagram and **b** Mode 2 equivalent circuit diagram

- There will be primary and magnetic flux drop when the switch is in OFF condition. Due to sign tradition, the secondary voltage becomes positive, and the diode is forward biased, which makes the current flow from secondary of the transformer as a result output capacitor get energized and supplies load.

Different circuit configurations should be accepted during the task of the flyback converter. Each circuit configuration is referred to as the mode of circuit operation. With the assistance of an equivalent circuit, different modes of operation of the flyback converter circuit are explained.

Mode 1: The primary winding of a transformer is associated with the input supply with its spotted end getting connected to the positive side when the switch is closed. Then, current starts to stream in the primary winding. Due to reverse biasing of diode, current flowing in secondary winding will get blocked. This primary current is in charge of establishing flux in the core of the transformer. In the equivalent circuit demonstrated below, the device which is conducting will be taken as a short circuit and which is not conducting is taken as an open circuit. The switches or diodes which we are utilizing are expected to perfect in nature with zero voltage drop during ON state and zero spillage current during OFF state. The current through primary winding can be communicated by giving connection DC voltage, given by Eq. (1).

$$E_{dc} = L_{pri} * \frac{dI_{pri}}{dt} \tag{1}$$

where E_{dc} is DC input voltage, L_{pri} is primary winding inductance, I_{pri} is current through the primary winding.

During this operation, voltage induced in secondary winding is constant and equals to Eq. (2),

$$V_{sec} = E_{dc} * \frac{N_2}{N_1} \tag{2}$$

The voltage across the diode which is arranged in series with secondary winding is as Eq. (3),

$$V_d = V_0 + V_{\text{sec}} \quad (3)$$

Mode 2: After conducting for some time, the switch is going to turn OFF. The primary side becomes an open circuit. Both voltage and the current drop in the primary winding. Forward biasing of diode takes place due to change in the polarity of secondary winding, so diode starts conducting and recharges the capacitor to supply load. The secondary current concerning secondary inductance during this operation is given by Eqs. (4) and (5),

$$L_{\text{sec}} * \frac{dI_{\text{sec}}}{dt} = -V_0 \quad (4)$$

The voltage across switch during this operation is

$$V_{\text{SW}} = E_{\text{dc}} + V_0 * \frac{N_2}{N_1} \quad (5)$$

3 Flyback Converter Transformer Design

Considering input ($V_{\text{ccmax}} = 370$ and $V_{\text{ccmin}} = 210$), $D_{\text{max}} = 0.45$ and $B_{\text{max}} = 0.2$ wb/m², the circuit parameters for first DC output ($V_{01} = 12$ V and $I_{01} = 2$ A) are calculated as follows.

The secondary power (P_{02}) can be calculated using

$$P_{02} = \sum V_{0i} * I_0 \left(\frac{1 - D_{\text{min}}}{D_{\text{min}}} \right) \quad (6)$$

$$V_{0i} = V_{01} + V_D \quad (7)$$

V_D is the voltage drop across the diode

V_{01} is the first output DC voltage.

3.1 Calculation of the Minimum Duty Ratio (D_{min})

The minimum duty ratio is calculated by using Eq. (8)

$$D_{\text{min}} = \frac{D_{\text{max}}}{D_{\text{max}} + (1 - D_{\text{max}}) * V_{\text{cc max}} / V_{\text{cc min}}} \quad (8)$$

Now, substituting the Eq. (8) in Eq. (6), we get the value of P_{02} . The calculation of turn's ratio N can be calculated using Eq. (9)

$$N = \frac{V_{0i}}{V_{cc\ min}} * \frac{(1 - D_{max})}{D_{max}} \tag{9}$$

where N is secondary to the primary turns ratio. Then, the core selection of the flyback converter is given by Eq. (10)

$$A_p = \frac{P_{02} \left(\frac{1}{\eta} \sqrt{\frac{4D}{3}} + \sqrt{\frac{4(1-D)}{3}} \right)}{K * \omega * J * B_m * f_s} \tag{10}$$

where K_w is the window utilization factor, J is the current density, B_m is the maximum flux density, and f_s is the switching frequency (100 kHz). Considering 80% efficiency and the calculated value of $P_{02} = 58.17$ W, we get $A_p = 5304$ mm². Therefore, the EE30/15/7 core is chosen for the design. And the calculation of the number of turns in primary and secondary winding can be found by Eqs. (11) and (12)

$$N_1 = \frac{V_{cc\ max} * D_{min}}{A_w * B_m * f_s} \tag{11}$$

N_1 is the number of turns in the primary winding

$$N_2 = N * N_1 \tag{12}$$

By substituting the calculated turns ratio $N = 0.0784$ and the number of turns in the primary winding (N_1), we can get the number of turns in the secondary (N_2) as 4.

Similarly, for $V_0 = 5$ v, $I_0 = 0.5$ A, and $P_{02} = 14$ W, the number of turns for the supply output is calculated in the same manner, and we get the number of turns = 2.

The design values of the flyback converter are stated in Table 1.

Table 1 Design parameters of flyback transformer

Parameter	Value
V_{dc}	210–370 V
V_{o1}, I_{o1}	12 V, 2A
$V_{o1}, I_{o2},$	5 V, 0.5A
D_{MAX}	0.45
B_{MAX}	0.2
Area product (A_p)	5304mm ²
No. of primary turns	50 turns
No. of secondary turns	4 and 2 turns

3.2 Selection of the MOSFET and the Output Diode

In a flyback converter, the selection of MOSFET is important, and it will carry the primary side current when the switch is in the ON position, and the voltage across the switch is zero. And the switch is OFF state, the voltage over the switch is input voltage (V_D) plus reflected secondary side voltage (V_0/n).

- The voltage over the MOSFET during ON period $V_{SW} = 0$ V
- The voltage over the MOSFET during OFF period $V_{SW} = V_D + (V_0/n)$

where V_D is the input DC voltage, $V_0 =$ output voltage. And while selecting the MOSFET, we should consider the transformer spillage inductance. We should take the ringing of the MOSFET that is 30–40% of the total voltage, and the voltage over the switch during turn OFF period is given as

$$V_{SW} = V_D + \frac{V_0}{n} + 0.03 * \left(V_D + \frac{V_0}{n} \right) \quad (13)$$

So, its desired to choose MOSFET of maximum voltage is more than the calculated value $V_{sw} = 676$ V. Accordingly, IRFBG30 is suitable. It consists of maximum voltage reading 1000 V and maximum current rating 3.1 A. And the selection of output diode in flyback converter, the voltage over the diode in when the MOSFET is ON state is

$$V_D = -(V_d * n) \quad (14)$$

Equation (14), according to the polarity of the secondary side of the transformer negative voltage, is applied over the diode. So, the diode can withstand the negative voltage (-30 V in this case). During the MOSFET OFF period, the diode is in an ON state, and the voltage over the diode is zero.

And also, in the flyback converter, the average value of the diode current is the output current is the diode must carry the output current of 2 A. So, the diode 1N4007 is a suitable diode for this application.

3.3 Selection of PWM Controller

Pulse width modulator (PWM) is a controller which is used to control the power flow from the input to the output. Varying the pulse width causes the duty ratio to change which can be used to control the power flow from the input to output. Here, the pulse generator is used to pulse for the switch we have used in the project that is MOSFET by keeping the operating frequency constant. The UC3845 is an 8 pin IC which is having high performance for fixed frequency operation usually designed for DC–DC converter applications allowing the designer with the least cost solution

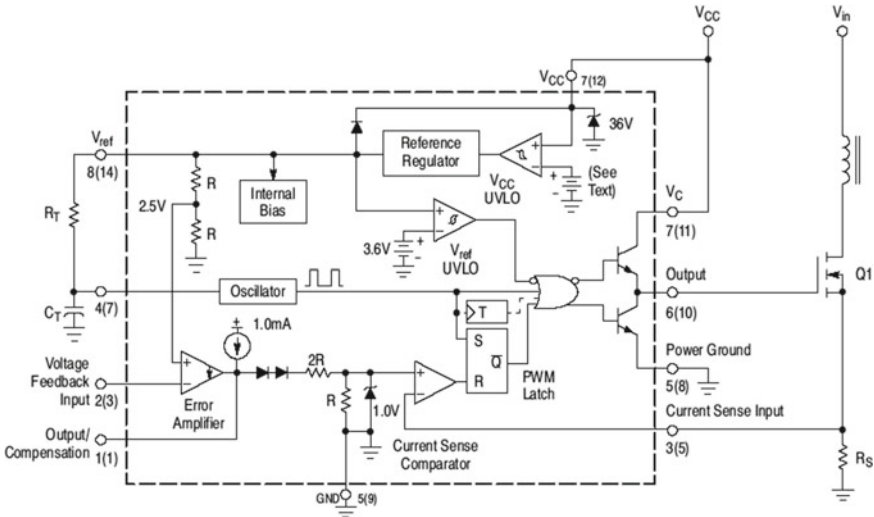


Fig. 4 Functional block diagram of PWM controller IC

with minimal connecting components. The functional block diagram of UC3845 IC is shown in Fig. 4.

Its integrated circuit has a high gain error amplifier, current sensing comparator, an oscillator, and a pulse generator very much suited for driving a power MOSFET. It also consists of protecting features like input and under-voltage lockouts. UC3845 is very much suitable for lower voltage applications with UVLO of 8.4 v (on) and 7.6 v (off). In this article, a timing resistor RRT of 20 KΩ and the timing capacitor CCT of 1000 pf are used to get the switching frequency of 100 kHz by the Fosi formula. As we have taken the duty cycle of 0.45, its period was given to 1/(45 kHz) = 22 μs.

4 Results and Discussion

The effectiveness of the designed flyback converter for different DC-link voltages is evaluated and verified through MATLAB/SIMULINK platform. And after the validation of the simulation results, the hardware realization is carried. The overall flyback converter design schematic with controller is shown in Fig. 5.

Simulation Results: Dual output waveforms of the flyback converter are shown in Figs. 6 and 7.

The hardware realization for the dual output flyback converter is shown in Fig. 8. The hardware circuit of the flyback converter with the desired output voltage levels can be seen. Hardware results are compared with the theoretical and simulation

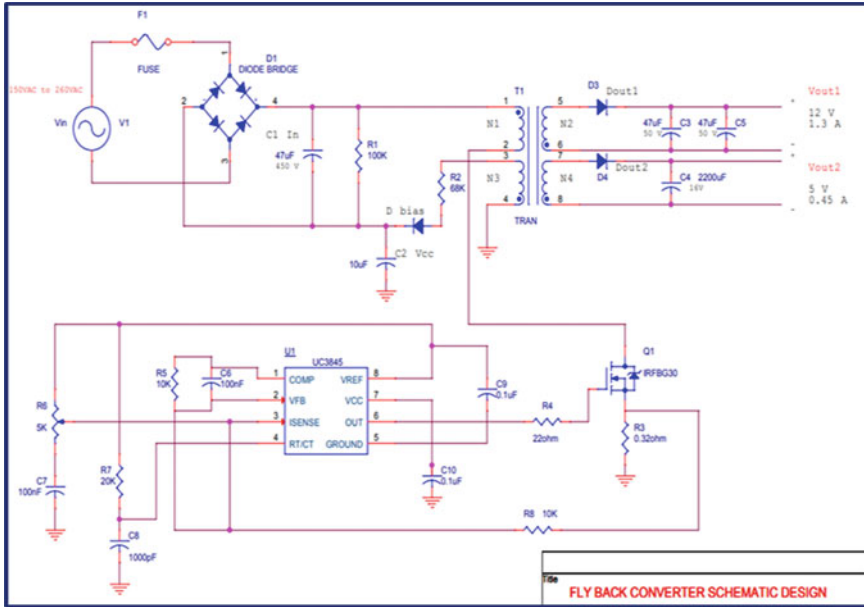


Fig. 5 Schematic diagram of flyback converter design

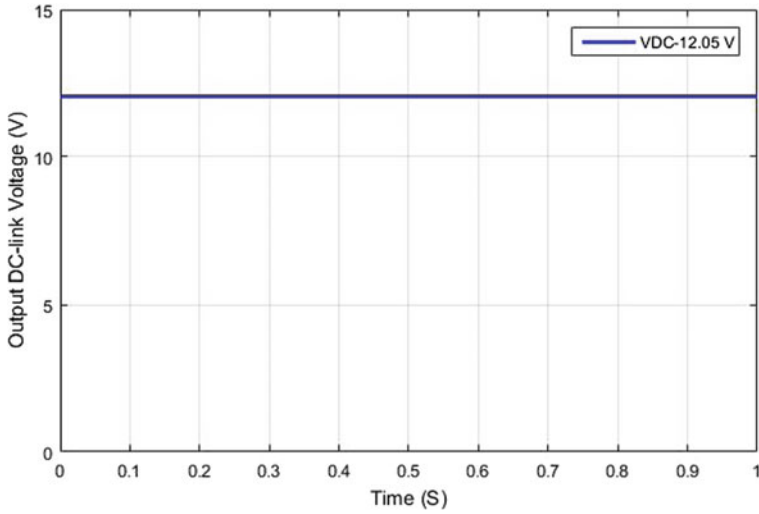


Fig. 6 Output DC-link voltage waveform $V_{01} = 12.05\text{ V}$

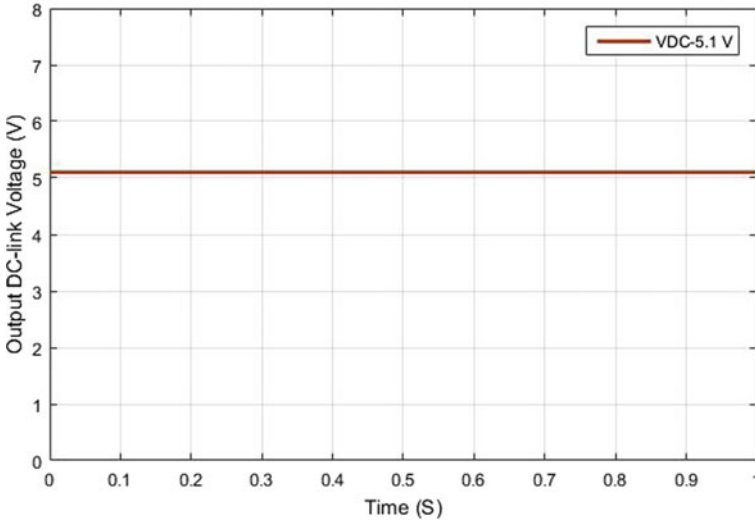


Fig. 7 Output DC-link voltage waveforms $V_{01} = 5.1$ V

results in Table 2. We can see the slight difference in the voltage levels obtained by the simulation and hardware results.

5 Conclusion

The idea of any power gadgets configuration is surely known at the point when exertion is made to foster the framework. Equipment improvement gives a chance to study issues that are typically disregarded during the hypothetical study. In this article, a methodology was made to plan and execute a flyback converter circuit. Flyback which is the most reasonable SMPS geography is designed for dual outputs with a switching frequency of 100 kHz. The flyback converter with the cautious improvement of converter transformer and PWM regulator was done. The MATLAB/SIMULINK reenactment results and the test consequences of the equipment were contrasted and recreation results and dissected.

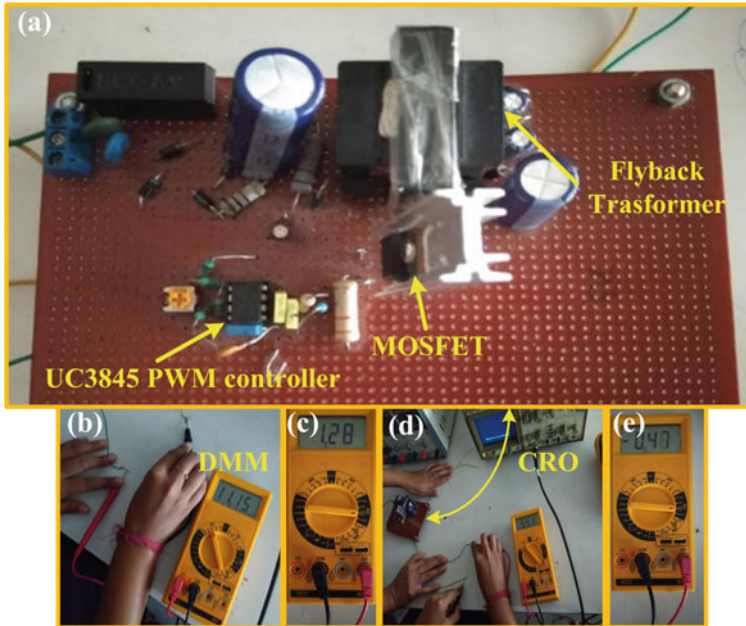


Fig. 8 Hardware realization: **a** flyback converter circuit, **b** output DC-link voltage of 11.15 V, **c** output current of 1.28 A for first supply output, **d** output DC-link voltage of 5.51 V, and **e** output current of 0.47 A for second supply output

Table 2 Comparison of output voltage with the theoretical and simulated value

Parameter	Theoretical value (V)	Simulation value (V)	Practical value (V)
V_{o1}	12	12.05	11.15
V_{o2}	5	5.1	5.51

References

- Rashid MH (2004) Solutions manual-power electronics: circuits, devices, and applications. Pearson/Prentice Hall
- Kanthimathi R, Kamala J (2015) Analysis of different flyback converter topologies. In: 2015 international conference on industrial instrumentation and control (ICIC). IEEE, pp 1248–1252
- Lin BR, Chiang HK, Chen KC, Wang D (2005) Analysis, design, and implementation of an active clamp flyback converter. In: 2005 International conference on power electronics and drives systems, vol 1. IEEE, pp 424–429
- Padiyar US, Kamath V (2016) Design and implementation of a universal input flyback converter. In: 2016 International conference on electrical, electronics, and optimization techniques (ICEEOT). IEEE, pp 3428–3433
- Ridley R (2005) Flyback converter snubber design. Switching Power Mag 12
- Bhattacharya T, Giri VS, Mathew K, Umanand L (2008) Multiphase bidirectional flyback converter topology for hybrid electric vehicles. IEEE Trans Industr Electron 56(1):78–84

7. Garcia O, Zumel P, De Castro A, Cobos A (2006) Automotive DC-DC bidirectional converter made with many interleaved buck stages. *IEEE Trans Power Electron* 21(3):578–586
8. Boyar A, Kabalci E (2018) Comparison of a two-phase interleaved boost converter and flyback converter. In: 2018 IEEE 18th international power electronics and motion control conference (PEMC). IEEE, pp 352–356
9. Zelnik R, Prazenica M (2019) Multiple output flyback converter design. *Trans Electr Eng* 8(3):32–39
10. Mukhtar NM, Lu DDC (2018) A bidirectional two-switch flyback converter with cross-coupled LCD snubbers for minimizing circulating current. *IEEE Trans Industr Electron* 66(8):5948–5957
11. Jayalakshmi NS, Gaonkar DN, Naik A (2017) Design and analysis of dual output flyback converter for standalone PV/Battery system. *Int J Renew Energy Res* 7(3):1032–1040
12. Muñoz JG, Angulo F, Angulo-García D (2021) Designing a hysteresis band in a boost flyback converter. *Mech Syst Signal Process* 147:107080
13. Kim HS, Jung JH, Baek JW, Kim HJ (2012) Analysis and design of a multioutput converter using asymmetrical PWM half-bridge flyback converter employing a parallel-series transformer. *IEEE Trans Industr Electron* 60(8):3115–3125
14. Sarani S, Abootorabi H, Delavaripour H (2021) Ripple-free input current flyback converter using a simple passive circuit. *IEEE Trans Indus Electron*
15. Ponce-Silva M, Salazar-Pérez D, Rodríguez-Benítez OM, Vela-Valdés LG, Claudio-Sánchez A, De León-Aldaco SE, Cortés-García C, Saavedra-Benítez YI, Lozoya-Ponce RE, Aquí-Tapia JA (2021) Flyback converter for solid-state lighting applications with partial energy processing. *Electronics* 10(1):60
16. Forest F, Labouré E, Gélis B, Smet V, Meynard TA, Huselstein JJ (2009) Design of inter-cell transformers for high-power multicell interleaved flyback converter. *IEEE Trans Power Electron* 24(3):580–591

Gesture Detection Using Accelerometer and Gyroscope



Raghav Gupta , Shashank Chaudhary, Akshat Vedant, Niladri Paul Choudhury, and Vandana Ladwani 

1 Introduction

Humans and machines, including computers, may now communicate more quickly because of recent electronics and sensor technology improvements. For IoT and universal computing, this human–machine interface (HMI) system will become increasingly important [1]. In most circumstances, communication begins when a machine (or an object) receives and interprets a human’s purpose (or the user). As a result, the HMI requires an input device to record the user’s intent.

Human gestures provide for a more natural approach to HMI input. Human body language is an intuitive communication technique for conveying, exchanging, interpreting, and understanding people’s thoughts, intentions, and emotions. As a result, physical language emphasizes or complements spoken language. It is a language in and of itself. Thus, human emotions, such as hand gestures, should be included for HMI input [2]. Gesture-based interactions are one of the most comfortable and straightforward ways to communicate. On the other hand, gesture recognition has various challenges before becoming widely recognized as an HMI input.

R. Gupta (✉) · S. Chaudhary · A. Vedant · N. P. Choudhury · V. Ladwani
PES University, Bangalore, India
e-mail: raghavjpr@gmail.com

S. Chaudhary
e-mail: shashankchdhry@gmail.com

A. Vedant
e-mail: akshat.shanky@gmail.com

N. P. Choudhury
e-mail: nilpc06@gmail.com

V. Ladwani
e-mail: vandanamd@pes.edu

Human hand motions are substantially less diversified than the tasks required by the HMI, which poses a considerable challenge. The functions of an (HMI) are more varied and complex. In the case of smartphones, this diversification tendency may be seen. Only a decade ago, a variety of handheld electronic devices, such as mp3 players, cell phones, and calculators, coexisted to meet various human needs. On the other hand, almost all these tasks have now converged into a single mobile device: the smartphone. On the other hand, all human intentions are only conveyed by swiping or tapping fingers on a smartphone's touch screen.

When it comes to HMI inputs, one prevalent approach is gesture-based interaction [3]. There are two hand gesture recognition techniques: vision-based recognition (VBR) and sensor-based recognition (SBR). There have been studies in gesture recognition, but most rely on computer vision. The efficiency of vision-based approaches or the operation of such devices is highly dependent on lighting conditions and camera-facing angles. It is inconvenient, and such limitations often limit the technology's usage in specific environments or for certain users.

Sensors include electromyography, touch, strain gages, flex, inertial, and ultrasonic sensors [4]. The most often utilized sensors are inertial sensors [5, 6]. Sensors with an accelerometer, gyroscopes, and magnetometers are inertial sensors.

Sensor-fusion algorithms frequently combine many sensors. For example, a glove with several wearable sensors has been claimed to monitor hand motions [7]. A 3D printer was used to create the glove housing, which includes flex sensors (on fingers), pressure sensors (at fingertips), and an inertial sensor (on the back of one's hand).

Inertial sensors are used to track hand motions in numerous sensor-fusion algorithms. Additional hand data, such as finger snapping, hand grabbing, or finger-spelling, is detected by other sensors, such as EMG. [8, 9]. Inertial and EMG sensors are a popular combination. [8–13]. The inertial sensor determines the hand location, while the EMG sensors offer additional information to comprehend complex finger or hand gestures fully. Instead of EMG sensors, strain gages, tilt, and even vision sensors can be used.

As a result, the amount of sensor data generated by these advanced gesture detection systems increases. Machine learning is being used to deal with the increasing data. Sensors are introduced to a variety of machine learning approaches. A sensor device processes a linear discriminant analysis or a support vector machine classifier. [9, 14]. In another study, a feedforward neural network (FNN) is used for digitizing, coding, and interpreting signals from a MEMS accelerometer [15].

In the meantime, inertial sensor-only techniques have been developed. This inertial-sensor-only technique may improve portability and mobility while minimizing processing needs in cases involving numerous sensors or complicated algorithms. The handwriting was rebuilt using the phone's gyroscope and accelerometer after users used a smartphone as a pen to write words. [16]. English and Chinese characters, as well as emojis, were written in handwriting. Kinematics based on inertial sensor inputs were employed in other studies to track the movement of hands and arms. [17–19]. Recognizing head or foot motions has also been described [20, 21], but they have not been modified for hand gesture identification.

Inertial sensors are unquestionably accurate and fast as HMI input devices. However, these two objectives are incompatible because increased accuracy typically increases computing load, resulting in sluggish speed. Furthermore, user movements should be uncomplicated. Inertial-sensor-based gesture recognition systems, yet again, have substantial drawbacks. One limitation is the accumulation of inertial sensor noise, which creates bias or drift in the system output [22]. The second disadvantage is that MEMS gyroscope and accelerometer signals can be jumbled [23].

From simple constructions (such as moving average filters) to the recently created outcomes, signal processing of inertial sensor outputs has been intensively researched to overcome these challenges (such as machine learning). Two recent approaches are digitizing sensor data to form codes and generating statistical measurements of the signs to describe their patterns. One method identified seven hand motions using a three-axis MEMS accelerometer. [24]. Hopfield network labels positive and negative symptoms on accelerometer signals, digitized, and restored.

These accelerometer-only systems are good at capturing linear gestures (such as up/down or left/proper patterns) but not so good at capturing circular motions (e.g., clockwise rotation or hand waving). Recognizing linear and rotational gestures has been suggested using accelerometers and gyroscopes. Using accelerometer and gyroscope sensors mounted on the forearms, the researchers used the Markov chain method to track the movement of the arms. [25]. Continuous hand gestures (CHGs), a real-time gesture identification method, were disclosed in another recent work paper. [26]. The approach begins by defining six basic gestures, determining their statistical measurements, such as means and standard deviations (STDs), then generating a database for each motion's actions.

These accelerometer-gyroscope combos are highly accurate, but they are neither portable nor inexpensive instruments. If we want to reduce the system's size and use and give numerous functions with a limited amount of hand motions, we need to find a solution. This research aimed to create a small gesture detection device and a modal HMI input device that could respond accurately and quickly to the user's intention to solve these issues.

The accelerometer, gyroscope, accelerometer-gyroscope fusion, ultrasonic, and combination accelerometer-gyroscope with electromyography approaches are used in reporting recent activities using sensor-based gestures as the HMI input. Rotational motions cannot be detected with merely an accelerometer. As a result, this paper opts for an accelerometer-gyroscope fusion system in the hopes of superior rotation sensing (than the accelerometer-only systems). We believe that the originality of this project is critical for portable HMI input devices, and it is demonstrated using an Arduino Nano 33 BLE board, a very light embedded device. It is one of the most suitable embedded devices for the project, with a weight of 5 g, a length of 45 mm, and a width of 18 mm. Although the Arduino Nano is one of the most portable and lightweight, it also poses a challenge, i.e., memory constraints. Due to its small size, it has only 1 MB of flash memory and 256 KB of static RAM, making working on it very difficult.

2 Design of the Gesture Recognition System

Our proposed system is set up to implement several essential features. First, we use a collection of simple hand motions, each with predefined functions for different application applications. Our system will be aware of the program that is now running. As a result, the procedure carried out by each motion can vary depending on the application, allowing for multifunction capabilities while reducing gesture complexity, resulting in a highly diverse HMI input device.

The second feature is that the complete hardware used is an Arduino Nano 33 BLE which is lightweight and quickly fitted on a stick. The Arduino Nano 33 BLE consists of an inbuilt IMU with one miniature three-axis gyroscope and one miniature three-axis accelerometer. The fusion of these two sensors provides us with a large amount of data about the object's acceleration and rotational motion, i.e., the Arduino Nano 33 BLE.

The third feature is hand gesture recognition in real time. To lessen the delay caused by computing load, we will train our model on a machine with a lot of processing capacity and then lower the model's size using quantization, which reduces the model's height to the extent that we can handle it.

The last feature is system accuracy. Even though the complexity grows and numerous sensors are employed as input sensors to produce a single gesture, sufficient accuracy should be ensured. We strive to apply a pre-processing approach of rasterization that turns the data from the accelerometer and gyroscope into a rasterized image. We train our model, which gives us a very high accuracy, to eliminate errors caused by hand tremors or inadvertent hand gestures.

Our input device can be used in a variety of ways. This technology can benefit input devices such as computers, laptops, portable multimedia players, wireless remote controllers for presentation applications, and virtual reality modules. For example, a user might connect the input device to a laptop and give a presentation to an audience. Pause, play, or turn up the volume if he wants to view a video.

Even if we want to interact with the computer in such situations, many input devices, such as a keyboard or mouse, may be required. However, all these can be replaced by a single input device, which is portable, accurate, and our approach's primary target.

An overview of the system design is shown in Fig. 1. The IMU containing accelerometer and gyroscope generates acceleration and angular velocity data from hand gestures. It feeds it to a process of rasterization, which converts this data feed to a rasterized image which is then given to a CNN machine learning model for training. This pre-processing of rasterization ensures that the model is free of sensor noise, limitations, or unwanted gestures. In addition, initially, while training, the machine "learns" the preferences and habits of users. The pattern is fitted according to the user's gestures needs through the data of a single motion multiple times (Fig. 2).

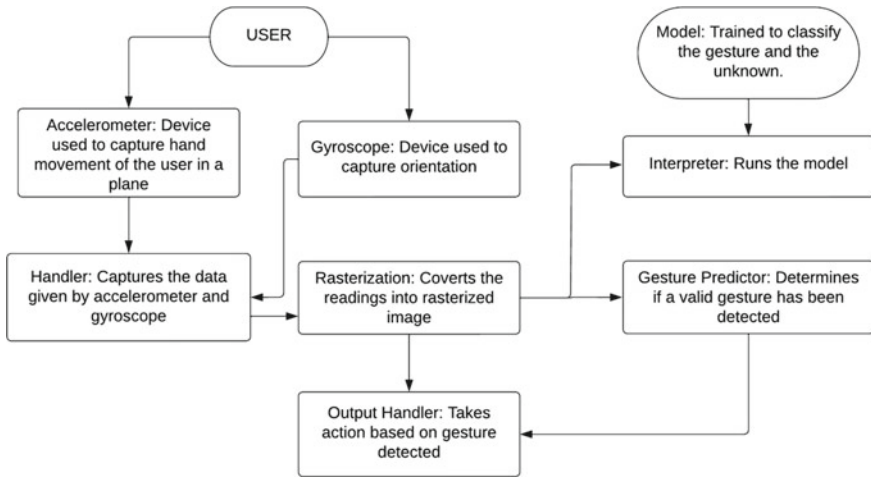


Fig. 1 Flowchart of the system design and data

While Training A Gesture Prediction Model:

Collect Gesture Data



Do the flattening or rasterization and convert to 2D images



Train the model using the rasterized image

At the time of Gesture prediction:

Take the information



Do the flattening



Use the machine learning method for classifications.

Fig. 2 Flowchart when the system is the indifferent process: a while training; b while predicting gesture

3 Hand Gesture Recognition Algorithm

The gesture recognition application accomplishes a reasonably complicated task by carefully crafting a 2D image from 3D IMU data. The dataflow is as follows:

1. The accelerometer data is read and passed to the EstimateGravityDirection (), which is used to determine the orientation of the Arduino concerning the ground.
2. The accelerometer data is passed to UpdateVelocity (), which is used to calculate the velocity of the Arduino.
3. The direction of gravity is passed to UpdateVelocity () and is used to cancel out the acceleration due to gravity from the accelerometer data.

4. The velocity is then passed to `EstimateGyroscopeDrift ()`, determining if the Arduino is stationary or moving.
5. The gyroscope data is passed to `EstimateGyroscopeDrift ()`, which calculates the gyroscope's sensor drift if the Arduino is not moving (velocity is 0).
6. The gyroscope data is passed to `UpdateOrientation ()`, where it is integrated to determine the angular orientation of the Arduino.
7. The gyroscope drift is also passed to `UpdateOrientation ()` and subtracted from the gyroscope reading to cancel the essence.
8. The 3D angular orientation is passed to `UpdateStroke ()` and transformed into 2D positional coordinates. `UpdateStroke ()` also handles whether the current gesture has ended, or a new motion has been started by analyzing the length of the gesture and testing whether the orientation data is still changing.
9. The direction of gravity is also passed to `UpdateStroke ()` to determine the roll orientation of the Arduino.
10. The 2D positional coordinates are passed to `RasterizeStroke ()`, which takes the 2D coordinates and draws lines between them on a 2D image. The color of the lines shifts from red to green to blue to indicate the direction of motion during the gesture.
11. The 2D image of the gesture is converted to ASCII art and printed on the serial monitor.
12. The 2D image of the gesture is passed to the model.
13. The model predicts the label of the gesture, and the title is printed on the serial monitor. Figure 3 depicts the above algorithm workflow.

4 Implementation

There are two primary components: the initialization phase and the main loop (Fig. 4).

4.1 Initialization

The initialization phase's job is primarily to set up the IMU, and all the resources needed to run the TensorFlow lite macro-model (Fig. 5).

The first step of the initialization phase is the IMU initialization, which is done using this **setup IMU** routine. When you go into the setup IMU routine, you will find device-specific calls that tap into the IMU functions that the library provides (Fig. 6).

The second component is setting up all the resources needed to run the model. This might be the model's pointer, the interpreter's initialization using the Tensor arena, the model, the observer, etc.

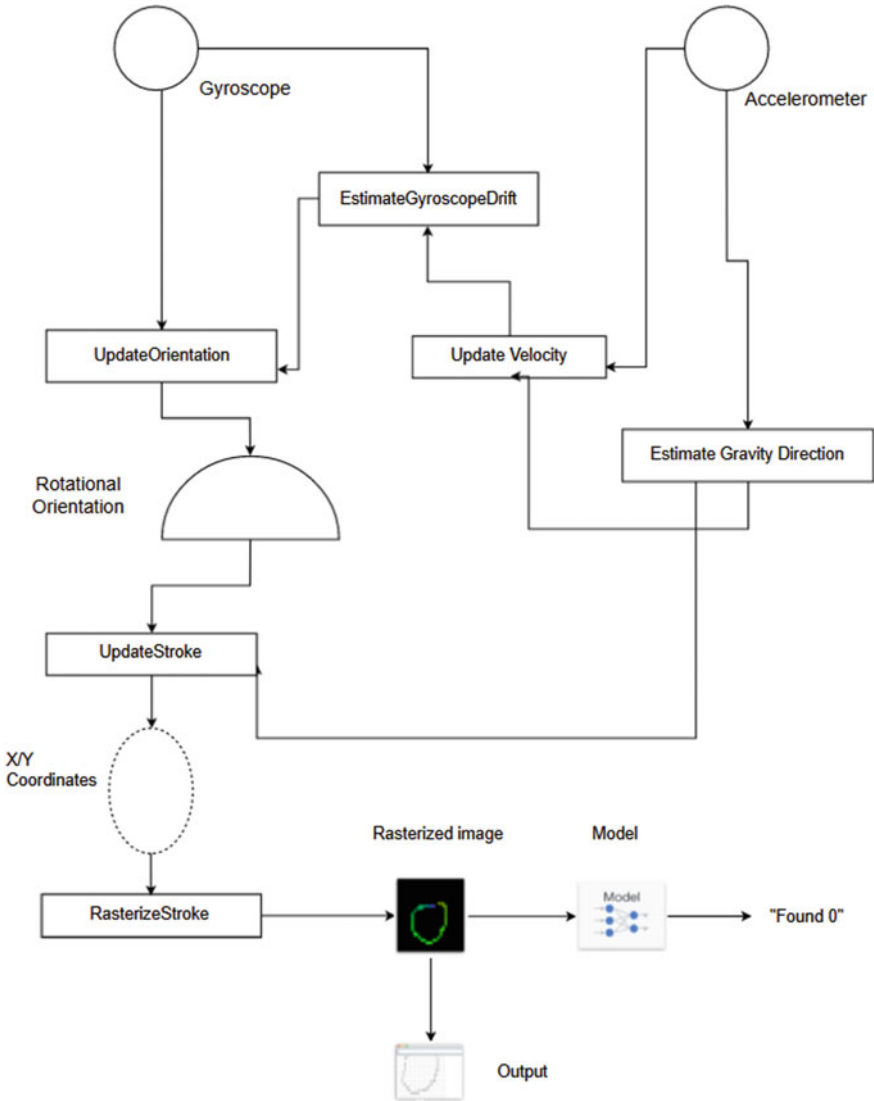


Fig. 3 Hand gesture recognition algorithm workflow

4.2 IMU Provider—Pre-processing

Its job is to get data from the gyroscope and the accelerometer and then process it. Function calls readily available will allow us to read the data from the gyroscope and the accelerometer. So, if data is available from the IMU, we will process that data (Fig. 7).

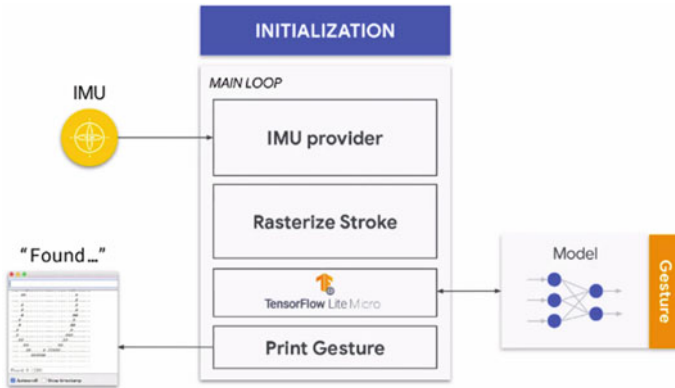


Fig. 4 Implementation of Arduino application

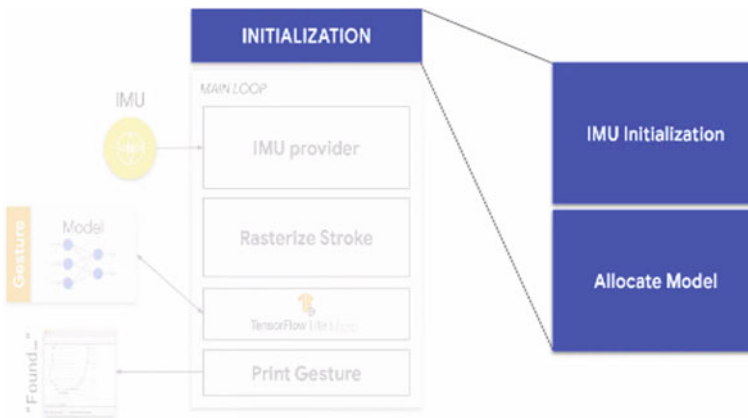


Fig. 5 Initialization of IMU

The gyroscope ends up having a little bit of drift, so we must compensate for that drift. And that is what this function **estimating gyroscope drift** is doing. When we determine that the IMU is not moving using the accelerometer, we can calculate the gyroscope’s importance and then account for it. Next, we want to integrate the gyroscope’s incremental angular changes that are coming in overtime because that will give us a part of the gesture in the spherical coordinate system, and that is how this function **updates orientation**. It is trying to capture that part that is coming in continuously. Next, we effectively want to project it into a two-dimensional plane inside this physical system. Well, that is because it is much easier to understand a 2D gesture than a complex 3D gesture. And therefore, **update stroke** is going to do that flat mapping.

Then comes the process of processing the accelerometer data. We want to estimate the gravity’s direction to control the sensor’s role in the gyroscope readings. Everyone

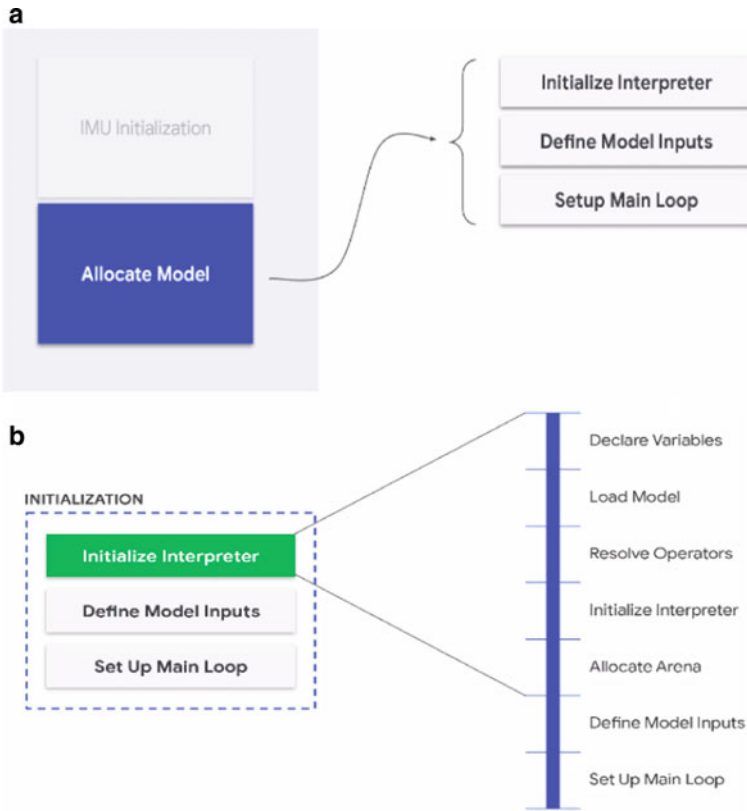


Fig. 6 The model initialization. **a** Model variable and space allocation, **b** model interpreter variable allocation

is going to be holding the stick at a specific angular momentum. So, this means that you must neutralize or normalize for that effect. For example, you might have the bar with your right hand or hold the post with your left hand. However, the gesture that you are performing is the same thing. Either way, we have got the same number written, so we get to compensate for that. And the way we do that is by effectively trying to figure out the role of the gyroscopes reading. And then, we update the velocity to know when the sensor is still, and we can correct the sensor drift.

4.3 Rasterize Stroke—Pre-processing

After that, the step of effectively capturing the data is to rasterize that stroke. We pre-process this because it is easy to feed an image into a convolutional neural network. And there is a function that helps us do that: rasterized stroke (Figs. 8 and 9).

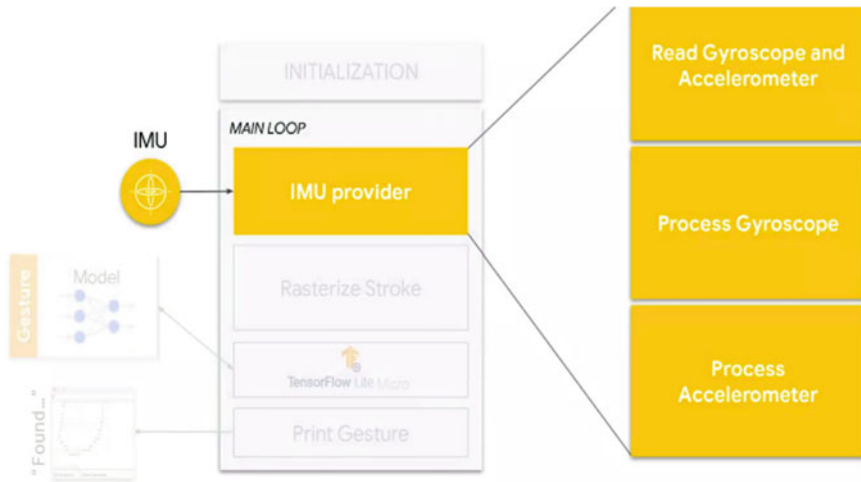


Fig. 7 Read the data of the gyroscope and accelerometer and estimate the drift of the gyroscope

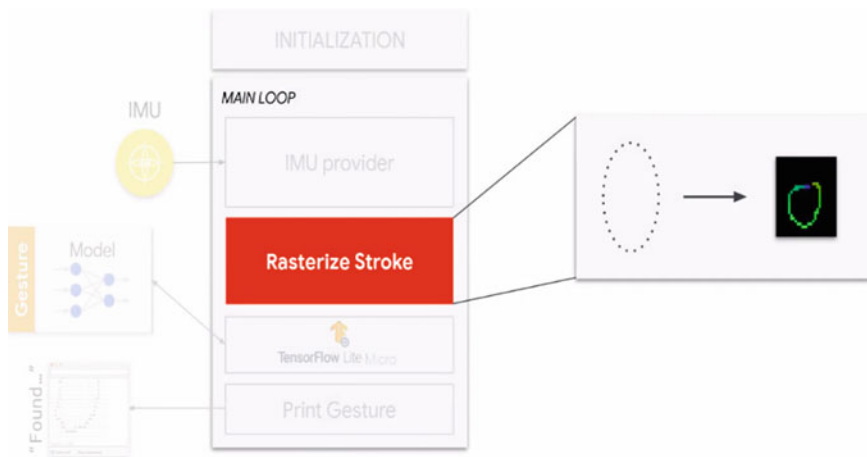


Fig. 8 Flatten the three-dimensional coordinates to two-dimensional coordinates and then rasterize that into an image

4.4 Model

After pre-processing, the next part is to hand that rasterized image directly to our convolutional neural net. In this case, we will pass in an RGB image, a red, green, and blue image. So, there are three challenges to the idea that we are giving into the net. And that input will then be run using a convolutional neural network, predicting the gesture. To invoke the model, we must set up the input buffers. Also, due to having

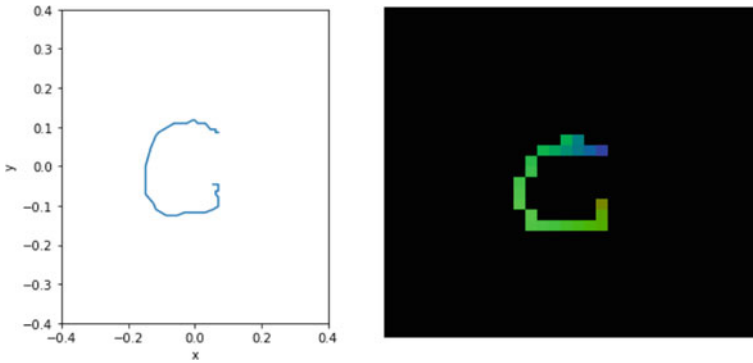


Fig. 9 Rasterization process: flatten the three-dimensional coordinates to two-dimensional and convert an image into an RGB image for building a CNN model for classification

Table 1 Reduction in size of the model to fit our need for Arduino Nano 33 BLE

Model	Size	
TensorFlow	683,299 bytes	
TensorFlow lite	98,812 bytes	(Reduced by 584,487 bytes)
TensorFlow lite quantized	30,576 bytes	(Reduced by 68,236 bytes)

memory constraints, we must quantize our model. Table 1 shows how quantization reduces the size of the model (Figs. 10, 11 and 12).

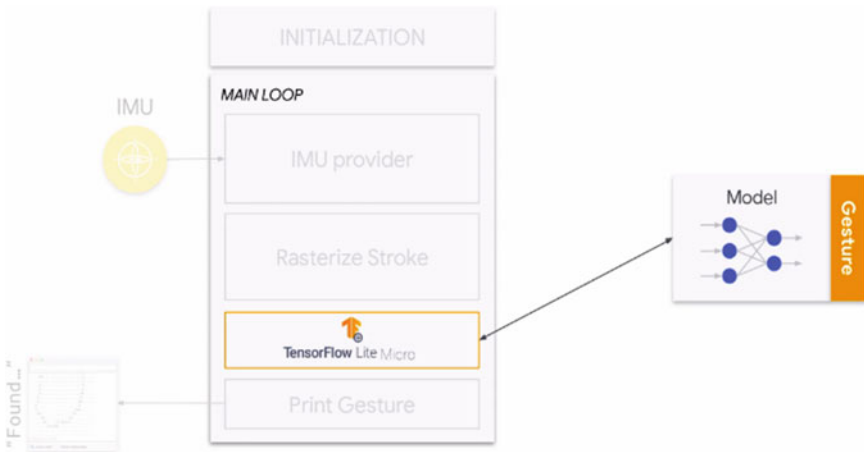
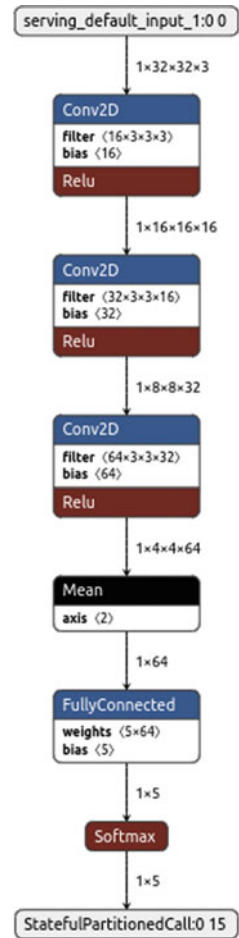


Fig. 10 Calling the TensorFlow lite micro-model for learning and classification

Fig. 11 Model flow



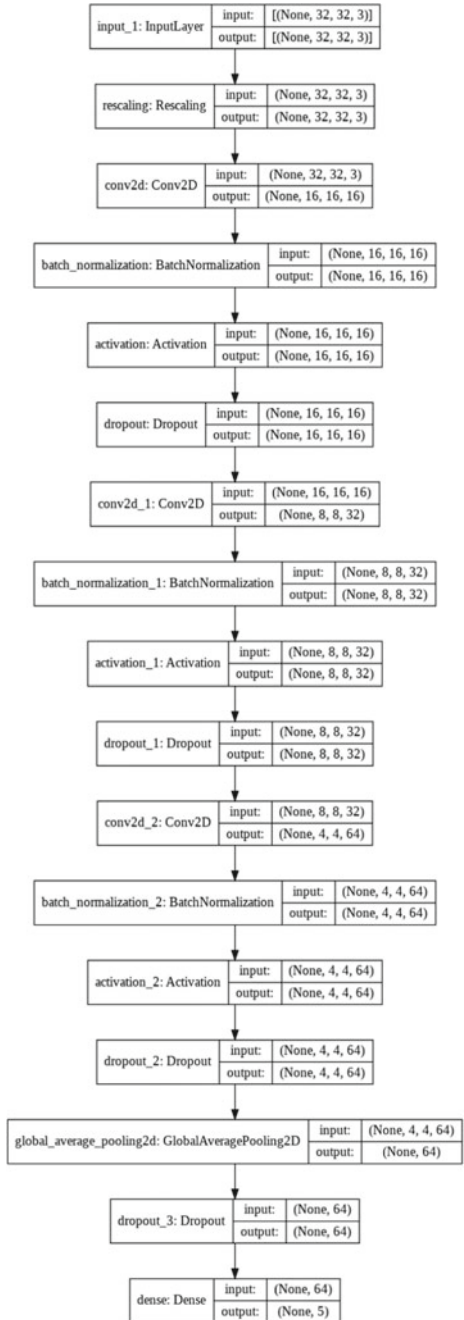
4.5 Output

We get the output from the neural network to see what it has determined as an actual gesture, and in terms of processing the result, we print the work to the screen (Fig. 13).

5 Hand Gestures Recognized

The input device for HMI must perform a wide range of operations, yet it can only recognize a limited amount of hand motions. The five gestures are depicted in Figs. 14, 15, 16, 17 and 18. We move our IMU consisting of an accelerometer and gyroscope in three-dimensional space.

Fig. 12 Complex machine learning model workflow. It uses the CNN machine learning model for classification



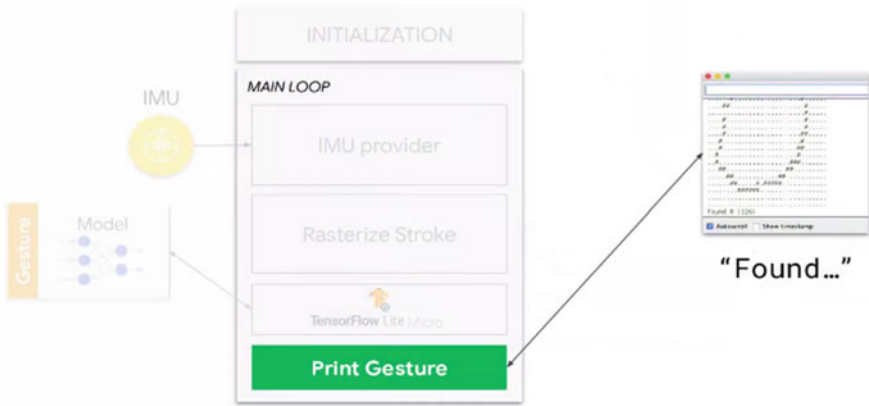


Fig. 13 Output of the gesture recognition

Fig. 14 C Alphabet



Fig. 15 L Alphabet



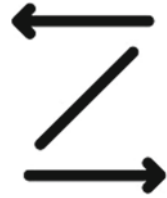
Fig. 16 I Alphabet



Fig. 17 O Alphabet



Fig. 18 Z Alphabet



6 Experimental Demonstration Gesture Recognition Device

Several critical approaches were previously discussed, such as rasterization, the model utilized for high-accuracy classification. These strategies created a varied HMI input device with easy, real-time, accurate, user-friendly, and multi-functional capabilities. These benefits were demonstrated in follow-up investigations.

Figure 19 depicts our experimental setup. The sensor system consisted of a micro-controller unit, inertial sensor IMU, and an inertial sensor system. An accelerometer, a gyroscope, and a magnetometer were all included in the inertial sensor, but only the three-axis accelerometer and three-axis gyroscope were used in this investigation. The sampling frequency was set to 25 Hz. The sensor system is constructed by mounting the microcontroller on a stick and interacting via USB. Using the BLE (Bluetooth) module, we may increase the usability.

6.1 Verification

The constructed input device is used on multiple application programs controlled by the input device to test the suggested concept—the program aimed to transition between various programs and a media player for playing video and media files.

Each experiment was carried out in a particular order. First, we tested the connectivity and operation of the input and gesture recognition device. We execute the target program and assess the functions once the device is connected correctly. The five hand gestures in Figs. 15–19 and if necessary, simple variations of five movements match the activities. After the initial setup, the first volunteer in the experiment activated the device and performed a scenario involving a series of hand gestures that fulfilled all the fundamental operations.

6.2 Verification of Media Player

The opening of the media player is the initial feature. The current playing file can be played and paused with the second function. The third option is to mute the movie

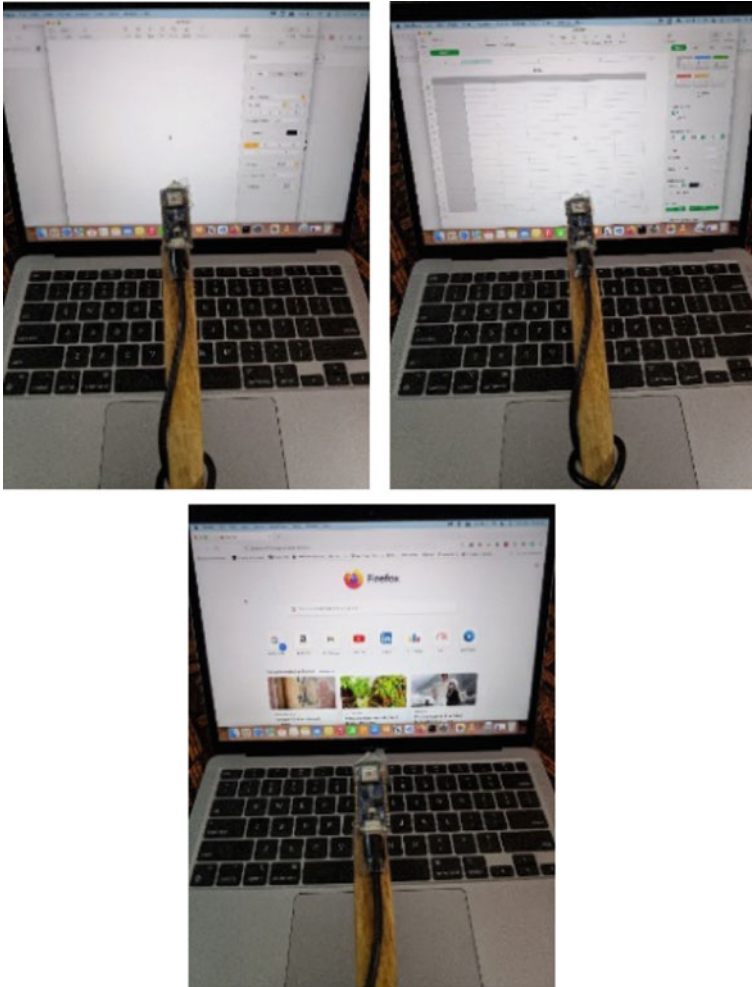


Fig. 19 Input device is used for the application program

and increase or decrease the volume. Figure 20 shows an experimental video file playback sequence. The following are the play, pause, and volume controls.

7 Conclusion

This paper proposes a sensor-based gesture recognition system that can be used as an input device for the HMI system. Five gestures are being used for multiple different applications. The same device behaves differently at some point for other



Fig. 20 Sequence of experiments uses gestures as input to a multimedia video player and plays and pauses the video

types of running on the device. If used to open an application for another application, it could mute the device, pause it, or play it. The project’s primary emphasis is on the project’s portability and fast and reliable recognition. For portability, we have used Arduino Nano 33 BLE. We use the rasterization process for fast and reliable recognition, which converts the three-dimensional spherical coordinates into two-dimensional coordinates and then rasterizes the image. This image is easy to build a highly accurate and robust model, which ensures our gesture recognition is perfect.

References

1. Pavlovic VI, Sharma R, Huang TS (1997) Visual interpretation of hand gestures for human-computer interaction: a review. *IEEE Trans Pattern Anal Mach Intell* 19:677–695
2. Zhang Z (2012) Microsoft Kinect Sensor and Its Effect. *IEEE Multimed* 19:4–10
3. Cavalieri L, Mengoni M, Ceccacci S, Germani MA (2016) Methodology to introduce gesture-based interaction into existing consumer product. In: *Proceedings of the international conference on human-computer interaction, Toronto, ON, Canada, 17–22 July 2016*; pp 25–36
4. Yang X, Sun X, Zhou D, Li Y, Liu H (2018) Towards wearable A-mode ultrasound sensing for real-time finger motion recognition. *IEEE Trans Neural Syst Rehabil Eng* 26:1199–1208
5. King K, Yoon SW, Perkins N, Najafi K (2008) Wireless MEMS inertial sensor system for golf swing dynamics. *Sens Actuators A Phys* 141:619–630
6. Luo X, Wu X, Chen L, Zhao Y, Zhang L, Li G, Hou W (2019) Synergistic myoelectrical activities of forearm muscles improving robust recognition of multi-fingered gestures. *Sensors* 19:610
7. Lee BG, Lee SM (2018) Smart wearable hand device for sign language interpretation system with sensors fusion. *IEEE Sens J* 18:1224–1232

8. Liu X, Sacks J, Zhang M, Richardson AG, Lucas TH, Van der Spiegel J (2017) The virtual trackpad: an electromyography-based, wireless, real-time, low-power, embedded hand-gesture-recognition system using an event-driven artificial neural network. *IEEE Trans Circ Syst II Exp Briefs* 64:1257–1261
9. Jiang S, Lv B, Guo W, Zhang C, Wang H, Sheng X, Shull PB (2018) Feasibility of wrist-worn, real-time hand, and surface gesture recognition via sEMG and IMU sensing. *IEEE Trans Ind Inform* 14:3376–3385
10. Pomboza-Junez G, Holgado-Terraza JA, Medina-Medina N (2019) Toward the gestural interface: a comparative analysis between touch user interfaces versus gesture-based user interfaces on mobile devices. *Univers Access Inf Soc* 18:107–126
11. Lopes J, Simão M, Mendes N, Safeea M, Afonso J, Neto P (2019) Hand/arm gesture segmentation by motion using IMU and EMG sensing. *Procedia Manuf* 11:107–113; *Sensors* 19:2562
12. Kartsch V, Benatti S, Mancini M, Magno M, Benini L (2018) Smart wearable wristband for EMG based gesture recognition powered by solar energy harvester. In: *Proceedings of the 2018 IEEE international symposium on circuits and systems (ISCAS)*, Florence, Italy, 27–30 May 2018, pp 1–5
13. Kundu AS, Mazumder O, Lenka PK, Bhaumik S (2017) Hand gesture recognition based omnidirectional wheelchair control using IMU and EMG sensors. *J Intell Robot Syst* 91:1–13
14. Tavakoli M, Benussi C, Lopes PA, Osorio LB, de Almeida AT (2018) Robust hand gesture recognition with a double channel surface EMG wearable armband and SVM classifier. *Biomed Signal Process Control* 46:121–130
15. Xie R, Cao J (2016) Accelerometer-based hand gesture recognition by neural network and similarity matching. *IEEE Sens J* 16:4537–4545
16. Deselaers T, Keysers D, Hosang J, Rowley HA (2015) GyroPen: gyroscopes for pen-input with mobile phones. *IEEE Trans Hum-Mach Syst* 45:263–271
17. Abbasi-Kesbi R, Nikfarjam A (2018) A miniature sensor system for precise hand position monitoring. *IEEE Sens J* 18:2577–2584
18. Wu Y, Chen K, Fu C (2016) Natural gesture modeling and recognition approach based on joint movements and arm orientations. *IEEE Sens J* 16:7753–7761
19. Kortier HG, Sluiter VI, Roetenberg D, Veltink PH (2014) Assessment of hand kinematics using inertial and magnetic sensors. *J Neuroeng Rehabil* 11:70
20. Jackowski A, Gebhard M, Thietje R (2018) Head motion, and head gesture-based robot control: a usability study. *IEEE Trans Neural Syst Rehabil Eng* 26:161–170
21. Zhou Q, Zhang H, Lari Z, Liu Z, El-Sheimy N (2016) Design, and implementation of foot-mounted inertial sensor-based wearable electronic device for game play application. *Sensors* 16:1752
22. Yazdi N, Ayazi F, Najafi K (1998) Micromachined inertial sensors. *Proc IEEE* 86:1640–1659
23. Yoon SW, Lee S, Najafi K (2012) vibration-induced errors in MEMS tuning fork gyroscopes. *Sens Actuators A Phys* 180:32–44
24. Xu R, Zhou S, Li WJ (2012) MEMS accelerometer based nonspecific-user hand gesture recognition. *IEEE Sens J* 12:1166–1173
25. Arsenault D, Whitehead AD (2015) Gesture recognition using Markov Systems and wearable wireless inertial sensors. *IEEE Trans Consum Electron* 61:429–437
26. Gupta HP, Chudgar HS, Mukherjee S, Dutta T, Sharma K (2016) A continuous hand gestures recognition technique for human-machine interaction using accelerometer and gyroscope sensors. *IEEE Sens J* 16:6425–6432

To Monitor Yoga Posture Without Intervention of Human Expert Using 3D Kinematic Pose Estimation Model—A Bottom-Up Approach



A. V. Navaneeth  and M. R. Dileep 

1 Introduction

In the past few decades, many researches are accomplished on yoga. As a result, few applications were developed on yoga which gives details of yoga on daily basis. Few databases are developed, which contains a collection of different types of yoga activities. In current trends, the researches on yoga have taken a different turn, where systems such as real-time posture monitoring system and analysis of human body temperatures and pressures during yoga, are developed.

1.1 Significance

The current paper mainly focuses on a real-time system which considers different aspects, while performing yoga, i.e., posture monitoring, based on comparing the gathered data with the existing data of yoga in the database.

The proposed system deals with monitoring the posture of yoga asanaas without human expert guidance while doing different steps in some asanaas, in real time. The system uses its underlying knowledge about the postures for asanaas as a comparing tool with the real-time yoga practitioner and thus monitors the posture. In summary, this system helps in smooth practicing of yoga for the practitioners without any human expert guidance.

A. V. Navaneeth (✉) · M. R. Dileep
Department of Master of Computer Applications, Nitte Meenakshi Institute of Technology,
Yelahanka, Bengaluru, Karnataka, India
e-mail: avnavaneeth25@gmail.com

1.2 *Pose Estimation Algorithm with Bottom-Up Approach for 3D Videos*

Human posture estimation goals at forecasting the postures of human body fragments and linkages in images or videos. Since posture gestures are frequently determined by some detailed human postures, the perceptive body posture of a human is acute for movement recognition.

Totally, methods for posture approximation can be gathered into bottom-up and top-down approaches. Bottom-up approach approximates body joints primarily and then clusters them to form a unique pose. Bottom-up methods were pioneered with deep cut. Top-down methods run an individual sensor primarily and approximation of body joints inside the spotted vaulting packets.

3D humanoid posture approximation is castoff to forecast the positions of body joints in 3D space. Also in the 3D posture, some approaches also improve 3D human weave from imageries or videos. This arena has attracted abundant attention in recent years; meanwhile, it is used to deliver widespread 3D structure info associated with the human body. It can be functional to numerous solicitations, such as 3D animation trades, virtual or augmented reality, and 3D action estimations. 3D humanoid posture approximation can be achieved on monocular images or videos.

Furthermost approaches habit an N-joints rigid kinematic model where a human body is characterized as an object with joints and members, comprising body kinematic construction and body shape info.

Here are three kinds of models for human body modeling:

Kinematic Model, even known as skeleton-centered model, is castoff for 2D posture approximation as sound as 3D posture approximation [9]. This stretchy and instinctual human physique classic embraces a set of mutual locations and limb alignments to characterize the human body construction. Consequently, skeleton posture approximation models are castoff to acquire the relationships among various body portions [10]. Conversely, kinematic models are restricted in demonstrating surface or outline data as shown in Fig. 1.

Planar Model, or contour-specific architecture, is used for 2D postures estimates. The planar replicas are recycled to characterize the presence and form of a human figure [11]. Typically, body fragments are embodied by several rectangles resembling the human physique delineations [12]. A prevalent illustration is the active shape model (ASM) that is recycled to acquire the complete human physique grid and the outline distortions by means of principal component analysis as shown in Fig. 1.

Volumetric model is implemented to estimate the 3D posture [13]. Here, several standard 3D human physique facsimiles recycled for deep learning built on 3D human posture estimate for mending 3D human weave exist [14, 15]. For example, GHUM and GHUML (ite) are completely trainable endways deep learning channels thought on a greater clarity dataset of complete physique probes above 60,000 human formations to archetypal arithmetical and enunciated 3D human physique form and posture as shown in Fig. 1.

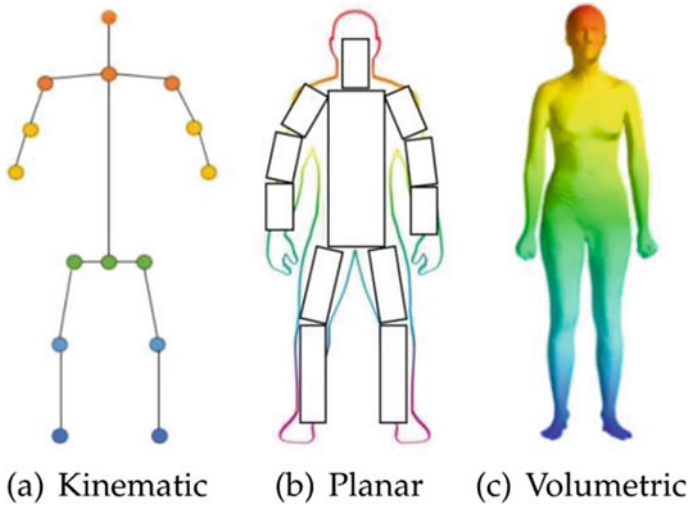


Fig. 1 Types of models for human body modeling

2 Literature Survey

Yoga is an activity which boosts the physical and mental health. Various asanas are performed in yoga. It is mandatory to practice yoga under the guidance of a yoga expert. In the absence of the expert if done, mistakes might happen that leads to physical problem. In this context, various systems are designed about posture monitoring system, yoga database, measuring the effectiveness of yoga, measuring for the right posture based on body temperature, blood pressure, etc.

Thangavelu and Mani [1] have proposed a real-time monitoring system for yoga practitioners, which monitors yoga activity. Bowyer and Kevin [2] have made a survey of approaches to three-dimensional face recognition for demonstration of ways of recognizing points in face. Dileep and Danti [3] demonstrated lines of connectivity-face model for recognition of the human facial expressions, which explains how to identify point in face for deciding different expressions. Lee et al. [4] have demonstrated a unique posture monitoring system for preventing physical illness of smartphone users, which says about negative impact of mobile phone looking posture. Muhammad Usama Islam, Hasan Mahmud, Faisal Bin Ashraf, Iqbal et al. [5] worked on yoga posture recognition by detecting human joint points in real time using Microsoft Kinect, where points are identified for detection. Patsadu et al. [6] have worked on human gesture recognition using Kinect camera, which demonstrates the type of gathering data. Obdržálek et al. [7] have proposed a real-time human pose detection and tracking for telerehabilitation in virtual reality, which signifies processing of pose detection. Lee and Nguyen [8] have demonstrated the human posture recognition using human skeleton provided by Kinect for posture recognition.

The prime focus in this paper is monitoring the posture of the body activity during performing yoga using video, by comparing with the existing data in the database, i.e., training set. The main yoga type used in this paper is Suryanamaskar, which consists of 12 different steps, and the proposed system will be built for identifying postures in it. The rest of this paper is being organized as follows: Sect. 3 describes the proposed methodology. Section 4 presents the proposed algorithm. Section 5 deals with experimental analysis and assumptions. Section 6 draws the conclusions and discussions. Finally, Sect. 7 provides the future scope of the proposed approach.

3 Proposed Methodology

The methodology of the proposed system is broadly classified as follows:

- a. **Posture capturing video:** While performing aasanaas, the data will be captured from high definition cameras fixed to the left side of the yoga practitioner. The camera captures the video and monitors for the body posture positions such as bending of arms, waist, with some angle, keeping knees straight in some positions, keeping entire body in straight position, parallel alignment of head and palms, inclination of hands, or legs. After that the captured data will be matched with the data which is stored in database for the particular aasanaa, the training set.
- b. **Offline data storing techniques in terms of templates:** The data about the aasanaa for which the system is built has to be constructed in series of steps, and the activities which are to be performed in the aasanaa have to be mentioned clearly and must be stored in some database. The database which has to be used in the proposed system must contain an arrangement, so that it supports the data for comparing with the real-time data in suitable formats. Therefore, the selection, design, and building of database must be done properly.

The particular type of yoga on which proposed system is implemented is Suryanamaskar.

3.1 Suryanamaskar

Prerequisite: In the world of yoga, Suryanamaskar is the basic level activity. Suryanamaskar should be performed on the flat floor with a yoga mat. The prerequisite posture for performing Suryanamaskar is as follows:

Stand straight on the floor; observe that both the feet are together with a reference line. By using this reference line only, the steps should be performed to follow proper method of Suryanamaskar. The sequence of steps in Suryanamaskar is shown below in Fig. 2.



Fig. 2 Sequence of steps in Suryanamaskar

Step 1: Namaskarasana

In this asana, bring both the hands in front of the chest, press the palm tightly, and keep the breathing normal.

Step 2: Urdhvasana

In this step, rise both hands upper side, with that rise the head in proportion with hands in 90degree inclination, inhale breath, and sustain the breath in that position for some seconds.

Step 3: Hastapadasana

In this step, keep both the palms in straight position on the floor with a width equal to shoulders width, note that knees should be straight, both palms and toes are in proper reference line, and exhale and sustain for few seconds.

Step 4: Ekapada Prasaranasana (right or left leg alternatively)

In this step, move one of the legs backward and land on the floor only on toes, and just place the knee of the corresponding leg on the floor, (no weight should be applied on the knee, and it should just touch the floor), and keep the breath normal.

Step 5: Dwipada Prasaranasana

In this step, move the other feet in the adjacent position to the first leg as in step 4, and keep your body in straight position like a stick parallel to the ground. Now, the entire body weight remains on palm and toes. Keep the breath normal and sustain in the same position for some seconds.

Step 6: Bhootharasana

In this step, with the help of your entire body strength, touch the entire feet to the ground by raising the hip level upward. The body shape looks like inverted "V" shape. Stay in same position for some seconds, with normal breathing.

Step 7: Saashtaangapraneepaataasana

In this step, bring your body in horizontal position with the floor, by just touching forehead, chest, and knees to floor (all the body weight should be on palm and toes) and inhale and sustain in the same position for some seconds.

Step 8: Bhujangaasana

In this step, raise the head, shoulders, and look upward, in this position waist should be near to the floor, and inhale and sustain the same position for some seconds.

Step 9: Bhoodharasana

Perform step 6.

Step 10: Ekapada Prasaranasana (right or left leg alternatively)

Perform step 4 for the corresponding leg in reverse order.

Step 11: Hastapadasana

Perform step 3 for the corresponding leg in reverse order.

Step 12: Namaskarasana

Return to Namaskarasana, i.e., step 1.

The proposed methodology of the system consists of identification of the necessary points in the human image/picture as in Fig. 3, and in between these points, a relationship should be established in terms of angles while performing aasanaa.

In this paper, the capturing will be done by left/right side view of the practitioner, so the points will be identified either in left side view or in right side view. The identification points for left side view are 1, 2, 4, 5, 7, 9, 11, 13, 15, 17, 19 and for right side view 1, 2, 3, 5, 6, 8, 10, 12, 14, 16, 18, respectively. Currently, capturing from left side of the practitioner will be done and identify the points as shown in Fig. 4.

Fig. 3 Identification of points

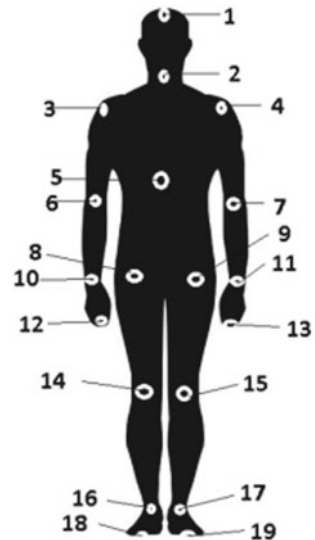


Fig. 4 Left side viewpoints

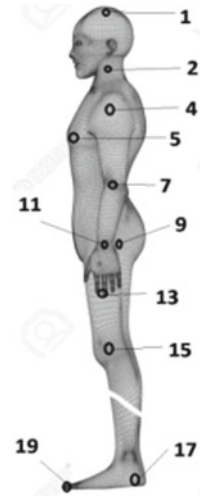
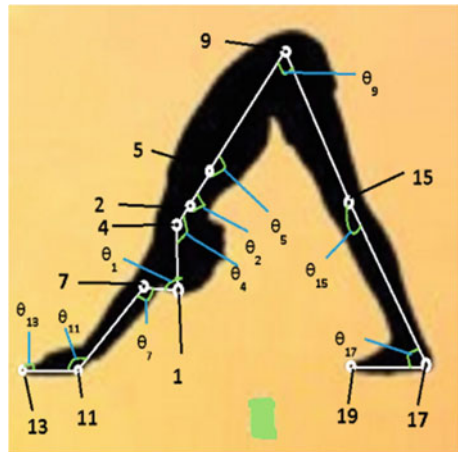


Fig. 5 Identification of angles



In the next step, the angle between various points is measured and recorded for each aasanaa. In Fig. 5, the angles are measured for **Bhujangasana** as a sample for presentation, and similarly for each aasanaa the angles will be measured and recorded. The angles are measured by using some vectors for each identified point.

4 Proposed Algorithm

Step 1: Read the input video data from both the cameras, i.e., left and front cameras and breath data from the breathing sensor fixed and store it instantaneously.

(The first reading will be taken as training set, store it in database, and repeat step 1 for testing set). The activities to be performed in the reading of the video data are identification of the points given by θn and the time spent for a particular posture in Tn , measurable in milliseconds.

Step 2: Verify the mapping between front camera data given by βF and left side camera data given by βL for the training set given by

$$\beta n = \text{MappingFunction}(\beta F, \beta L) \quad (1)$$

Step 3: Compare the data collected from front and left side camera with the training set data instantaneously, while the system completes the reading of identification points for the testing set for the video data βn . The difference in the angle between the training and the testing set is given by

$$\mu n = \text{MatchDistance} \sum_{n=0}^{n-1} (\theta n - \beta n) \quad (2)$$

Step 4: Verify the breath data for each posture given by BTn , at time intervals Tn , by comparing with training set data given by Bn , at time intervals Tn . The difference is given by notation $BTRn$, at time intervals Tn is as follows:

$$\text{BTR}n = \text{Match Distance} \sum_{n=0}^{n-1} (Bn - BTn) \quad (3)$$

Step 5: Set the allowable threshold value in terms of angle for the video data and in terms of BPM for the breath data for the successful completion of a posture. The reason to include the threshold values is because of the different sized body structures of the practitioners. The calculations can be done in the way shown as below

$$\text{For video data : } \mu n = \Theta n + 10 \quad (4)$$

Repeat step 1 to step 5 for each aasanaa.

Step 6: Calculate the result in terms of percentage of successful completion of yoga by consolidating the results for all the aasanaas, prepare report in proper format, and display in front of the practitioner.

5 Experimental Analysis and Assumptions

The experimental analysis and assumptions includes the following: The posture of the yoga practitioner will be captured in terms of video, and single camera is used to capture the video from one of the side angles that is, in this paper, left sided video

capturing is preferred. In the next stage, the proposed algorithm compares the angles from the collected vectors (testing set) with training set which is already captured in terms of video from the same proposed algorithm and which is already stored in the database.

A threshold value should be fixed for compensation for the postures in the aasanaas with some differences in angles. The threshold is termed as T , to be fixed for the training set, and it consists of the angle value for a vector for a particular aasanaa along with plus or minus values. These plus or minus values are considered to be threshold value, and the vector value for a particular aasanaa in testing set is valid when it matches with the training set plus threshold values to draw the conclusion.

6 Conclusions and Discussions

The proposed model specifies an automated system for monitoring yoga posture. The system gets the input from a camera in terms of video; in the proposed system, the camera is placed in the left side of the practitioner, from where the data is captured. This data is called testing set. The testing set has to be compared with the training set which is already stored in the database in video format. The training set consists of the angles for the aasanaas for yoga by an expert practitioner with perfect postures, and hence, this will be considered as training set. A mathematical model compares both training set and testing set in terms of vectors, and conclusion can be drawn.

Since proposed system is developing globally, by considering the various body structures of human, an attempt has been made in giving some relaxation for the angles in aasanaas; i.e., a threshold T gives the plus or minus angles of errors in performing aasanaas, the method of fixing threshold T for each aasanaa is out of the scope of this paper, and it may be considered as a future development.

7 Future Scope

In this paper, a model system for monitoring yoga posture is proposed. The proposed system uses video capturing for monitoring yoga postures. In this context, the system uses a high definition camera, which is placed in the left side of the yoga practitioner, and posture information will be gathered in terms of video by identifying some of the point from the video.

The proposed system can be further developed by placing one more high definition camera from the front side of the yoga practitioner, thereby gathering again some data about the posture, by some identification points. In this stage, the yoga monitoring system will be having two HD cameras that is one from left side and one from the front side. The necessity of the front side camera is to ensure the balanced structure of the body while performing some particular yoga aasanaa. In the next level, an algorithm

compares both the data, which are collected by front and side angle cameras with some constraints and thresholds, and conclusion will be drawn.

References

1. Thangavelu A, Mani P (2017) A real time monitoring system for yoga practitioners. *Int J Intell Eng Syst*
2. Bowyer KW (2004) A survey of approaches to three-dimensional face recognition, *Pattern Recogn* 1:358–361
3. Dileep MR, Danti A (2013) Lines of connectivity-face model for recognition of the human facial expressions. *Int J Artif Intell Mechatron* 2(2):2320–5121
4. Lee H, Lee S, Choi YS (2013) A new posture monitoring system for preventing physical illness of smartphone users. In: 10th annual IEEE CCNC
5. Islam MU, Mahmud H, Ashraf FB, Hossain I, Hasan MK (2017) Yoga posture recognition by detecting human joint points in real time using Microsoft Kinect. *UTC from IEEE Xplore*
6. Patsadu O, Nukoolkit C, Watanapa B (2012) Human gesture recognition using Kinect camera. In: 2012 International joint conference on computer science and software engineering (JCSSE). IEEE, 2012, pp 28–32
7. Obdržálek Š, Kurillo G, Han J, Abresch T, Bajcsy R (2012) Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. *Stud Health Technol Inf* 173:320–324
8. Le TL, Nguyen MQ (2013) Human posture recognition using human skeleton provided by Kinect. In: 2013 International conference on computing, management and telecommunications (ComManTel). IEEE, 2013, pp 340–345
9. Yuan Y, Wei SE, Simon T, Kitani K, Saragih J (2021) Simpose: simulated character control for 3d human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7159–7169
10. Zhou X, Sun X, Zhang W, Liang S, Wei Y (2016) Deep kinematic pose regression. In: European conference on computer vision. Springer, Cham, pp 186–201
11. Mondragón IF, Campoy P, Martínez C, Olivares-Méndez MA (2010) 3D pose estimation based on planar object tracking for UAVs control. In: 2010 IEEE international conference on robotics and automation, pp 35–41. IEEE, 2010
12. Frohlich R, Tamas L, Kato Z (2019) Absolute pose estimation of central cameras using planar regions. *IEEE Trans Pattern Anal Mach Intell* 43(2):377–391
13. Fabbri M, Lanzi F, Calderara S, Alletto S, Cucchiara R (2020) Compressed volumetric heatmaps for multi-person 3d pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7204–7213
14. Gessert N, Schlüter M, Schlaefer A (2018) A deep learning approach for pose estimation from volumetric OCT data. *Med Image Anal* 46:162–179
15. Li Y, Wang Y, Case M, Chang SF, Allen PK (2014) Real-time pose estimation of deformable objects using a volumetric approach. In: 2014 IEEE/RSJ international conference on intelligent robots and systems, pp 1046–1052. IEEE

Upshot and Disparity of AI Allied Approaches Over Customary Techniques of Assessment on Chess—An Observation



A. V. Navaneeth  and M. R. Dileep 

1 Introduction

Chess is one of the few arts where the composition of the tactics should be applied appropriately for achieving best performance. The best human chess players implement several tactics, strategies, and plans towards achieving best outcomes. But with computer chess engine the case is different, the thought process of computer completely dependent on calculating all the possible moves and choosing best which is suitable. But this approach has its own limitations in terms of size, speed, dimension of the data, and depth of the search. To compensate this problem, computer scientist came up with different approaches so that the accuracy of the system is achieved without diminishing the speed. The upcoming subsections demonstrate few such techniques which are implemented in traditional and modern approaches of evaluation.

In the traditional approach, the chess engine considered is Stockfish. The evaluation methodology followed by the engine involves two famous methodologies that is Minimax algorithm and Alpha–Beta pruning algorithm.

1.1 Minimax Algorithm

Minimax is a backtracking technique that is castoff in judgement building to find the ideal move for the performer, considering the challenger also performs optimally. The

A. V. Navaneeth (✉) · M. R. Dileep
Department of Master of Computer Applications, Nitte Meenakshi Institute of Technology,
Yelahanka, Bengaluru, Karnataka, India
e-mail: avnavaneeth25@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_12

127

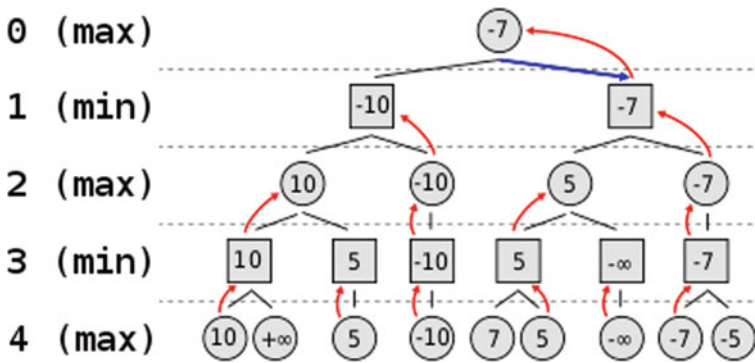


Fig. 1 Minimax algorithm

terms considered for the players in the algorithms are termed as minimizer and maximizer. The maximizer attempts to acquire the maximum mark promising, whereas the minimizer attempts to fix the conflicting and become the deepest mark probable. Each board state obligates a value related with it. At some position if the maximizer has higher indicator then, the mark then positive value will be assigned to the board position. In case if minimizer is higher indicator, then it will result in negative value assignment. Some heuristic techniques are used to perform these calculations in the game. The working principle of Minimax algorithm is as depicted in Fig. 1, with an example.

1.2 Alpha–Beta Pruning Algorithm

Alpha–Beta pruning is an optimization method. It decreases the calculation interval tremendously. The algorithm allows searching considerably faster and even going into profound stages in the game tree. It avoids branch search in the game tree which by deciding it as unfeasible solution. Two more extra parameters are added to the minimax algorithm, and hence, the algorithm is termed as Alpha–Beta pruning algorithm. Alpha represents the best value for the maximizer at respective level or above. Beta represents the best value for the minimizer at respective level or above. The working of the algorithm with an example is as specified in Fig. 2.

1.3 Monte Carlo Tree Search (MCTS)

In the modern approach, the chess engine considered is AlphaZero. The evaluation methodology followed by the engine involves famous methodology that is Monte Carlo Tree Search (MCTS) algorithm.

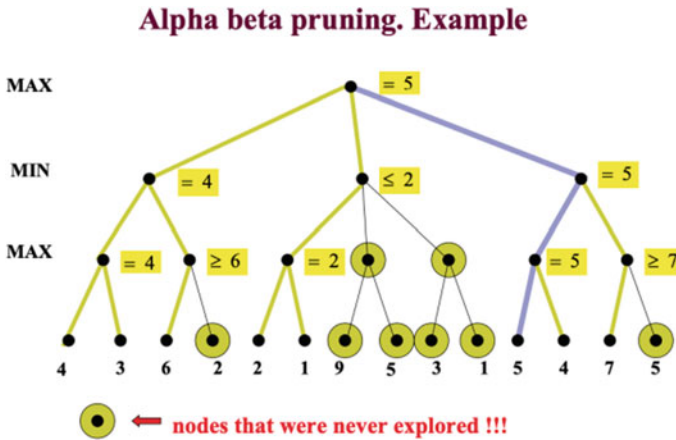


Fig. 2 Alpha-Beta pruning algorithm

Monte Carlo Tree Search (MCTS) is a method in the arena of artificial intelligence (AI). It is a probabilistic and exploratory determined search algorithm that syndicates the typical tree exploration applications together with machine learning ideologies of reinforcement knowledge n tree search, and there’s always the probability that the existing finest action is really not the maximum ideal action. In such cases, MCTS algorithm becomes valuable as it endures to estimate extra substitutes occasionally throughout the learning stage by performing them, instead of the existing apparent ideal policy. This is identified as the “exploration-exploitation trade-off”. It exploits the activities and policies that is originate to be the finest till now but also essential to endure to discover the native space of substitute choices and catch out if they can substitute the existing paramount. Various steps in Monte Carlo Tree Search algorithms are selection, expansion, simulation, and backpropagation.

Selection: The MCTS algorithm navigates the up-to-date tree from the root node by means of a precise policy. The policy habits an assessment utility to pick the nodes with the maximum assessed significance. MCTS habits the Upper Confidence Bound (UCB) principle implemented to trees as the policy in the selection procedure to navigate the tree. It equilibrums the exploration-exploitation trade-off.

Expansion: In this procedure, a fresh child node is supplemented to the tree to that node which was optimally extended throughout the selection procedure.

Simulation: In this method, a simulation is accomplished by selecting moves or policies till a consequence or predefined condition is attained.

Backpropagation: Subsequently defining the assessment of the freshly supplemented node, the residual tree need be restructured. So, the backpropagation method is accomplished, where it backpropagates from the fresh node to the root node. In the progression, the number of simulation deposited in each node is incremented. If

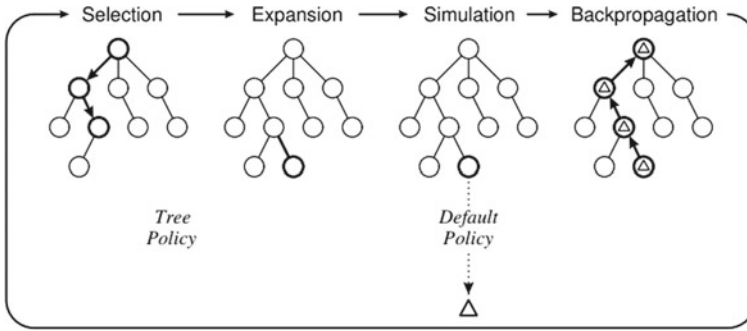


Fig. 3 Monte Carlo tree search (MCTS)

the fresh node's simulation consequences in a victory, then the number of victories is as well incremented. The working of MCTS is as shown in Fig. 3.

2 Literature Survey

From the ancient times, chess dragged the interest of the mathematicians because of its enormous set of possibilities and combinations. Several works has been done in the past towards development of efficient chess engines. During 1984, Google's DeepMind was the first officially rated chess engine to show superhuman capabilities by defeating the best human grandmaster. In the subsequent years, tremendous research work has been done towards the development of chess engines by incorporating scientific and mathematical approaches. As a result, many efficient chess engines are incubated to perform in the real world. The algorithms which are developed for the chess engines are also capable of solving other real world problems other than chess which mainly contains huge set of combinatorial and complex data, and the decisions are drawn on the basis of random factors.

Browne et al. [1] has conducted a survey on Monte Carlo Tree Search methods. Carlsson et al. [2] has demonstrated the process involved in AlphaZero to alpha hero: A pre-study on additional tree sampling within self-play reinforcement learning. Chan et al. [3] has made a detailed study on Theory and applications of Monte Carlo simulations. Dehghani et al [4] has made cumulative study on GA-based method for search-space reduction of chess game tree. Fuller et al. [5] has made the detailed analysis of the Alpha-Beta pruning algorithm. Gao et al. [6] has made a detailed survey on efficiently of Mastering the Game of NoGo with Deep Reinforcement Learning Supported by Domain Knowledge. Johnson et al. [7] has made an holistic survey on new family of probability distributions with applications to Monte Carlo studies. Kasparov [8] has made deep analysis on chess: a Drosophila of reasoning, which is

composed of advanced evaluation and deep calculation towards positional possibilities. Lai et al. [9] has demonstrated a model by name Giraffe: using deep reinforcement learning to play chess. Maesumi et al. [10] has demonstrated the strategies of playing chess with Limited Look Ahead. McGrath et al [11] articulated a survey on Acquisition of chess knowledge in AlphaZero. Moerland et al [12] has made a detailed study on A0c: AlphaZero in continuous action space. Mordechai et al [13] has made a detailed study on applications of Monte Carlo method in science and engineering. Motohiro et al. [14] demonstrated various applications of Monte Carlo simulation in the analysis of a sputter-deposition process. Pearl et al [15] has made extensive research on the solution for the branching factor of the Alpha-Beta pruning algorithm and its optimality Silver et al [16] described a summary on general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Stockman et al [17] has demonstrated how minimax algorithm better than Alpha-Beta. Vardi et al [18] has made an analysis on new minimax algorithm. Wang et al [19] has made a survey on Adaptive Warm-Start MCTS in AlphaZero-like Deep Reinforcement Learning. Wang et al [20] has worked on Tackling Morpion Solitaire with AlphaZero-like Ranked Reward Reinforcement Learning.

In this paper, a comparative study has been made between two famous chess engines by considering some specific parameters like depth of search, number of move per unit time, number of positions analyses per unit time, first move advantage, depth analysis with varying CPU, GPU, and TPU sizes, thereby to showcase how the incorporation of artificial intelligence has changed the perspective of evaluation process involved in the game and towards development of the state-of-the-art engines.

3 Comparative Analysis

Since both Stockfish and AlphaZero chess engines are built on different evaluation methods, the process of making comparative analysis also differs. Some of the common parameters considered for comparison are search depth, time complexity, and accuracy towards making the best move.

3.1 Stockfish

The performance analysis of the Stockfish chess engine can be visualized in several perceptions. Table 1 depicts the searching depth of the evaluation method used in the Stockfish where search will be performed on the game tree created during the game.

In Table 1, the search for the best move was experimented at certain stage of the game with varying depths, and for different set of CPU which show diversity in playing strength under certain time constraints.

Table 2 represents the playing capability of the Stockfish when executed on single CPU, 4 CPUs, and 8 CPUs. The games played were 100 among each set of CPUs

Table 1 Searching depth analysis of Stockfish

Number of CPU's used	Depth of the search	Nodes visited per second	Time incurred in visiting all the nodes (in min)	Nodes evaluated
1	40	1,446,372	14:16	1,323,070,320
4	40	4,544,046	15:16	4,526,190,013
8	40	7,345,567	18:45	6,855,124,670
1	42	1,517,110	14:49	1,411,851,512
4	42	4,932,887	15:57	4,526,190,013
8	42	7,811,970	19:50	7,232,217,120

Table 2 Performance analysis under fixed depth varying capacity of Stockfish

S. No.	Number of CPU's used (depth 42)	Wins	Losses	Draws	No of games played	Points acquired
1	8	54	0	46	100	77
2	4	13	24	63	100	44.5
3	1	1	44	55	100	28.5

Table 3 Stockfish performance analysis based on mode of play

S. No.	Chess engine	Played as	No. of games played	Wins	Losses	Draws
1	Stockfish	White	100	52	8	40
2	Stockfish	Black	100	43	13	44

with a fixed depth of 42, and it is observed that the performance was greatly increased with number of processors. Each win is equal to 1 point, loss equals 0 point, and draw signifies 0.5 point.

Another most important dimension of performance analysis of Stockfish is the mode of play. The pre-requisites in this case are fixed depth, fixed no of CPUs, fixed/varying time, and fixed opponent. A remarkable set of performance difference is observed, when the engine is compelled to play in different modes as given in Table 3.

3.2 AlphaZero

AlphaZero searches just 80,000 positions per second in chess compared to 70 million for Stockfish. AlphaZero recompenses for the subordinate amount of estimations by means of its deep neural network to emphasis abundant extra selectively on the utmost auspicious variant. It was taught expending 5000 tensor processing units (TPUs), and a 44-core CPU in its matches as given in Table 4.

Table 4 Searching depth analysis of AlphaZero

Number of TPU/CPU's used	Number of positions per second
5000-TPU	80,000
44 Core CPU	75,000

Table 5 AlphaZero performance analysis based on mode of play

S. No.	Chess engines played	Played as	No. of games played	Time control	Wins	Losses	Draws
1	AlphaZero versus Stockfish	White/black	100	–	25-white 3-black	0	72
2	AlphaZero versus Stockfish	White/black	1200	–	290	24	886
3	AlphaZero versus Stockfish	White/black	1000	3 h (15 s/move)	155	6	839

Each engine was assigned exactly 60 s in AlphaZero’s chess competition against. Throughout the competition, AlphaZero executed on a mono host along with four programme-oriented TPUs. Among 100 meets beginning from the regular preliminary point, AlphaZero accomplished 25 meets as White, accomplished 3 as Black, and drew the rest 72. In a sequence of twelve, 100 meets of indefinite period or source restrictions, contrary to Stockfish beginning since 12 best prevalent humanoid starts, AlphaZero conquered 290, drew 886 and got defeated in 24 games as given in Table 5.

4 Observations and Discussions

As per the study done on two famous chess engines, the following observations are listed as following. With the increase in processing capacity of the computer, the performances of both the engines were enhanced greatly. As per the depth levels, Stockfish has achieved greater performance with an exponential raise in the node analysed versus node visited towards finding the best solution from the game tree. However, AlphaZero purely relayed on the logical perspective of finding the best solution by partially considering the depths of the tree, which basically depends on the randomization process involved within. Coming to the first move advantage, both the engines are able to achieve superior win, loss, and draw ratios. Finally, when both the engines made to play against each other under standard environments, it is observed that AlphaZero was the strongest as shown in Fig. 4.

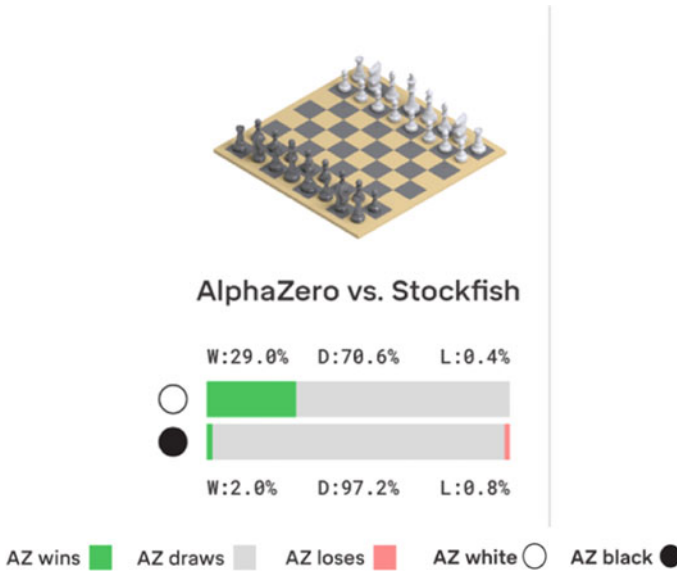


Fig. 4 Performance analysis AlphaZero versus Stockfish

5 Conclusion

According to comparative analysis made, it is observed that, with the introduction of artificial intelligence and related technologies in chess which was thrown complex structural combinations and remarkable set of possibilities, the most feasible and appropriate paths were chosen towards finding the best solution. It is also observed that with the increase in speed and capacity of the processor, a remarkable difference in the performance related issues were achieved. In overall, artificial intelligence has brought tremendous changes in the evaluation methodology even for the complex problems like chess thereby achieving increased outcomes.

6 Future Enhancement

The approaches discussed in the paper are not only limited to the context of the subject conferred, but can be extended also to the fields where high dimensions of data with complex combinations arises, and the problems where randomization is required. The fields where such kind of problems arises can be seen are as following. First in the field of stock marketing, where continuous variations arises over commodities, and collectively if need to analyse them collectively may give raise to the complex structures. Second field is medical, where protein structure can be studied based on some structural parameters where there is a chance high size and dimension of the

data may arise. Third is in the field of image processing in image recovery since high dimensions of data can be expected from the concerned images.

References

1. Browne CB, Powley E, Whitehouse D, Lucas SM, Cowling PI, Rohlfshagen P, Tavener S, Perez D, Samothrakis S, Colton S (2012) A survey of Monte Carlo tree search methods. *IEEE Trans Comput Intell AI Games* 4(1):1–43
2. Carlsson F, Öhman J (2019) Alphazero to alpha hero: a pre-study on additional tree sampling within self-play reinforcement learning
3. Chan WKV (ed) (2013) Theory and applications of monte Carlo simulations. BoD–books on demand
4. Deghani H, Babamir SM (2017) A GA based method for search-space reduction of chess game-tree. *Appl Intell* 47(3):752–768
5. Fuller SH, Gaschnig JG, Gillogly JJ (1973) Analysis of the alpha-beta pruning algorithm. Department of Computer Science, Carnegie-Mellon University
6. Gao Y, Lezhou W (2021) Efficiently mastering the game of NoGo with deep reinforcement learning supported by domain knowledge. *Electronics* 10(13):1533
7. Johnson ME, Tietjen GL, Beckman RJ (1980) A new family of probability distributions with applications to Monte Carlo studies. *J Am Stat Assoc* 75(370):276–279
8. Kasparov G (2018) Chess, a drosophila of reasoning 362:1087–1087
9. Lai M (2015) Giraffe: using deep reinforcement learning to play chess. arXiv preprint [arXiv:1509.01549](https://arxiv.org/abs/1509.01549)
10. Maesumi A (2020) Playing chess with limited look ahead. arXiv preprint [arXiv:2007.02130](https://arxiv.org/abs/2007.02130)
11. McGrath T, Kapishnikov A, Tomašev N, Pearce A, Hassabis D, Kim B, Paquet U, Kramnik V (2021) Acquisition of chess knowledge in AlphaZero. arXiv preprint [arXiv:2111.09259](https://arxiv.org/abs/2111.09259)
12. Moerland TM, Broekens J, Plaat A, Jonker CM (2018) A0c: AlphaZero in continuous action space. arXiv preprint [arXiv:1805.09613](https://arxiv.org/abs/1805.09613)
13. Mordechai S (2011) Applications of Monte Carlo method in science and engineering
14. Motohiro T (1986) Applications of Monte Carlo simulation in the analysis of a sputter-deposition process. *J Vac Sci Technol A Vac Surf Films* 4(2):189–195
15. Pearl J (1982) The solution for the branching factor of the alpha–beta pruning algorithm and its optimality. *Commun ACM* 25(8):559–564
16. Silver D, Hubert T, Schrittwieser J, Antonoglou I, Lai M, Guez A, Lanctot M et al (2018) A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* 362(6419):1140–1144
17. Stockman GC (1979) A minimax algorithm better than alpha–beta? *Artif Intell* 12(2):179–196
18. Vardi A (1992) New minimax algorithm. *J Optim Theory Appl* 75(3):613–634
19. Wang H, Preuss M, Plaat A (2021) Adaptive warm-start MCTS in AlphaZero-like deep reinforcement learning. arXiv preprint [arXiv:2105.06136](https://arxiv.org/abs/2105.06136)
20. Wang H, Preuss M, Emmerich M, Plaat A (2020) Tackling Morpion solitaire with AlphaZero-like ranked reward reinforcement learning. In: 2020 22nd international symposium on symbolic and numeric algorithms for scientific computing (SYNAS). IEEE, pp 149–152

Network Intrusion Detection Using Neural Network Techniques



**B. A. Manjunatha, Aditya Shastry, G. Nishchala, N. Pavithra,
and S. Raheela Banu**

1 Introduction

With the advent of technology, network security has become a major concern in today's society. The network intrusion detection system (NIDS) observes the data traffic for distrustful activity and issues alarms when such activity is detected. Any hazardous activity is usually reported to the supervisor or collected centrally using security information and event management system (SIEM). Using the SIEM framework, one can detect dangerous activity from false alarms by integrating output from a variety of sources. Even though intrusion detection systems monitor networks for potentially harmful activity, false alarms might be generated as well. Henceforth, IDS products should be adjusted when first launched for organizations. It means appropriately setting up the intrusion system to determine what normal traffic on the network is similar to when compared to dangerous activity.

A. Classification of Intrusion Detection System:

Network Intrusion Detection System (NIDS):

Monitoring all network traffic with network intrusion detection systems (NIDS) is set up at a fixed location within the network.

B. A. Manjunatha (✉) · A. Shastry · G. Nishchala · N. Pavithra · S. Raheela Banu
Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology,
Bangalore, India
e-mail: manjunatha.ba@nmit.ac.in

A. Shastry
e-mail: Adityashstry.k@nmit.ac.in

Host Intrusion Detection System (HIDS):

Host intrusion detection systems (HIDS) work on independent host gadgets (devices) on the network. Inbound and outbound packages are detected by HIDS and issue a warning to the regulator if any dangerous action is suspected. It takes a depiction of existing system records and collates them to a previous summary. If the system examination records have been altered or erased, a warning is sent to the chairman for examination. A picture of the utilization of HIDS can be found in the main machines, which can be depended upon to change its construction.

Protocol-based Intrusion Detection System (PIDS):

In protocol-based intrusion detection (PIDS) system, the specialist lives toward the end of the server, controlling and interpreting the convention/protocol between the server and the device/client. It attempts to track down a web server by routinely checking the HTTPS convention stream and getting the connected HTTP protocol.

Application Protocol-based Intrusion Detection System (APIDS):

Application protocol-based intrusion detection system is a framework or specialist that dwells inside a gathering of servers that gets impedence by interpreting and making an interpretation of interchanges into application-explicit protocols. For instance, this will screen the SQL-explicit middleware protocol as it collaborates with the web server.

Hybrid Intrusion Detection System:

Integrating two or more access systems triggers a crossover access system. Hybrid access system incorporates facilitating specialist or system information and organization data to make a total image of the organization system. They are more effective than other section level systems. Introduction is a Hybrid IDS.

B. *Detection Method of Intrusion Detection System:*

Signature-based Method:

Signature-based IDS identifies design put together assaults with respect to arrange traffic like number bytes or number 1 or 0. It can likewise identify assaults dependent on malignant orders utilized by malware. IDS marks designs are identified by the framework.

Signature-based IDS can identify a current example (signature) assault as of now inside the framework while it is truly challenging to recognize new malware assaults as their examples (signatures) are obscure.

Anomaly-based Method:

To identify unknown malware attacks as the new malware is quickly creating, a mysterious ID has been presented. The anomaly-based IDS utilizes AI to make a dependable work model, and anything that comes in is contrasted with that model, and assuming it does not fit into the model, it is depicted as suspicious. Contrasted with the

signature-based IDS, the machine-based methodology gives better material as these models can be adjusted relying upon the applications and equipment arrangement.

2 Related Work

In this paper [1], the model of NID (network intrusion detection) is proposed based on the CNN-IDS. Using a variety of size reduction techniques, irrelevant features for traffic data network are removed first. Then, the dimensions of reduction data size are automatically extracted using convolutional neural network, and to obtain the most effective information to identify interference data is extracted by supervised learning. To minimize computational costs, we converted the first vector traffic format into an image format and used standard dataset, i.e., KDD-CUP99 to evaluate performance of the CNN model. Test outcomes suggest that AC, FAR and time-keeping convolution neural network intrusion system models are better than conventional algorithms. Thus, proposed model is not only of research value but also has practical values.

In this paper [2], network traffic models are in series of time, especially the Transmission Control Protocol or Internet Protocol (TCP/IP) for a particular period with supervised learning methods such as multilayer perceptron (MLP), CNN, CNN-recurrent neural network (CNN-RNN), a long-term memory for CNN (CNN-LSTM) and the CNN-gated recurrent unit (GRU), using millions of well-known network connections and poor network connections. To measure the effectiveness of the above methods, we need to test on dataset such as KDD-CUP99. Select complete network configuration, complete test for various MLP, CNN, CNN-RNN, CNN-LSTM and CNN-GRU and titles, using network settings and parameters. Models in each test are used up to 1000 times for obtaining level information in the range [0.01–05]. Compared to the classic machine learning classes, CNN and its distinctive architecture have performed very well. This is due to the fact that CNN is responsible for issuing high-quality feature presentations that represent the type of low-level network connection features.

As we know, deep learning is the most modern technology which automatically removes features from the samples [3]. The accuracy of intrusion detection is not high on traditional machine technology considering this fact; the paper proposes a network intrusion model-based CNN algorithms. The model can extract the powerful features of input samples, so the input samples can be in the correct order. The test results in the dataset KDD99 show that the model which is been proposed can significantly enhance/improve the accuracy rate of the intrusion detection model.

The author proposed a novel NIDS system which is based on CNN (convolutional neural network) [4]. We train deep learning-based detection models using both extracted features and original network traffic. They conducted comprehensive experiments using well-known benchmark datasets. The results verify the effectiveness of our system and also demonstrate the model trained through raw traffic has better than the model trained using extracted features.

Search based on network threats from attackers has been increased [5], and also, system security has become increasingly important, because the number of devices connected to the Internet is growing up. And there are common attacks, such as the DDoS (Distributed Denial of Service) that caused widespread damage for the companies. A new ID (network intrusion system) is proposed which is based on Tree-CNN algorithm with SRS (Soft-Root-Sign) feature. The test results show that hierarchical model been proposed achieves a certain reduction in kill time, by about 36%, with an average detection accuracy of 0.98 taking into account all attacks analyzed.

Main purpose to propose this paper [6] is to improve the Internet security on IDS (intrusion detection system) which is based on CNN. The IDS model proposed is intended to detect network intrusion by separating package traffic into a network such as benign or malicious training. A detailed study of the proposed model referred to nine different classifiers has been presented.

In this paper [7], we propose a novel intrusion detection method based on the adaptive synthetic sampling algorithm (ADASYN) and the convolutional neural network (CNN), to develop complete IDS capabilities and strengthen network security. First, we use the ADASYN method to stabilize the sample distribution which can effectively prevent the model from becoming resistant to large samples and forgetting small samples. Second, enhanced CNN is based on the split convolution module (SPC-CNN), which can enhance a variety of features and eliminate the impact of unnecessary data between channels training models. Then, the AS-CNN model mixed with ADASYN and SPC-CNN is used for entry-level operations. At last, a standard dataset NSL-KDD was chosen for AS-CNN testing. The simulation shows an accuracy of 4.60% and a pair of 0.79% is better compared to traditional CNN and RNN models, with the detection rate (DR) increasing with 11.34% and 10.27%, respectively.

In many applications, the skill of the deep learning methods is proven to be superior to the old techniques [8]. Similarly, this network research focuses on intrusion detection and use of LeNet-5-based convolutional neural networks (CNNs) to detect network threats. Tests show that accuracy of IDS goes up to 99.6% with more than thousand samples. The average accuracy is 97.53%.

Based on the convolution neural network [9], two layers of convolution and pooling layers have been used; a batch normalization layer is introduced for each convolution layer to enhance community distance and speed from the collapse mode. During this test, Adam optimizers and SGD were used to train the model, respectively. By this, we come to know that Adam optimizer has high performance. When epoch = 200, model accuracy can reach 0.9507, and the average $F1$ value can reach 0.9438.

In this paper [10], a network detection structure has been proposed based on examining different algorithms such as NB and XGBoost and then applied the SSA as the FS technique. Here in this proposed model, most applicable and best features are selected by applying the SSA which increases the model performance. A sturdy Anomaly Network Detection model was built using the SSA-XGBoost and SSA-NB classification algorithms. Here to examine the model performance, NSL-KDD and

UNSW_NB15 primary datasets were used. A high accuracy and performance were achieved by our proposed network detection model with a high detection rate, low false alarms, and its effectiveness. Further, the detection rate could be increased in the further work by using various other methods with the unlike datasets and prevail over the difficulties proposed by data imbalance in order to improve the model efficiency.

In the given paper [11], a wireless network intrusion detection model which depends on the improved convolutional neural network [CNN] has been proposed. Training and testing experiments were done in IBWNIDM with the help of the training and test sets of data which was pre-processed. The experiments stated with a high true positive rates and low false positive rates of network intrusion detection of IBWNIDM.

This paper [12] has proposed network intrusion detection based on CNN which is designed with a combination of SMOTE-ENN. The model is adaptive to different data environments as the CNN has the feature which selects the features automatically. Minority samples are synthesized by applying the algorithm SMOTE-ENN. The accuracy rate is 83.31%. The detection rate has gradually increased in particular from 26 to 77%. Here, it can be concluded that this proposed CNN model best suits for imbalanced network traffic of network intrusion detection system.

3 Challenges and Issues

A. *Feature Extraction*

Feature extraction has been pre-requisite for an efficient working of an intrusion detection system. It aims to decrease the number of resources required to report a large set of data. Performance of the model will be majorly affected when the features are selected inappropriately.

B. *Classifier Construction*

As it is extremely tough to find new attacks by only training on limited audit data, the classification precision of the majorly existing models needs to be enhanced. As the modeling of normal patterns is difficult and normally the false alarm rates are huge but can detect novel attacks. Hence, the classifier construction for an intrusion detection based on machine learning remains as other technical challenge.

C. *The False Positive Rate*

When the IDS detects an activity as an attack but also accepts the behavior of an activity, it can be stated as false positive state. And the most serious or dangerous state can be the false negative state. It is the activity which is actually an attack but the IDS detects it as acceptable. It has been calculated that up to 99% of alerts stated by IDSs are not related to security issues.

4 Comparison Table

Author	Year of publishing	Methods	Dataset /software	Challenges and issues
Yihan Xiao, Cheng Xing, Taining Zhang, and Zhongkai Zhao	2019	DL algorithm, principle component analysis (PCA) and auto-encoding	CICIDS2017	Low recognition rate and trouble in learning highlights in few assault classifications (U2R, R2L)
Vinayakumar R, Soman KP and Prabakaran Poornachandran	2017	Hybrid network, hyperparameter identification and TensorFlow	KDD-CUP99	The major issue observed during training was the high computational cost associated with complex architecture. As a result, they were unable to train more complex architecture
Riaz Ullah Khan, Xiaosong Zhang, Mamoun Alazab, Rajesh Kumar	2019	Pooling algorithm and a strategy consolidating convolutional neural network algorithm, softmax algorithm	KDD-CUP99	The challenge is to work on the precision of detection of attacks and to work on the adequacy of detection systems
Lin Chen, Xiaoyun Kuang, Aidong Xu, Siliang Suo, Yiwei Yang	2020	Feature selection, SVM	KDD-CUP99	The challenge is to train the model using raw traffic than using extracted features. Model trained using raw traffic is more accurate
Robson V. Mendonça, Arthur A. M. Teodoro, Renata L. Rosa, Muhammad Saadi, Dick Carrillo Melgarejo, Pedro H. J. Nardelli, and Demóstenes Z. Rodríguez	2021	A hierarchical algorithm based on soft-root-sign (SRS) activation function and Tree-CNN is presented	CICIDS2017	Examination of the experimental information shows that the proposed progressive model decreases the performance time by around 36% and accomplishes a precision of 0.98 acquisitions while thinking about completely analyzed attacks
Samson Ho, Saleh Al Jufout, Khalil Dajani, and Mohammad Mozumdar	2021	Convolutional neural network (CNN)	CICIDS2017	Upgrades can in any case be made to the exactness of the proposed IDS model when cyberattacks are identified. This is for the most part because of the constant improvement of neural network guideline

(continued)

(continued)

Author	Year of publishing	Methods	Dataset /software	Challenges and issues
Zhiqian Hu, Liejun Wang, Lei Qi, Yongming Li, and Wenzhong Yang	2020	Improved CNN and AS-CNN models using adaptive synthetic sampling (ADASYN)	NSL-KDD	Existing algorithms accessible for CNN-based access neglected to resolve the issues of inconsistent data dissemination and the requirement for exchange data
Wen-Hui Lin, Hsiao-Chung Lin, Ping Wang, Bao-Hua Wu, Jeng-Ying Tsai	2018	LeNet-5 model	KDD-CUP99	By using advanced behavioral highlights from prepared CNNs, the proposed technique improves the accuracy of intervention detection to detect threats
Wei-Fa Zheng	2020	Models were trained using (SGD) stochastic gradient descent and Adam optimizers	KDD-CUP99	To overcome the technical limitations of intervention detection, such as their low accuracy and flexibility
Alanoud Alsaleh and Wojdan Binsaeedan	2021	Salp swarm algorithm, feature selection	NSL-KDD and NSW-NB15	Anomaly NIDS performs much better than the latest strategies suggested in the literature when it comes to memory, detection rate and false alarm level on both databases
Hong Yu Yang and Fengyan Wang	2019	TensorFlow, convolutional neural network (CNN)	ICNN and IBWNIDM	There are issues with over-fitting and generalization during the model training process
Xiaoxuan Zhang, Jing Ran, Jize Mi	2019	Synthetic minority oversampling technique combined with edited nearest neighbors (SMOTE-ENN) algorithm	NSL-KDD	Using traditional machine learning to improve IDS performance

D. *Unbalanced Dataset*

The great differences in the distribution of the classes in the dataset are defined as the unbalanced dataset. It means that the dataset is biased toward a class in the dataset. If the dataset is biased toward one class, an algorithm trained on the same data will be biased toward the same class.

E. *Lack of Realism*

To calculate the performance of an intrusion detector with the help of synthetic traces, it is necessary that the traces reflect with the environment in which the

detector can be deployed. If the traces are not real, then the detection task can be too difficult or too easier, which results in an underestimation or overestimation of the performance of the detector.

F. *Low Detection Rate*

The classifier lacks the ability to classify the instance (events) correctly. This affects the detection rate, and the accuracy of the system is reduced.

G. *Understanding and Investigating Alerts*

Investigation of IDS alerts is huge time and resource-consuming and requires supplementary information from other systems which help in deciding whether the alarm is serious. Professional skills are required to predict the system outputs, and many organizations are in need of devoted security experts which are capable of executing this crucial function.

5 Proposed Approach

The proposed architecture experimental setup is shown in Fig. 1; a system of an intrusion detection model was developed that uses pre-processing, feature analysis and machine learning applications for attack detection and classification.

(i) *Dataset Used*

The most generally used dataset among research study for network intrusion detection systems is briefly discussed below.

Data pre-processing is the process of preparing raw data and adapting it to a machine learning model. It is the first and most important step in creating a machine learning model. Pre-processing is mainly performed to assess data quality.

The KDDCUP 1999 dataset, and better version of the previous DARPA 1998 dataset, is not significantly comparable to the complex and common studies, yet it is useful in distinguishing between pouring attacks and reproductive attacks even today.

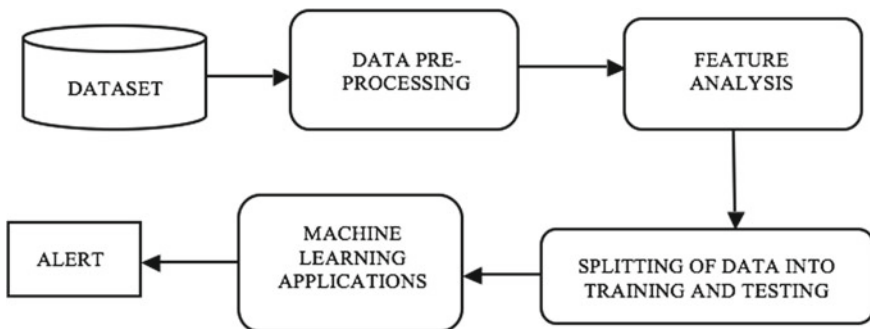


Fig. 1 Proposed architecture

The NSL-KDD dataset is nothing new of its kind, as that is a refined model of the dataset, i.e., KDD 1998 developed by Ali A. Ghorbanifar in the Network Security Laboratory (NSL), and seems to address some of the underlying issues, but problems like talking about McHugh.

The IDS dataset advanced by Tezpur University (TUIDS), India, created features caused during float stage method in the physical test bed and did not include features that could take place all through the flow capturing process. The dataset features consist of attacks only.

The USW-NB15 is the 100 GB synthetic dataset used on this research, coined by Mustafa et al. At the University of New South Wales, Canberra, Australia IXIA Perfect Storm device in the Cyber Range Laboratory. This dataset is generated in PCAP documents with an aggregate of ordinary and intrusive traffic v and other exquisite features.

(ii) *Data Pre-processing*

Data pre-processing is the process of preparing raw data and adapting it to a machine learning model. It is the first and most important step in creating a machine learning model. Pre-processing is mainly performed to assess data quality.

Steps involved in Data pre-processing:

Data Cleaning: Data cleaning is the manner to remove incorrect information or data, incomplete information and inaccurate information from the datasets, and it additionally replaces the missing values.

Data Integration: It is the procedure of combining a couple of sources into a single dataset. The data integration system is one of the important components.

Data Reduction: This process allows in the reduction of the volume of the information which makes the analysis less complicated yet produces the same or almost the equal end result. This reduction additionally helps to reduce storage space. There are a number of the techniques in record discount which are dimensionality discount, numerosity reduction and data compression.

Data Transformation: The alternate made within the layout or the structure of the data is known as data transformation. Based on the requirements, the above step can be simple or complex. The techniques involved in data transformation are smoothing, aggregation, discretization and normalization.

(iii) *Feature Analysis*

The efficiency of an intrusion detection system can be greatly improved by performing the reduction of the traffic feature without any negative effect on the accuracy of the classification. The very important challenge here is to select the best feature selection methods that can exactly decide the features that are

relevant to the intrusion detection task and also the redundancy in between the features.

(iv) *Machine Learning Applications*

- *Decision Tree Algorithm:*
Decision tree algorithm evaluates the information and acknowledges the critical qualities in the system that demonstrates the malicious activities and then increases the value of some security frameworks by checking the positioning of intrusion identification details.
- *Naive Bayes Algorithm:*
Naive Bayes algorithm is an administered learning calculation; it is a successful and most basic classification calculation that aids in making of the quick Artificial Intelligent models which makes fast forecasts. It is a classifier; that is, it can anticipate which depends on the nature of an item.
- *Convolutional Neural Network:*
Convolution neural network [CNN] is a profound learning method, which has many cutting edge executions on order undertakings. Essentially, CNN was found with the execution of picture handling which contrasts in having a convolutional channel contrasted with the completely associated neural organization. The three parts in a CNN are, firstly, the convolutional layer, the pooling layer and the order layer. Here, the convolutional layer and pooling layer are used for the highlight extraction; the characterization layer associated toward the finishing of the organization performing interruption discovery.
- *MLP Classifier:*
A multilayer perceptron (MLP) is the forward fake neural organization. The word MLP is used uncertainly and infrequently to mean any feed forward ANN and now and then thoroughly to identify with the organizations that are made up of various different layers of perceptron. Multilayer perceptron is alluded to as “vanilla” neural organizations, mostly when they have a secret layer. MLP consists of least three layers of hubs, namely an info layer, a secret layer and a result layer. Every hub is a neuron that utilizes a nonlinear initiation work. MLP utilizes a learning method got back to engendering for preparing.

6 Conclusion

This paper shows various issues/challenges of network intrusion detection by way of studying different papers of recent research. A comparison of different IDSs is provided. An approach of IDS model is proposed. The proposed paper gives an intrusion detection model which is focused on a convolutional neural network [CNN] which depends on classifier for increasing the accuracy of model and goals at reducing the false positive rates and increases the accuracy rate.

References

1. Mendonça RV, Teodoro AAM, Rosa RL, Saadi M, Melgarejo DC, Nardelli PHJ, Rodríguez DZ (2021) Intrusion detection system based on fast hierarchical deep convolutional neural network
2. Chen L, Kaung X, Xu A, Suo S, Yang Y (2020) A novel network intrusion detection system based CNN. In: 2020 eighth international conference on advanced cloud and big data (CBD)
3. Vinayakumar R, Soman KP, Poornachandran P (2017) Applying convolutional neural network for network intrusion detection
4. Xiao Y, Xing C, Zhang T, Zhao Z (2019) An intrusion detection model based on feature reduction and convolutional neural networks. *IEEE Access*
5. Ho S, Jufout SA, Dajani K, Mozumdar M (2021) A novel intrusion detection model for detecting known and innovative cyberattacks using convolutional neural network
6. Khan RU, Zhang X, Alazab M, Kumar R (2019) An improved convolutional neural network model for intrusion detection in networks. In: 2019 cyber security and cyber forensics conference (CCC)
7. Hu Z, Wang L, Qi L, Li Y, Yang W (2020) A novel wireless network intrusion detection method based on adaptive synthetic sampling and an improved convolutional neural network
8. Lin W-H, Lin H-C, Wang P, Wu B-H, Tsai J-Y (2018) Using convolutional neural networks to network intrusion detection for cyber threats. In: Meen, Prior, Lam (eds) Proceedings of IEEE international conference on applied system innovation 2018 IEEE ICASI 2018
9. Zheng W-F (2020) Intrusion detection based on convolutional neural network. In: 2020 international conference on computer engineering and application (ICCEA)
10. Alsaleh A, Binsaeedan W (2021) The influence of salp swarm algorithm-based feature selection on network anomaly intrusion detection
11. Yang H, Wang F (2019) Wireless network intrusion detection based on improved convolutional neural network
12. Zhang X, Ran J, Mi J (2019) An intrusion detection system based on convolutional neural network for imbalanced network traffic. In: 2019 IEEE 7th international conference on computer science and network technology (ICCSNT)

Performing Cryptanalysis on the Secure Way of Communication Using Purple Cipher Machine



V. P. Srinidhi, B. Vineetha, K. Shabarinath, and Prasad B. Honnavali

1 Introduction

A cipher machine is an electromechanical device which is used for encrypting and decrypting messages. The primary element of such a machine is the rotors or switches. The wiring between these elements leads to a fixed substitution of characters. The security for these machines is enhanced by the fact that the position of the elements advances every time a character is enciphered or deciphered. This mechanism ensures that the machine is able to produce a complex polyalphabetic substitution cipher. One such machine which uses stepping switches as the primary element is called **THE PURPLE MACHINE**.

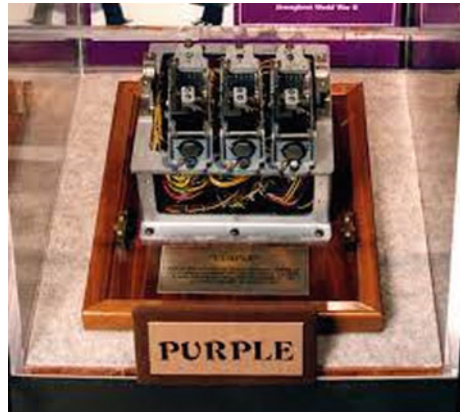
Purple machine was the codename given by the United States to Japan's "System 97 Typewriter for European Characters" (also called "Type B Cipher Machine"). This machine is a World War II era machine which was used to send messages from Japan to officials in Washington, Berlin and London. The machine was compatible with English, Roman and Romaji. It was the most difficult Japanese code to break and was used from 1939 until 1945. The deciphering machine to break the code was created by William and Elizebeth Friedman. The purple machine followed the basic principle where the letters had a fixed substitution and the position of switches changes after every substitution. Both the enciphering and deciphering machine had to have the same switch position configurations. These positions were sent as a coded

V. P. Srinidhi · B. Vineetha · K. Shabarinath (✉) · P. B. Honnavali
Department of CSE, PES University, EC Campus, Bangalore, India
e-mail: shabarinathkookal@gmail.com

B. Vineetha
e-mail: vineethab@pes.edu

P. B. Honnavali
e-mail: prasadhb@pes.edu

Fig. 1 Fragment of an original purple cipher machine [4]



text, which was preceded by a set of coded numbers which revealed the permutations used (Fig. 1).

2 Design of a Purple Machine

The Purple machine is constituted of an input plugboard, stepping switches and an output plugboard. The stepping switches were used to permute the given text. The stepping switch replaced the more exasperating half-rotor. The machine used a super-encipher approach, in which the text was enciphered repeatedly with variations, yielding millions of cipher alphabets to choose from. The machine was designed in such a way that the input and output plugboards were independent. However, the plugging sequence of the machine must be identical or at least related.

The division of the English alphabets was done in two groups. The first group were the **SIXES** which consisted of six characters, namely AEIOUY. The second group were the **TWENTIES** which consisted of the remaining twenty characters of the English alphabet. The purple machine did not specifically encipher sixes as sixes or the twenties as twenties.

The switches used were stepping switches with multiple levels and 25 positions. The switch advances through these positions on the levels providing a new permutation each time.

The input of the **SIXES** is passed to a six-level, 25-position stepping switch. As the switch advances through its 25 positions, the output is **25** scrambled alphabets (out of a possible $6!$ or 720 alphabets). The working when a **TWENTIES** character is considered is a little more complicated. The input of the **TWENTIES** is passed through three twenty-level, 25-position stepping switches. The input for the first switch is the primary permutation of the input character, and the output from this switch is passed to the second switch and the output of second to the third switch.

As a result, the number of possible permutations obtained is equal to $25 \times 25 \times 25$ or **15,625** [2].

The Purple Machine also had a secret key which was important for the functioning of the machine. This code changed daily which made it that much harder to break the code of the machine.

3 Elements of a Purple Machine

A. *Input Plugboard:*

The input plugboard has an internal alphabet part and an external alphabet part. The external alphabet is the keyboard where the user enters the message that needs to be enciphered or deciphered. Each character from the external alphabet is mapped onto a fixed internal alphabet. This permutation is done manually to make sure we get a valid permutation. The result from the internal plugboard is used for the encipherment or decipherment process. Any of the letters can be mapped to either the sixes or the twenties in the plugboard.

B. *Switches:*

The output from the input plugboard (internal alphabet) is then encrypted. The character based on whether it is a SIXES or a TWENTIES is passed to the six-switch or the twenty-switch, respectively. As seen earlier, the sixes switch permutes the sixes character and gives out 25 possible permutations out of the 720 permutations in the total space. On the other hand, the TWENTIES have a larger permutation space due to the three-switch pipeline where each switch produces 25 permutations. As a result, the total number of permutations is equal to 25^3 . The TWENTIES switches can have modes (fast, medium and slow switches), producing six possible switch motions. The switches advance by one position every time a character is enciphered. There are certain rules followed:

- When a sixes character is encrypted, the switch moves up by one position.
- When a twenties character is encrypted, usually the fast switch advances by one position except when the sixes switch is at position 24 or position 25.
 - When the sixes switch is at position 24, and middle switch is at position 25, the slow switch advances.
 - When the sixes switch is at position 25, the middle switch advances.

While advancing if the switch is at position 25, it returns back to position 1.

The switch advancing mechanism makes sure that every time a new alphabet is generated when the next letter is encrypted.

C. *Output Plugboard:*

The output from the switches is passed on to the internal alphabet of the output plugboard. These characters are mapped from the internal alphabet to the external alphabet. This mapping is identical to the mapping done in the input plugboard. The external alphabet is the output typewriter (Fig. 2).

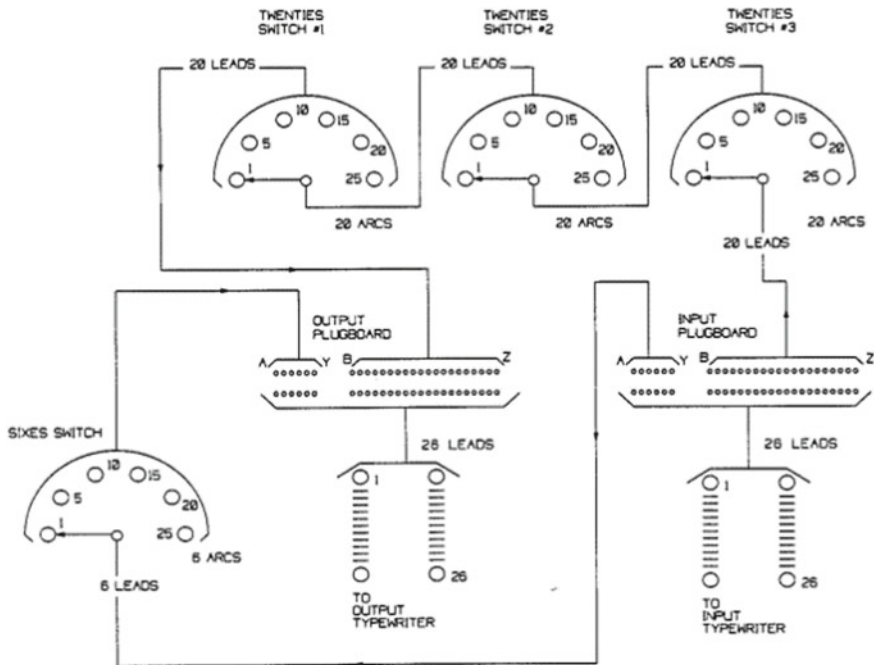


Fig. 2 Simplified wiring diagram of a purple machine [1]

4 Working of a Purple Machine

Algorithm

1. The machine starts with accepting input through the typewriter. Every character of the input is the primary permuted through a fixed substitution. Any input from the external plugboard can be connected to any internal plugboard. For example, If the letter 'O' is connected to 'U,' it will be encrypted as a sixes character. If the letter 'O' is connected to 'S,' then it will be encrypted as one of the twenties characters
2. The primary permuted character is then passed through a switch based on whether it is SIXES character or a TWENTIES character. The connections of the switches are defined in a switch table. Refer to appendix A and appendix B of [3] for a detailed switch table.
3. The output of the switches makes its way to the output plugboard where fixed substitution takes place, and the output is sent to the typewriter.

This encryption method made sure that there was no logical way of predicting how the character might change from the first typewriter to the second typewriter. This proved frequency analysis to be useless for decoding the code. Purple relied on five-digit groups. It allowed 45,000 entries in both an alphabetic encoding dictionary

Table 1 Steps and positions of switches during encryption of “HELLO WORLD”

Plain text	Input	Six	Twenty			Output
			1	2	3	
H	G	4	14	7	12	M
E	C	5	15	7	12	S
L	F	6	16	7	12	C
L	F	7	17	7	12	J
O	E	8	18	7	12	T
W	V	9	19	7	12	P
O	E	10	20	7	12	K
R	J	11	21	7	12	J
L	F	12	22	7	12	M
D	M	13	23	7	12	V

and a numerical encoding dictionary. The numerical encoding dictionary scrambled the alphabetic code, thereby making it harder to decipher [2].

There was also false addition code added during encryption which added an arbitrary five-digit number from a separate codebook for each pre-coded five-digit group. Similarly, false subtraction was used during decryption. This also provided a corruption checking mechanism as this additive was made divisible by three. If the received code was not divisible by three, then it was discarded. In all, there were 18,000 additives in the enciphering book. That meant that there were 45,000 potential code groups that could each be encoded in 18,000 different ways. That is **810 million different possibilities** [3].

The additives were needed to break the basic-code groups and vice versa making it almost impossible to break the machine code (Table 1).

For example, let us review the encryption process of “HELLO WORLD.”

The position of the switches is as follows:

1. Sixes: 3
2. Twenty Switch 1: 21 (fast)
3. Twenty Switch 2: 7 (medium)
4. Twenty Switch 3: 12 (slow).

The motion given for the switches is 123.

The primary permutation used is:

AEIOUY–NOKTYU

BCDFGHJKLMNPQRSTUVWXYZ XEQLHBRMPDICJASVWGZF

The encrypted cipher for “Hello World” with the given configuration is: “**MSCJT PKJMV.**”

5 Pseudo-Code

def StepSwitch (pos):

```

    if pos == 25:
        pos = 1
    else:
        pos += 1
    return pos

```

def StepTwentySwitch (fast, slow, medium, pos):

```

    if pos == 24:
        if middle == 25:
            slow = StepSwitch(slow)
        else if pos == 25:
            middle = StepSwitch(middle)
        else:
            fast = StepSwitch(fast).

```

def Encipherment (input, [position of switches]):

for every character in input:

 Categorize either as sixes or twenties.

 Permute the letter based on the internal plugboard.

 Based on category pass to appropriate switch.

if category == sixes:

 pass through a single 25-level switch

else:

 pass through 3 25-level switches (1, 2, 3)

Permute the character (output of switches) based on the output plugboard and display the result.

Decipherment follows similar algorithm as encipherment. The difference is that when a character passes through the twenties switch, the order is 3, 2, 1.

6 Drawbacks

Purple machine was one of the most complex yet well-developed cryptographic methods of its time. It was able to work without being decoded for more than 2 years during the World War II period. Even though the machine was this advanced, it still had it fair share of drawbacks.

The machine was larger and heavier and was therefore not suitable to be used in combat location.

The splitting of the alphabets in sixes and twenties reduced the total number of permutations from $26!$ to $6! \times 20!$. This reduced the time of cryptanalysts in deciphering the code.

The machine inherited a property of its predecessor RED, which had been broken earlier, that the six letters were encrypted separately. Due to this, the U.S. Army was able to break this before the twenties.

The cryptanalysts used the hill climb attack to decipher the message in an easier manner. It used the concept that once the sixes are separated, the number of possibilities for the twenties reduces and it will be easier to decode them.

7 Hill Climb Attack

Hill climb attack is an approach which uses graph search algorithm where the current path is extended with a successor node which is closer to the solution. For a Purple cipher message, the attack works in the following way: It investigates and determines which trial key is more likely to produce the best result based on English letter frequency statistics, as well as how to improve an existing trial key (hill-climbing). For a more detailed understanding, please refer [1].

8 Breaking of the Purple Code

In 1936, when the RED machine was broken by the U.S. Signal Intelligence Service, the Japanese began creating a new machine to encipher the messages. The machine was named “97-shiki O-bun In-ji-ki” which was later given the codename PURPLE by U.S. This machine was an improvement of the Enigma machine.

This machine was used only to send the most secretive messages because of which there was only a little cipher to work with. One advantage for the cryptanalysts was that, since the machine was relatively new, the messages were sent on both RED as well as PURPLE, which helped them compare.

In August 1939, the U.S. Army hired cryptanalyst William Friedman to help with breaking the PURPLE code. Eighteen months into his work, Friedman suffered a mental breakdown and was institutionalized. After this, his team were able to use him work thus far and started making progress. Eight functional replicas of the machine were created. This was a huge achievement as the U.S. Army had never seen an actual PURPLE machine [4].

After much effort from the team, the working of the machine was completely discovered. However, since the daily keys kept changing and the U.S. Army had still not discovered how these keys work, they could not break the messages.

By this time, Lt. Francis A. Raven was able to crack how the daily keys worked. He found out that the pattern used was that each month was broken into three ten-day segments in which a pattern was discerned. He later made a few changes and ultimately broke the code. With this, the PURPLE cipher was totally broken and the messages could be deciphered [5].

U.S. cryptanalysts broke the 14-part message from Japan which indicated to break-off negotiations with the United States on 7 December 1941. This message gave information on the pearl harbor attack, but it was not delivered on time due to typing difficulties and ignorance [4].

Given below is an exert from the 14-part message:

“YHFLO WDAKW HKKNX EBVPY HHGHE KXIOH QHUIHW IKYJY
HPPFE ALNNA KIBOO ZNFRL QCFLJ T TSSD DOIOC VTAZC KQTSH XTIJC
NWXOK UFNQR TTAOI HWTATW VHOTG CGAKV ANKZA NMUIN YOYJF
SR”

To decode this, we will use the following configuration:

- (i) Plugs NOKTYU-XEQLHBRMPDICJASVWGZF
- (ii) Sixes switch at 13
- (iii) Twenties switch at 1, 24, 10
- (iv) Switch motion 4, i.e., 231.

The message was decoded as:

“THEGO VEENM ENTOF JAPAN LFLPR OMPTE DBYAG ENUIN EDESI
RETOC OMETO ANAMI CABLE UNDERSTAND INLWI THTHE GOVER
NMENT OFTHE UNITE DSTAT ESINO RDERT HATTH ETWOC OUNTR IES”

9 Implementation

We used Python to develop the working of a purple machine. The code contains both encryption and decryption modules.

1. The code starts with accepting input from the user—enciphering or deciphering mode, text to be processed, positions of the switches and motion of the twenties switch.
2. The text is then primary permuted with the fixed substitution. The pattern used for permuting is:
SIXES: AEIOUY to NOKTYU
TWENTIES: BCDFGHJKLMNPQRSTUVWXYZ to XEQLHBRMPDIC-JASVWGZF.
3. As every character is permuted, the permutation character is added to an array. A for loop is used to iterate through the array and perform encryption or decryption.
4. Either sixes or twenties permutation is called based on the character, and encryption takes place incrementing the switch position every time based on the provided conditions.
5. Once the letter is permuted, the letter is changed based on the internal alphabets and appended to a string which is sent out as the output to the user.

10 Experimental Results and Analysis

Encryption:

- The text we would like to encrypt here is “Hello World.”
- We have specified the switch positions as (3, 6, 2, 15) where 3 is the position of the sixes switch and the other three are of the twenties switch.
- The switch motion is 231, i.e., the second switch is the fast, third switch as medium and first as slow switch.
- The encrypted text is “CZMVT PKWJC (Fig. 3).”

Decryption:

- We use the output of encryption as the text for decryption, i.e., “CZMVT PKWJC.”
- We use the same configuration for the position of switches and the switch motion.
- The decrypted text shows “HELLO WORLD (Fig. 4).”

We performed time analysis for the code passing strings of various sizes. The result is shown in Table 2.

From the graph, we can infer that the time taken for encryption and decryption is almost the same for strings of shorter length whereas as the length increases, decryption takes a little longer than encryption (Fig. 5).

```
I:\Engineering\Crypto>python PurpleMachine.py
Enter 1 to Encrypt
Enter 2 to Decrypt
Enter Your Choice: 1

Enter text to be Enciphered/Deciphered: Hello World

Enter Positions of sixes switch , twenties switch 1,2, and 3 seperated by commas3,6,2,

Enter the motion: 231
CZMVT PKWJC
```

Fig. 3 Encryption of “HELLO WORLD”

```
I:\Engineering\Crypto>python PurpleMachine.py
Enter 1 to Encrypt
Enter 2 to Decrypt
Enter Your Choice: 2

Enter text to be Enciphered/Deciphered: CZMVT PKWJC

Enter Positions of sixes switch , twenties switch 1,2, and 3 seperated by commas3,6,

Enter the motion: 231
HELLO WORLD
```

Fig. 4 Decryption of “CZMVT PKWJC”

Table 2 Time analysis for strings of different lengths

Number of characters	Size in bytes	Time taken (encryption)	Time taken (decryption)
500	549	0.00099	0.00089
1000	1049	0.00199	0.00187
5000	5049	0.01196	0.01562
10,000	10,049	0.02590	0.02526
50,000	50,049	0.17154	0.17185
100,000	100,049	0.58195	0.55365
250,000	250,049	3.12316	3.03277
500,000	500,049	12.6320	15.1566



Fig. 5 Graph between the number of characters in the string and time taken to encrypt and decrypt

11 Conclusion and Future Work

The purple machine was a complex yet unique machine. It was considered one of the best cryptographic devices during the World War II era. Though it kept the alphabet partition of that of its predecessor RED, it was able to function and deliver secret military messages for over 2 years. The machine was an improvement on the famous Enigma machine. The Enigma machine provided output as blinking lights, whereas purple used an output typewriter. The driving factor of the purple machine was the stepping switches which advance one position each time a character is enciphered.

The U.S. Army considered PURPLE machine as one of the hardest machines to break. U.S. cryptanalysts were able to decrypt the cipher text as fast as the Japanese counterparts due to the limit of the machine and hill climb attack. They were able to decrypt the 14-part message which broke the ties between Japan and United States.

Our implementation of the PURPLE machine enables enciphering and deciphering of any text with any desired configuration of the position of sixes and twenties

switch. It also gives a way to specify the motion of the switches, i.e., which of the twenties switch will act as the fast, medium and slow switch.

Future work would include designing a front end for the machine. This could be done using any front-end libraries including PHP and React.

Another aspect of future work would include finding a way to incorporate hill climb attack while deciphering the code as we noticed that deciphering takes longer than encryption.

Acknowledgements To begin, we would want to express our gratitude to PES University for giving us with the opportunity to learn, investigate and implement this project. None of this would have been feasible without their resources.

We are appreciative to IFSCR and Prasad B Honnavali, Professor, Dept. of CSE, PES University, for providing us with this fantastic opportunity to learn so much. Being able to learn more about cryptography and cryptographic equipment was a fantastic experience.

We would like to express our heartfelt gratitude to Vineetha B, Assistant Professor, Department of Computer Science and Engineering, PES University, Bengaluru. We would not have been able to complete this internship satisfactorily without her invaluable advice.

References

1. Freeman W, Sullivan G, Weierud F (2003) Purple revealed: simulation and computer-aided cryptanalysis of Angooki Taipu B. *Cryptologia* 27(1):1–43. <https://doi.org/10.1080/0161-110391891739>
2. Garner W (n.d.) Inside purple. <http://Math.ucsd.edu/~crypto/Projects/WillGarner/machine.htm>
3. Dao T, Stamp M (2005) Purple cipher: simulation and improved hill-climb attack. <http://www.cs.sjsu.edu/faculty/stamp/papers/180H.pdf>
4. Alberto-Perez A-P (2015) How the U.S. cracked Japan’s “purple encryption machine” at the dawn of World War II. *Gizmodo*. <https://gizmodo.com/how-the-u-s-cracked-japans-purple-encryption-machine-458385664>
5. Wikipedia Contributors (2021) Type B cipher machine. *Wikipedia*. https://en.wikipedia.org/wiki/Type_B_Cipher_Machine

Artificial Intelligence and Machine Learning for Foundry Industry—A Case Study of Belagavi Foundry Industry



Praveen M. Kulkarni, Prayag Gokhale , L. V. Appasaba, K. Lakshminarayana, and Basavaraj S. Tigadi

1 Introduction

Artificial intelligence has supported for the growth and enhancement to the production and customer experience. This technology has the ability to interact and personalize the process in the manufacturing [5].

This technology is supported by the real-time data analysis from different channel of production and hence increases the response to the production planning and control. Artificial intelligence combined with machine learning acts as a powerful collaboration which supports in leveraging the data and information related to production planning and control in the foundry units [11].

Foundry industry is a primitive industry which consists of pouring a material into a mould after melting it, where it hardens into the preferred profile. The products of foundry industry are applied in various sector such as automotive, water management, agriculture, aeronautic, defence, etc. Hence, the products produced by the foundry units need to make sure that the final quality is as per the highest quality [9].

Foundry units can apply artificial intelligence, which support in accelerating manufacturing and support in digital transformation by reducing costs and enhancing

P. M. Kulkarni (✉)

Department of MBA, K.L.S. Institute of Management Education and Research, Belagavi, Karnataka 590006, India
e-mail: pmkulkarni90@gmail.com

P. Gokhale

Department of MBA, KLE Technological University Dr. M. S. Sheshgiri College of Engineering and Technology, Udyambag Belgaum 590008, India

L. V. Appasaba · K. Lakshminarayana

Department of Management Studies, Visvesvaraya Technological University, Belagavi, India

B. S. Tigadi

Visvesvaraya Technological University, Belagavi, India

efficiency in the foundry units. However, there are challenges with regards to implementation of this technology in the foundry units, namely (a) shortage of talent, (b) technology infrastructure, (c) data quality, (d) real-time decision-making, and (e) trust and transparency [3].

Given the challenges of implementation of this technology, there is also a great potential for implementing this technology in foundry, this ascended by the study conducted by Mckinsey and Co, mentions that production cost can be reduced up to 40% through application of this technology, further the study also indicates that in longer duration this technology would reduce the cost of depreciation by 17% [12].

Therefore, based on the backdrop of these challenges and opportunities of implementation of this technology in the foundry industry, this study is undertaken to understand the influence of these challenges on the implementation of artificial intelligence and machine learning in the foundry units.

These challenges are understood by applying the multi-criteria decision-making method, in particular, TOPSIS which stands for “Technique for Order of Preference by Similarity to Ideal Solution” and Fuzzy AHP means Fuzzy Analytical Hierarchy Process (AHP) method to evaluate select and rank challenges implementing AI and ML in foundry industry [4].

The use of Fuzzy set theory aids in eliminating any uncertain and supports in identification of challenges and gives directions for improving the decision-making process. Likewise, TOPSIS Grey method is applied to rank established on the challenges identified by the Fuzzy AHP method. Therefore, this study provides a direction in understanding the challenges in implementing the AI and ML in foundry units.

The next section of the study includes literature review, methodology, and results followed by discussions and conclusions.

2 Literature Review

The literature review for the study includes the following aspects: (a) artificial intelligence (AI), (b) machine learning (ML), and (c) methods of applying ML in production, (d) challenges of implementing machine learning (ML) and artificial intelligence (AI) in foundry industry.

2.1 Overview of Machine Learning and Artificial Intelligence

ML is a branch of AL which offers the computers the capacity to learn without being definitely programmed. ML and AI correlate the learning from the data like the process and quality information about the manufacturing and services in the organization [2].

In the AI and ML process, this technology makes distinction of learning based on three aspects, they are (a) supervised, (b) unsupervised, and (c) reinforced learning.

In the supervised learning, machine learning is focused and developed to identify the patterns based on the previous data such as product quality data, production output data, and service quality data [10]. In the unsupervised learning, data is analysed without any previously known results, instead, the technology develops an algorithm that learns from the environment which sensor data of machine such as detect and compares the anomalies, and then, results are analysed. In the reinforced learning, there are agent to take action in an environment so as to capitalize on some notion of cumulative return [1].

2.2 Application of AI and ML in Production

As the domain of AI and ML is increasing in the manufacturing, this technology acts as a powerful tool for the possible applications in production. There are different domains in the production where this technology is applied, namely quality management, machining, inventory and logistics management, fault diagnostics, energy management, and job shop scheduling [7].

The application of this technology with regards to quality management is applied for monitoring and even predicting the quality of the product based on process data and reducing the rejection rates in the manufacturing. Apart from quality management, robotics and AI and ML have supported in the object recognition and motion planning which result in higher productivity in the manufacturing [6].

In the domain of supply chain management, this technology supports in predicting the value stream in the logistics of components for production and reduces the supply chain cycle and improves customer satisfaction. Other exemplary fields of application include the job shop scheduling, fault diagnostics, and energy management. The above discussion indicates that artificial intelligence and machine learning play an important role in manufacturing process of the organization.

2.3 Challenges of Implementation of AI and ML in Foundry Industry

Foundry industry operates into different eco-system, where in the products are more into a customized mode of production. However, there is mass production, but patterns and moulds are developed as per the requirement of the customer. The challenges of implementing the artificial intelligence and machine learning in foundry industry are from the five domains, namely (a) talent shortage, (b) technology infrastructure, (c) data collection and management, (d) real-time information, and (e) edge deployment [8].

Talent shortage: Skilled data scientists and AI professionals are unusual. AI tasks require an interdisciplinary group of data scientists, software architects, ML engineers, and BI analysts and SMEs. This need is particularly evident in manufacturing, a sector that many young data scientists consider to be monotonous, repetitive, and unstimulating [8].

Technology infrastructure: Manufacturing locations often have a variety of machineries, tools, and manufacturing methods that use different and sometimes competing technologies, some of which may be running on outdated software that is not compatible with the rest of their system. In the absence of standards and common frameworks, plant engineers must determine the best way to connect their machines and systems, and which sensors or convertors to install [8].

Data quality: Access to clean, meaningful, and high-quality data is critical for the success of AI initiatives, but can be a challenge in manufacturing. Manufacturing data often is biased, outdated, and full of errors, which can be caused by multiple factors. One example is sensor data collected on the production floor in extreme, harsh operating conditions, where extreme temperature, noise, and vibration variables can produce inaccurate data. Plants have historically been built using many proprietary systems, which do not talk to one another, where operational data also may be spread across several databases in numerous formats not appropriate for analytics, requiring extensive preprocessing [8].

Real-time decision-making: This is becoming increasingly important in manufacturing applications, such as monitoring quality, meeting customer delivery dates, and more. Often, decisions need to be acted upon immediately—within seconds—to identify a problem before it results in unplanned outages, defects, or safety issues. Rapid decision-making requires streaming analytics and real-time prediction services that enable manufacturers to act immediately and prevent undesirable consequences [8].

Edge deployments: There are many potential use cases of edge computing in manufacturing, to allow manufacturers to process data locally, filter data, and reduce the amount of data sent to a central server, either on site or in a cloud. Furthermore, a key goal in contemporary manufacturing is to be able to use data from several processes, machines and systems to get used to the manufacturing process in real time. This careful monitoring and control of manufacturing assets and processes use large amounts of data and need machine learning to decide the best action as an outcome of the insight from the data, and likewise entails edge-based computing. The capability to install predictive models is critical to enable smart manufacturing applications [8].

3 Proposed Challenges for Implementing AI and ML in Foundry Industry

In this study, a wide-ranging literature review was steered to identify and evaluate several critical for understanding the challenges for implementation of AI and ML in foundry industry. The study picks five challenges and they formed the criteria for the present study. These criteria include talent shortage (A), technology infrastructure (B), data quality (C), real-time data (D) and edge deployments (E). Table 1 gives the list of criteria and sub-criteria related to the study of challenges of implementation of AI and ML in foundry industry.

4 Research Methodology

The planned approach, i.e., Fuzzy AHP and TOPSIS Grey, is useful to apprehend the challenges of application of AI and ML in foundry units in India. The respondents designated for the study include 35 foundry units operating in the Belgaum Foundry Cluster, Belagavi, and Karnataka, India. This cluster of foundry was selected as this cluster has the more foundry units operating and also contributes to export of foundry products across the globe. The study elaborate twelve experts to allocate weights to various criteria’s and sub-criteria’s and score each supplier for each sub-criterion. The profile of the foundry units and respondents profile is indicated in Table 2. The suggested criteria for understanding the challenges of implementation of AI and ML in foundry industry are analysed using the Fuzzy. TOPSIS has been used to assess and highlight the significant factor to act as an opportunity for the overcoming these challenges in implementation of AI and ML in foundry units.

5 Fuzzy AHP Method

AHP method was proposed by Saaty proposed for multi-criteria decision-making. This method has grown towards more erudite options. Fuzzy-based AHP smears to build a pairwise matrix of decision-makers’ preference by means of TFNs. The Fuzzy scale applied in this research is given in Table 3.

The Fuzzy AHP is adopted in the subsequent stages:

Stage 1: Construct of pairwise matrix

Stage 2: Define the Fuzzy

$$Y = \sum_{j=1}^m T_{gi}^j \times \left[\sum_{i=1}^n \sum_{j=1}^m T_{gi}^j \right]^{-1} \tag{1}$$

Table 1 Challenges of implementation of AI and ML in foundry industry (criteria and sub-criteria)

Criteria	Sub-criteria
Talent shortage (A)	Lack of experienced manpower in AI and ML (A1)
	Lack of interdisciplinary manpower (A2)
	Fresh graduates find AI and ML lacks career growth (A3)
	Present employees find it difficult to learn this technology (A4)
	Foundry unit employees need more time to learn this talent (A5)
Technology infrastructure (B)	Foundry units require investment in this technology (B1)
	Present technology of foundry units is not acceptable to AI and ML (B2)
	Technology to connect sensors to foundry process is difficult (B3)
	The system lacks standard and common framework of technology (B4)
	Technology of AI and ML for foundry is still in development stage (B5)
Data quality (C)	High quality data is important for AI and ML implementation in foundry (C1)
	Employee need training on data feeding in manufacturing (C2)
	Sensors need to be monitored for data collection and data quality output (C3)
	Operating conditions are harsh due heat and dust in manufacturing, and this might influence data quality (C4)
	Data collection from multiple sources are be effective for data quality in manufacturing (C5)
Real-time data (D)	Real-time data in foundry is difficult to complexity of production process (D1)
	Real-time data collection points need more advanced sensors in foundry units (D2)
	Foundry units may not able to provide real-time data due to complex manufacturing infrastructure of foundry (D3)
	To collect real-time data of foundry requires more safety for sensors due to heat and dust in manufacturing (D4)
	Real-time data collection requires more time due to complexity of foundry manufacturing process (D5)
Edge deployments (E)	Deployment of predictive models in foundry units requires more time (E1)
	Collecting precision and monitoring information is difficult in foundry units (E2)
	Edge computing requires to very level infrastructure either locally or through cloud (E3)
	Edge computing required training manpower in AI and ML for foundry (E4)

(continued)

Table 1 (continued)

Criteria	Sub-criteria
	Edge computing requires effective networking system for real-time outcomes (E5)

Table 2 Profile of the respondents

<i>Year of establishment</i>	<i>N</i>	<i>%</i>	<i>Respondent experience</i>	<i>N</i>	<i>%</i>
5–10 years	6	17	5–10 years	25	21
10–15 years	21	60	10–15 years	39	32
15 years and above	8	23	15 years and above	57	47
Total	35	100	Total	121	100
<i>Customer category</i>	<i>N</i>	<i>%</i>	<i>Designation of respondents</i>	<i>N</i>	<i>%</i>
Automobile	18	51	Production managers	80	66
Heavy machinery	10	29	Supply chain managers	32	26
Aerospace	3	9	Quality managers	9	7
Others	4	11		121	100
Total	35	100			

Table 3 Triangular fuzzy numbers (TFNs) scale

Linguistic preference	TFN's
Equally	(1, 1, 1)
Moderately	(2/3, 1, 3/2)
Strongly	(3/2, 2, 5/2)
Very strongly	(5/2, 3, 7/2)
Extremely	(7/2, 4, 9/2)

$$\left[\sum_{i=1}^m \sum_{j=1}^m T_{gi}^j \right] = \left(\frac{1}{\sum_{n=1}^{i=1} \sum_{m=1}^{j=1} b3ij}, \frac{1}{\sum_{n=1}^{i=1} \sum_{m=1}^{j=1} b2ij}, \frac{1}{\sum_{m=1}^{i=1} \sum_{m=1}^{j=1} b1ij} \right)$$

$$(SY_j \geq SY_i) = (d) = \left\{ \begin{array}{l} 1, \text{ incase of } b2_j \geq b2_i \\ 0, \text{ incase of } b1_i \geq b3_j \\ \frac{b1_i - b3_j}{(b2_j - b3_j) - (b2_i - b1_i)}, \text{ otherwise} \end{array} \right\} \tag{2}$$

Stage 4: Calculate minimum possibility degree using equation

$$V(SY \geq SY_1, SY_2, SY_3, SY_4, SY_5 \dots SY_k),$$

for $(i = 1, 2, 3, 4, 5, 6, 7, \dots, k)$

$$V[(SY \geq SY_1), (SY \geq SY_2), \text{ and } \dots (SY \geq SY_k)] = \min V(SY \geq SY_i) \quad (3)$$

for $(i = 1, 2, 3, 4, 5, 6, 7, \dots, k)$

Stage 5: Let's assume weight vector

$$d'(A_i) = \min V(SY \geq SY_i); \text{ for } (i = 1, 2, 3, 4, 5, 6, 7, \dots, k)$$

Then weight vector can be defined as

$$W' = (d'(A_1), d'(A_2), d'(A_3), d'(A_4), d'(A_5), \dots d'(A_n))^T \quad (4)$$

Finally, the weight vector can be normalized using Equation

$$W = (d(A_1), d(A_2), d(A_3), d(A_4), d(A_5), \dots d(A_n))^T \quad (5)$$

where W represents a non-Fuzzy number,

6 Topsis

TOPSIS is generally used for deciphering complex decision problems. The TOPSIS method is adopted using the subsequent seven stages:

Stage 1: Build H matrix

$$[\text{labelsep} = 2.8 \text{ mm}]H = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \quad (6)$$

Stage 2: H matrix normalization

$$g_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^m x_{ij}^2}}, (j = 1, 2, \dots, m), (i = 1, 2, \dots, n) \quad (7)$$

Stage 3: Weighted matrix development

$$q_{ij} = w_j g_{ij}, (j = 1, 2, \dots, m), (i = 1, 2, \dots, n) \quad (8)$$

Stage 4: Use Eqs. 9 and 10 to get positive and negative solution

$$A^+ = \left\{ \max_i q_{ij} \mid j \in J, \left(\min_i q_{ij} \mid j \in J' \mid i \in n \right) \right\} = [q_1^+, q_2^+, \dots, q_m^+]z \quad (9)$$

$$A^- = \left\{ \min_i q_{ij} \mid j \in J \right\}, \left(\max_i q_{ij} \mid j \in J' \mid i \in n \right) \left. \right\} = [q_1^-, q_2^-, \dots, q_m^-] \tag{10}$$

Stage 5:

$$d_i^+ = \left[\sum_{i=1}^m (q_{ij} - q_j^+)^2 \right]^{1/2}, (i = 1, 2, \dots, n) \tag{11}$$

$$d_i^- = \left[\sum_{j=1}^m (q_{ij} - q_j^-)^2 \right]^{1/2}, (i = 1, 2, \dots, n) \tag{12}$$

Stage 6:

$$C_i^+ = \frac{d_i^-}{d_i^+ + d_i^-}, (i = 1, 2, \dots, n) \tag{13}$$

Stage 7: Rank the alternatives on the basis of C_i in stage 6.

Grey System Theory

Prof. Deng proposed the Grey system theory on the basis Grey set concept.

The theory uses a grey no. to minimize uncertainty in the data.

$$\otimes a + \otimes b = [\underline{a} + \underline{b}; \bar{a} + \bar{b}] \tag{14}$$

$$\otimes a - \otimes b = [\underline{a} - \underline{b}; \bar{a} - \bar{b}] \tag{15}$$

$$\otimes a \times \otimes b = [\min(\underline{ab}, \overline{ab}, \bar{a}\underline{b}, \underline{a}\bar{b}); \max(\underline{ab}, \overline{ab}, \bar{a}\underline{b}, \underline{a}\bar{b})] \tag{16}$$

$$\otimes a : \otimes b = \otimes a \times \left[\frac{1}{\underline{b}}, \frac{1}{\bar{b}} \right]; 0 \notin \otimes b \tag{17}$$

TFNs can be converted into grey numbers using $a^\sim = (a1, a2, a3)$, and $b^\sim = (b1, b2, b3)$ into grey numbers $\otimes a = [a1, a2]$, and $\otimes b = [b1, b2]$ using Euclidean distance between $\otimes a$ and $\otimes b$ as given in the equation below:

$$d(\otimes a, \otimes b) = \sqrt{\frac{1}{2} \left[(\underline{a} - \underline{b})^2 + (\bar{a} - \bar{b})^2 \right]} \tag{18}$$

7 Results and Analysis

The results are indicated in two stages. In the first stage, Fuzzy AHP results are presented, where in results with regards to weights for the main criteria and sub-criteria are presented. In the second stage, TOPSIS Grey results are indicated with ranking indicating the alternatives for the challenges for implementation of AI and ML in foundry units.

Fuzzy AHP Results

The analysis with Fuzzy AHP has four levels, firstly development of Hierarchical structures, secondly, main criteria weights, thirdly, sub-criteria weights, and fourth, final weights sub-criteria.

The first level is hierarchical structures is developed based on the four parts, namely goals, criteria, sub-criteria, and alternatives. The details hierarchical structure is presented in Fig. 1.

The results from main criteria weights indicated in Table 4 gives that the challenge with regards to technology infrastructure (B), 0.221 ranked a highest challenge in foundry units for implementation of AI and ML in foundry. Further, the second ranked weight is 0.216 Data Quality (C) which indicates second challenge with regards to implementation of this technology in foundry units in the study units of foundry. The third weight age is 0.21, real-time data (D) collection challenge for analysing the information for AI and ML technology in foundry units. Talent shortage (A) is ranked fourth with 0.205 weights in the ranking of the weight age and fifth ranking of weight is edge deployment (E) 0.149 as weight age.

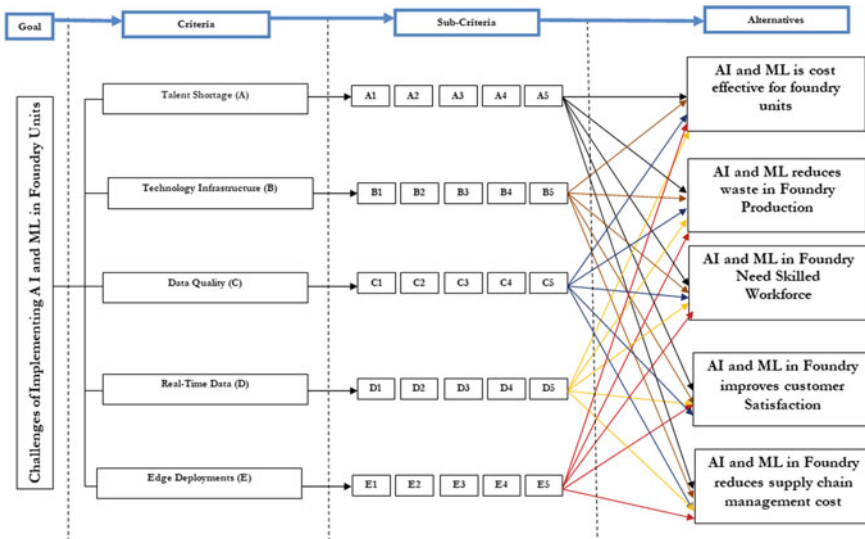


Fig. 1 Hierarchical structure

Table 4 Results with regards to main criteria, sub-criteria, and ranking of the criteria on challenges of implementation of AI and ML in foundry

Criteria	Main criteria weight	Sub-criteria code	Sub-criteria weight	Global weight	Rank
Talent shortage (A)	0.205	A1	0.225	0.0461	12
		A2	0.2	0.0410	16
		A3	0.17	0.0349	20
		A4	0.24	0.0492	11
		A5	0.165	0.0338	21
Technology infrastructure (B)	0.221	B1	0.276	0.0610	7
		B2	0.173	0.0382	17
		B3	0.165	0.0365	18
		B4	0.232	0.0513	10
		B5	0.154	0.0340	20
Data quality (C)	0.216	C1	0.201	0.0434	13
		C2	0.199	0.0430	14
		C3	0.167	0.0361	19
		C4	0.239	0.0516	9
		C5	0.194	0.0419	15
Real-time data (D)	0.210	D1	0.301	0.0632	4
		D2	0.291	0.0611	6
		D3	0.541	0.1136	2
		D4	0.612	0.1285	1
		D5	0.356	0.0748	3
Edge deployments (E)	0.149	E1	0.231	0.0344	21
		E2	0.33	0.0492	11
		E3	0.415	0.0618	5
		E4	0.122	0.0182	22
		E5	0.356	0.0530	8

After the application of Fuzzy AHP in the sub-criteria weights show that in the rank of in the range of 1–5 shows in the Global Weight is ranked higher, with regards to complexity of foundry technology, and it working environment to implement sensors is ranked first with 0.1285, while manufacturing complexity due to technology infrastructure is ranked second with 0.1136. Third is ranked with regards to complexity due to batch production method applied in the foundry, this influences the AI and ML implementation in foundry. Fourth is ranked after difficulty in production, planning, and control method in foundry units, and fifth is ranked after difficulty in connections of sensors and computers for collection of data for application of AI and ML in foundry. The detailed results from the other weight age are information is presented in Table 4.

Table 5 Ranking of opportunity for implementation of AI and ML in foundry units through TOPSIS grey

Alternatives	D+	D–	C+	Ranking
AI and ML reduce cost in overall production of foundry	9.66908	6.77085	0.4118	3
AI and ML reduce wastage	9.59228	6.24699	0.3943	4
AI and ML need skill development	10.0157	5.76561	0.3653	5
AI and ML improve customer satisfaction	8.35958	7.78878	0.4823	1
AI and ML reduce supply chain cost	9.03350	6.98951	0.4362	2

7.1 Ranking of Alternatives Using TOPSIS Grey

The developed TOPSIS Grey-integrated methodology has been used to assess and prioritize the alternatives of ranking for opportunities for implementation of AI and ML in foundry units. The results show that AI and ML provide an opportunity for improving customer satisfaction of foundry industry (0.482326). This technology also provides opportunity to reduce the supply chain cost of foundry industry (0.436217), and this is ranked second in results analysis. This technology reduces overall cost of production (0.411854). The detailed information with TOPSIS Grey result analysis is provided in Table 5.

8 Discussion

The result analysis indicates that technology infrastructure is ranked among the key factor of challenge for implementation of AI and ML in the foundry units. The technology infrastructure development is influenced by the factors associated with data collection of analysis through AI and ML such as influence of heat, dust, and batch method of production process of foundry.

The study findings have given a new directions in the study of challenges with regards to AI and ML implementation in foundry, as previous studies have indicated that talent shortage is the key factor and challenge for implementation of this technology in foundry units. Further, studies have also indicated that edge technology as a factor of challenge of implementation. This technology is related to networking and advanced computing for implementation of this technology in foundry. However, there is an opportunity of implementation of this technology in foundry units, the results analysis indicated that, this technology supports the foundry units with improved customer satisfaction and reduced cost in the supply chain management.

The above discussion provides a direction for practical implications for improving implementation of AI and ML in foundry units. Firstly, foundry units need to invest in technology especially related to sensor and cloud computing for data capturing and analysis, this improves the efficiency in real-time data analysis. Secondly, foundry

units need to invest in this technology as this technology supports in cost reduction in supply chain management and other manufacturing cost as this technology collect real-time data and based on this data faster decision-making can be taken by the managers of foundry. Thirdly, foundry units need to train employees to using this technology in the foundry units.

9 Conclusion and Future Research

The overall results indicated that this technology is effective in foundry industry as this technology supports in customer satisfaction and reducing cost of production. However, there are challenges with regards to technology development for collecting real-time data due to foundry manufacturing eco-system. Further studies can be undertaken other foundry cluster of other states of India and also studies can be carried out and compared with developed and underdeveloped countries foundries. Finally, the study results indicate that AI and ML is a powerful tool for foundry industry for improving production efficiency and enhancing customer satisfaction.

References

1. El-Tantawy S, Abdulhai B, Abdelgawad H (2013) Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto. *IEEE Trans Intell Transp Syst* 14(3):1140–1150
2. Goldenberg SL, Nir G, Salcudean SE (2019) A new era: artificial intelligence and machine learning in prostate cancer. *Nat Rev Urol* 16(7):391–403
3. Gramegna N, Greggio F, Bonollo F (2020) Smart factory competitiveness based on real time monitoring and quality predictive model applied to multi-stages production lines. In: IFIP international conference on advances in production management systems. Springer, Cham, pp 185–196
4. Hanine M, Boutkhoum O, Tikniouine A, Agouti T (2016) Application of an integrated multi-criteria decision making AHP-TOPSIS methodology for ETL software selection. *Springerplus* 5(1):1–17
5. Krishnamoorthy CS, Rajeev S (2018) Artificial intelligence and expert systems for artificial intelligence engineers. CRC Press
6. Mayr A, Kißkalt D, Meiners M, Lutz B, Schäfer F, Seidel R, Franke J (2019) Machine learning in production-potentials, challenges and exemplary applications. *Proc CIRP* 86:49–54
7. Mayr A, Weigelt M, Masuch M, Meiners M, Hüttel F, Franke J (2018) Application scenarios of artificial intelligence in electric drives production. *Proc Manuf* 24:40–47
8. Peres RS, Jia X, Lee J, Sun K, Colombo AW, Barata J (2020) Industrial artificial intelligence in industry 4.0-systematic review, challenges and outlook. *IEEE Access* 8:220121–220139
9. Ravi B (2010) Casting simulation–best practices. In: Transactions of 58th IFC. Ahmedabad, pp 19–29
10. Renz A, Hilbig R (2020) Prerequisites for artificial intelligence in further education: identification of drivers, barriers, and business models of educational technology companies. *Int J Educ Technol High Educ* 17(1):1–21

11. Sahu CK, Young C, Rai R (2021) Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: a review. *Int J Prod Res* 59(16):4903–4959
12. Thomas DS, Gilbert SW (2014) Costs and cost effectiveness of additive manufacturing. *NIST Spec Publ* 1176:12

Conversion of Sign Language to Text Using Machine Learning



Aishwarya Bhagwat , Poonam Gupta, and Nivedita Kadam

1 Introduction

This template sign language is a form of correspondence used by people who are deaf. People who are differently abled use multiple language movements as a non-verbal communication technique to communicate their feelings and ideas to other people. However, since these ordinary people have a hard time understanding their expressions, experienced sign language experts are required for medical and legal appointments, as well as educational and training sessions. There has been an upsurge in demand for all these services in recent years. Other types of services, such as video distant location human interpreter using an increased connection to the internet, have been presented, providing an easy-to-use sign language interpreter service that can be used and stands to benefit, but with significant limitations. To address this, we created a custom CNN model to recognize sign language gestures. Three convolution: multiple max-pooling layers, dense layer and smoothing layer; these factors are used to create a convolutional neural network. To train the model to recognize the gesture, we utilize the MNIST Indian Sign Language dataset. The collection includes the characteristics of Indian alphabets and digits and used OpenCV for custom CNN (fully convolutional) model to recognize a sign from a real-time webcam. The precise extraction of hand movements actions, as well as face emotions, is vital. Studies are mostly focused on particular sign computational linguistics, such

A. Bhagwat (✉) · P. Gupta · N. Kadam
Computer Engineering, G.H Rasoni College of Engineering and Management, Pune, India
e-mail: aishwarya.24.ab@gmail.com

P. Gupta
e-mail: poonam.gupta@raisoni.net

N. Kadam
e-mail: nivedita.kadam@raisoni.net

as video as well as sensor-based multiple languages recognition and sign translation software. Furthermore, the performance of sign vernacular interpretation and portrayal methods is significantly influenced by accurate sign language recognition. Sign language contains alphabets, letters or words. Communication is a way to exchange feelings. The deaf community have to be dependent on interpreter to interact with the normal people; due to this, they suffer very much. The objective of our system is to develop an application that can convert the Indian Sign Language to text form.

People who are differently abled are suffering because they are isolated from social interactions. People who have disability have to be dependent on interpreters or use costly means of communications like using IoT-based data gloves to recognize signs. Associative technology can assist them and change their life entirely to lead a great life. Vocal impairment and deafness are a disability from which many people are suffering. Due to this reason, person cannot communicate with the world which ultimately leads to isolation from society. So researchers are trying to find a solution for the deaf-mute people. The solution to this is sign language recognition, an application that will recognize the sign language and transform the output to textual form. Earlier the work was done on sign language using IoT devices like data gloves which is very difficult and costly to use in daily life. In glove method, IoT devices like sensors and motion tracker are used. In vision-based system, images are captured using web camera. Convolutional neural network automatically detects the features. The proposed system is very effective and efficient in compared to earlier works as very less work has been done on Indian Sign Language. A complete review of the application (which are capture, identification, interpretation and reproduction) is offered as well as their significance in their area is analyzed.

Why we have used convolutional neural network (CNN):

The main advantage of using CNN over other algorithms is that it is more accurate as compared to other algorithms. It can be used for both unsupervised and supervised algorithms. It classifies image datasets easily. Very less human intervention is required for preprocessing the dataset as the CNN algorithm automatically detects all the important and required features in an image. This will reduce the number of steps required for preprocessing the dataset. The CNN algorithm uses local spatial coherence that uses adjacent pixels for local connectivity. CNN uses pooling layers that can help to reduce the size of image and downscale an image. Thus, CNN will help to increase the accuracy of the system as compared to other neural networks.

2 Related Works

Communicating is one of the most fundamental prerequisites for societal life. Individuals with hearing problem communicate with one another using gestures, which are difficult to understand for normal persons. In today's world, almost every language has its own established gesture-based communications. It is critical to provide an understanding of signs for various populations that are unfamiliar with the sign

language [1]. It is a system of hand signals that includes visual gestures and signs. Conversational signals, regulating gestures, manipulating motions and instructive motions are some of the other types of hand motions [2]. The use of a few parts of our body, such as our fingers, hand and arm to express data, is known as sign language. A system that recognizes hand movements from a digitally enhanced dataset is known as recognition of image [3]. This technique is now used in a wide range of applications, including robotics and telerobotics, games, virtual reality and human–computer interaction (HCI). The majority of the presented models rely on traditional example recognition, which necessitates human competence for extraction and recognition. Deep learning’s recent success, particularly the residual neural network (ResNet) for machine vision, is being utilized to detect Indian Sign Language (ISL) as an object recognition challenge [4].

This paper [5] suggests employing Markov Chain models to categorize the outcome, trajectory parameters and alignment structure of the multiple languages (HMM). This technique’s intrinsic features make it suitable for use in gesture recognition [6]. Here, author provided a technique in which a total of 262 signs was gathered from two distinct endorsers, with a precision of 95% using a HMMs classifier. When the database is taught and evaluated using the signals of various people, the accuracy is considerably reduced. Artificial neural network-based approaches for real-time American Sign Language (ASL) identification were suggested [7]. The Microsoft Kinect sensor is utilized in the study to detect signs for two applications: arithmetic calculation and the rock-paper-scissors game, with an accuracy of more than 90% for ASL. Setting subordinated HMMs and a technique for linking three-dimensional strategies on ASL were used [8]. The framework grouped 53 ASL with an accuracy of 89.91%. Convolutional neural network (CNN) was utilized [9] to recognize sign for Indian Sign Languages. To identify Arabic Sign Language, author employed a neural network [10]. Kadam and Ghodke [11] developed an Indian Sign Identification System for twenty-five English letters and nine number signs. This study used PCA for sign classification and segmentation for fingertip method for feature extraction. The accuracy of this method was 94%. Others suggested a technique for detecting Tamil sign letters in their paper [12]. This approach employed images with a resolution of 640×480 pixels. The photographs are then transformed to grayscale images. This technique achieved a precision of 96.87% for the static method and a rate of 98.75% for the dynamic method [13].

3 System Analysis

3.1 Existing System

The current systems use instrumented gloves, block-based picture information, improved Kalman filter and skin color segmentation. All these existing methods use multiple sensors, utilize complex algorithm to recognize the sign language.

3.2 Proposed System

We describe the system conversion of sign language to text using machine learning to support better communication between the hearing and speech impaired people and common people. This system is very efficient as well as economical as we do not have to buy instrumented gloves or buy any external device. Our system will eliminate the communication barrier by converting the sign language to text. There is no need for the hearing and speech impaired people to rely or be dependent on human for converting the sign language, and they can be more independent.

4 System Implementation

We have proposed a system that helps to translate the Indian Sign Language into text form.

The proposed system follows below-mentioned process:-

1. Dataset
2. Image preprocessing and hand segmentation
3. Feature extraction from data
4. Classification algorithm

4.1 Dataset

The suggested technique is divided into many parts. We collected data from a standard dataset in the first stage. We turned each picture to grayscale and resized it to 6464 pixels after gathering the data. After that, we used normalizing methods to transform the gray level data from 0–255 to a range of 0–1 values. We skip two convolution processes in the convolution layer and put the input well before final ReLU activation. In this stage, we applied the algorithm to recognize Indian Sign Language by extracting the attributes of picture data. To increase efficiency, we used an optimization approach (Adam optimizer) in the fourth phase. Finally, we employed real-time augmentation to improve the variety of data accessible for the training set in the final stage. We have used MNIST dataset (Kaggle) for Indian Sign Language recognition. The pictures of Indian Sign Languages come from a common dataset. In order to provide fixed length input to our suggested model, we turned each picture to grayscale and resized it to 6464 pixels. In the field of image processing, pattern matching is a rapidly developing discipline. ResNet is a crucial component of computer vision. Convolutional layers are the most significant layer for extracting features from images in ResNet. To comprehend ResNet, think of it as a collection of residual blocks, each of which has a convolutional layer, batch normalization and the ReLU activation function. ResNet also includes a skip layer

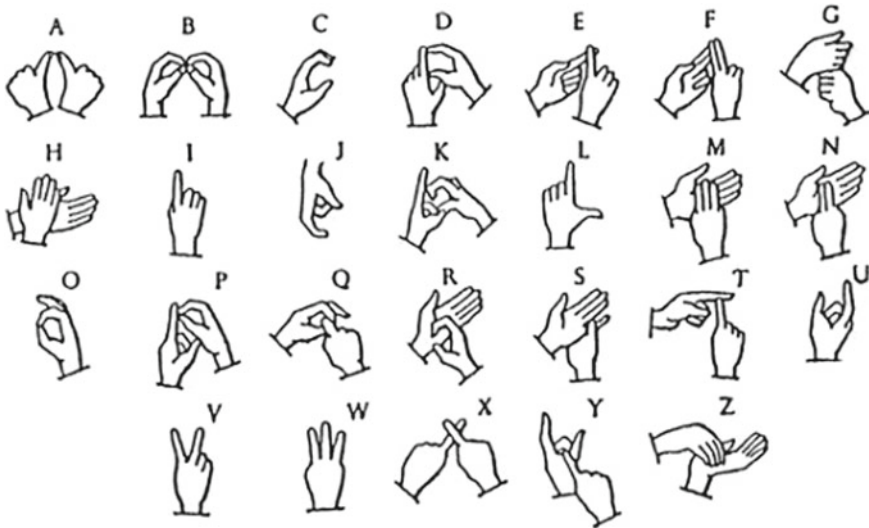


Fig. 1 Indian Sign Language sample dataset alphabets. Image Source [14]

Fig. 2 Indian Sign Language sample dataset numbers. Image Source [15]

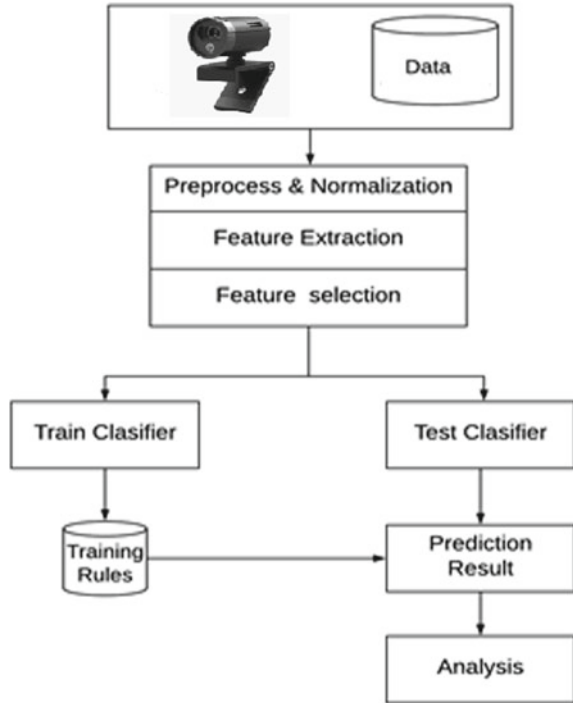


that aids in the resolution of the vanishing gradient issue. To create this model, we utilized Keras API with TensorFlow as a backend. The output is then combined with the result of the first layer, a process known as multiple hops, and lastly the transfer functions ReLU is applied to the output. This is maintained until the last layer, then we utilize flatten layer and linked thick layer with 24 classes, followed by Softmax function for posterior distribution (Figs. 1 and 2).

4.2 Image Preprocessing and Hand Segmentation

The resolution of all camera device is not same. The output will be images with different resolution. All images are rescaled to uniform size in order to decrease computational effort and to compare the features accurately. To obtain accurate prediction, we preprocess the image before feeding it to the algorithm. We have converted image to grayscale. In the region of hand movement, the color will be converted to grayscale. This will result in binarized image. Image is converted to small segments for more accurate prediction. Color is utilized as descriptor for object detection. We use color space YCbCr for hand detection. We have used YCbCr over RGB as we need to channelize all the colors in one direction. Then, next step is

Fig. 3 System flowchart for project



applying thresholding to analyze the image. We convert grayscale image to binary image. In thresholding, we compare each pixel value with the threshold value. An image consists of noise and target objects. In order to extract target part from image, thresholding is used (Fig. 3).

4.3 Extraction of Features

In data that we have collected, we need to extract features from it. Dimensionality reduction is also known as feature extraction. There are a lot of variables in a dataset which increase the computing cost. Dimensionality reduction helps to extract the best features that describe the dataset. We use feature extraction for easy processing of dataset and provide the extracted features to algorithm.

Distance transformations are used to blur the location of features. Fourier descriptors are used to draw a boundary shape around image. In content-based image, shape is very important feature. Fourier descriptors can be applied irrespective of size, scale of image. Therefore, we have used Fourier transformation as it is independent of shape and size of object in image. To match the query object with object in

database, we have used feature vector. Some of the features are color, area, length and gradient direction. These features are than compared with feature vector.

4.4 Classification Algorithm

We have used convolutional neural network algorithm as it is very beneficial when we want an internal representation of an image. Convolutional neural network allows the algorithm to show the position and scale in various structures of image dataset. Convolutional neural network is used in classification problems, regression problems and image dataset. Convolutional neural network is used more in data containing spatial relationship. In feedforward neural network, there are large number of neurons, this will increase the computational cost, so we have used convolutional neural network as it will extract the important features and transform the image from high dimension to low dimension range.

Convolutional Layer:

It is the foundation for constructing a CNN model. This layer conducted mathematical calculations on the picture that was used as input, as well as resizing the image into the $M * M$ format. This layer's output describes the image's features, such as edge and corner mapping, also known as a feature map.

Dense Layer:

This layer works for classification, and it uses the training data as background knowledge and identifies the current activity accordingly. In practice, input is given through a graphical user interface (GUI). Now, for the GUI, we have built a new file in which we have constructed an interactive window in which we can draw object on the canvas and identify them using buttons (Fig. 4).

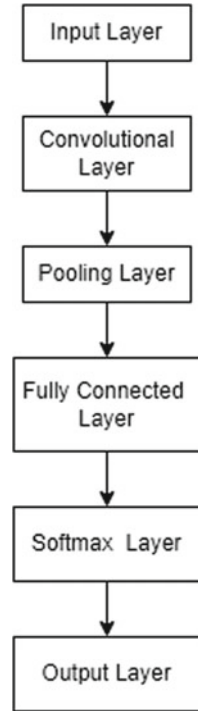
Layer 1: Input layer:

Input layer of convolutional neural network uses image. Image is three-dimensional matrix.

Layer 2: Convolutional Layer (ReLU + Convo):

All the features are extracted in the convolutional layer. Convolutional layers are important part in convolutional neural networks. A convolutional layer converts the inputs image with a kernel or filter. A kernel is very small representation of the image which is to be converted. It is called convolution mask and convolution matrix. The length with which kernel glides is called stride length. The kernel glides along the width and height of image, and the scalar product of image and kernel are computed at spatial distribution. The output image is called as convolved feature.

Fig. 4 Layers of convolutional neural network



Layer 3: Pooling Layer:

The output feature maps are precise to location of input image. Therefore, we have to downsample the feature maps. So, we have used pooling layers as it will reduce the feature map dimensions. This will increase the computational accuracy by reducing the huge parameters. Thus, pooling layer will make the model more fast and accurate. Pooling layers accumulate the feature in patches of feature map. In each patch, we calculate average value of feature map; this is called average pooling. The output of pooling layer is accumulated version of features in input image. Thus, we have reduced the parameters, reduced overfitting, extracted important features from input layer and reduced computational cost by using pooling layer (Fig. 5).

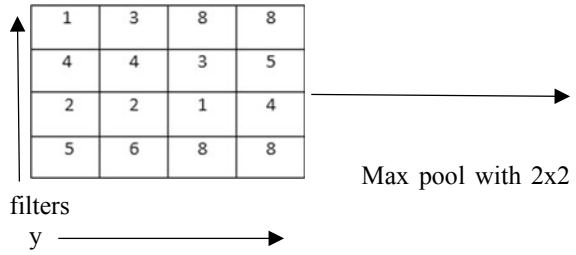
Layer 4: Fully Connected Layer:

In neural networks, input from one layer is used in activation unit of next layer. The final output is generated which uses the connected layers of previous layers in fully connected layers. This layer takes a lot of time to compile the output.

In Fig. 6:

- (1) There are 4 feature units and 5 activation units in the next hidden layer.
- (2) Bias units in every layer are 1's.
- (3) Input values are b01, b02, b03 and b04. These are basic features.

Fig. 5 Pooling layer



Output: After pooling

4	8
3	6

Pooling Layer

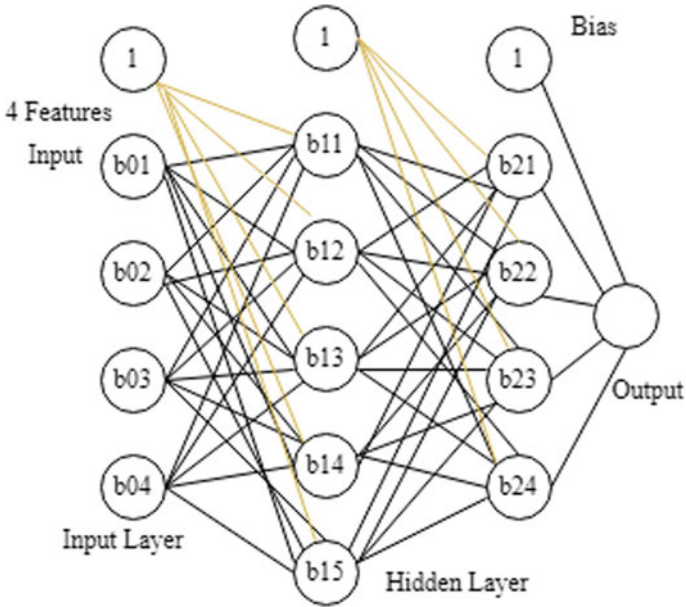


Fig. 6 Fully connected neural network

- (4) The 4 feature units are connected to 5 activation units of hidden layers. The weights of each feature connect the two layers.

Layer 5: Logistic/Softmax Layer:

As the input layers contains positive, zero and negative values, it must be converted to values between 0 and 1 so that they can be used as probabilities. If input is small, it is converted into smaller probability, and if input is large, it is converted into larger probability; the value will be always between 0 and 1. Softmax function is also called multi-class logistic regression or Softargmax function. Softmax is like sigmoid function. This is used to represent probability.

Layer 6: Output Layer:

The output layer contains one-hot encoded label. The output layer can be designed uniquely to increase the accuracy of end results. It is the last layer.

4.5 Convolutional Neural Network Architecture Used in Project

We have used here three convolutional layers. The first level accepts the low-level features $50 * 50$ size grayscale image. The activation map contains size $49 * 49$ and 16 filters, so total size is $16 * 49 * 49$. An activation layer is applied to remove negative values, and it is placed with zero value. Then, max-pooling is applied which results in $25 * 25$ size which takes only maximum value of regions in map. Second convolutional layer recognizes curves and angles. Here, it has 32 filters and $23 * 23$ size activation map, so total is $32 * 23 * 23$. Then, max-pooling is applied, and result is $32 * 8 * 8$ size considering the maximum values of regions. High level of gestures is identified in third layer of convolutional layer; here 64 filters are used, and total size is $64 * 4 * 4$. The max-pool is used to reduce the map $64 * 1 * 1$. The output is 1D array with 64 length. The dense layer then expands the array to 128. The next layer removes random elements of map. In the last step, the dense layer decreases the array to 44 elements which will be number of classes (Fig. 7).

5 Training and Testing

Keras Image Data Generator is used for testing and training the dataset. The validation of dataset is used to measure the loss and accuracy of each epoch and to prevent the model from overshooting loss minima.

Fig. 7 Convolutional neural network architecture for project

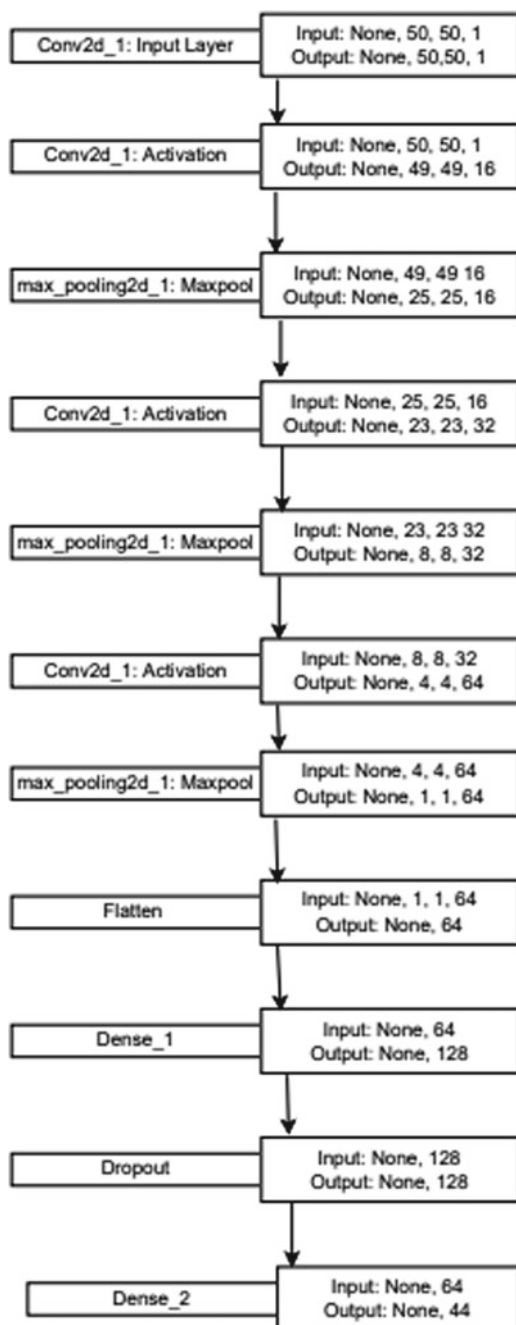


Table 1 Comparison of various algorithms and their accuracy

Authors information	Methodology	Algorithms	Accuracy (%)
Shape texture-based hand gesture	Shape and localized features extraction technique has used for detection of sign language	SVM, KNN, DTW	86.20
KNN classification techniques	KNN-based image feature selection for classification	SIFT and HOG	95.70
Real and static hand gesture	OpenCV methodology has been used for detection of sign language	Euclidian distance approach for template matching	62.10
Wavelet descriptor and FMCC technique	Collaboration of wavelet descriptor as well as Mel sec frequency cepstral coefficients (MFCC) approach has been used for extraction of features	KNN and SVM	93.20
Proposed	CNN features were extracted	Deep CNN	95.90

6 Accuracy

The accuracy is evaluated using recall and F -score. F -score is defined as harmonic mean of two values recall and precision.

7 Results and Discussions

Intel i7 CPU 2.7 GHz is used with 16 GB random access memory for execution. The ResNet (32, 50, 101 and 152) version is used for experimental investigation of proposed systems including 5G network. The major factors considered are execution time (including data processing, data uploading and downloading, etc.), memory consumption, network overhead and energy for evaluating the efficiency of proposed systems (Table 1, and Fig. 8).

8 Conclusion

In order to enhance mobile artificial intelligence, we expect that this framework can provide knowledge on the use and implementation of machine learning. These techniques will stimulate additional research and implementation of scenarios that will allow the network and services to be increasingly automated in the future. A machine learning framework that allows flexible allocation of resources in machine

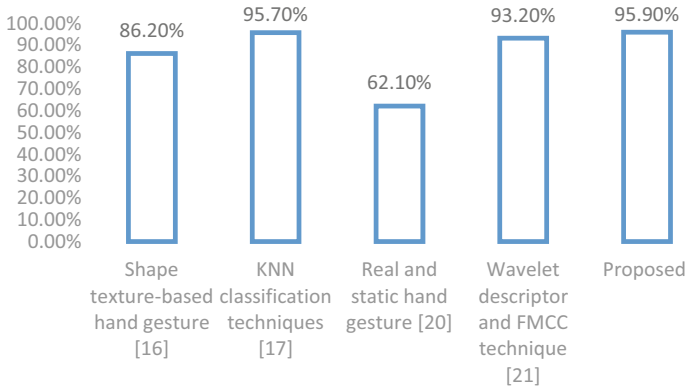


Fig. 8 Analysis of proposed system using ResNet-100 with CNN various machine learning algorithms

learning systems is defined in this paper. Time consumption as well as memory usage, the system trades off system performance. To this end, through code block execution, we proposed a comprehensive mechanism that does not lead to any loss of accuracy and is friendly to the resource restriction environment. The experiments investigate the performance, good configurability and original machine learning process of the system by installing the application with Caffe and TensorFlow frameworks.

9 Future Scope

Our system can be extended with the additional features and techniques that can create a system of sign language detection system that can recognize the facial expression and video streaming. A fully automated Indian Sign Language recognition with a sign to text and text to speech converter which can analyze the video in real time and generate an output based on voice and text can be developed. For future work to detect moving objects in runtime direction using hybrid machine learning will be interesting task for enhancement of current research.

Acknowledgements I take the opportunity to thank my mentor Mrs. Poonam Gupta and Mrs. Nivedita Kadam for their most valuable guidance.



References

1. Wu Y, Huang TS (1999) Human hand modeling, analysis and animation in the context of HCI. In: Proceedings 1999 international conference on image processing (Cat. 99CH36348), vol 3, pp 6–10. <https://doi.org/10.1109/ICIP.1999.817058>

2. Muhammed MAE, Ahmed AA, Khalid TA (2017) Benchmark analysis of popular ImageNet classification deep CNN architectures. In: 2017 International conference on smart technologies for smart nation (SmartTechCon), pp 902–907. <https://doi.org/10.1109/SmartTechCon.2017.8358502>
3. Shawon A, Jamil-Ur Rahman M, Mahmud F, Arefin Zaman MM (2018) Bangla handwritten digit recognition using deep CNN for large and unbiased dataset. In: 2018 international conference on Bangla speech and language processing (ICBSLP), pp 1–6. <https://doi.org/10.1109/ICBSLP.2018.8554900>
4. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 770–778. <https://doi.org/10.1109/CVPR.2016.90>
5. Das, Gawde S, Suratwala K, Kalbande D (2018) Sign language recognition using deep learning on custom processed static gesture images. In: 2018 international conference on smart city and emerging technology (ICSCET), pp 1–6. <https://doi.org/10.1109/ICSCET.2018.8537248>
6. Mustafa M (2021) A study on Arabic sign language recognition for differently abled using advanced machine learning classifiers. *J Ambient Intell Human Comput* 12:4101–4115. <https://doi.org/10.1007/s12652-020-01790-w>
7. Gurbuz SZ et al (2021) American sign language recognition using RF sensing. *IEEE Sens J* 21(3):3763–3775. <https://doi.org/10.1109/JSEN.2020.3022376>
8. Roy PP, Kumar P, Kim BG (2021) An efficient sign language recognition (SLR) system using Camshift tracker and hidden Markov model (HMM). *SN Comput Sci* 2:79. <https://doi.org/10.1007/s42979-021-00485-z>
9. Sruthi CJ, Lijiya A (2019) Signet: a deep learning based Indian sign language recognition system. In: 2019 international conference on communication and signal processing (ICCSP), pp 0596–0600. <https://doi.org/10.1109/ICCSP.2019.8698006>
10. Aly S, Aly W (2020) DeepArSLR: a novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition. *IEEE Access* 8:83199–83212. <https://doi.org/10.1109/ACCESS.2020.2990699>
11. Kadam S, Ghodke A, Sadhukhan S (2019) Hand gesture recognition software based on Indian sign language. In: 2019 1st international conference on innovations in information and communication technology (ICIICT), pp 1–6. <https://doi.org/10.1109/ICIICT1.2019.8741512>
12. Sharma S, Singh S (2020) Vision-based sign language recognition system: a comprehensive review. In: 2020 international conference on inventive computation technologies (ICICT), pp 140–144. <https://doi.org/10.1109/ICICT48043.2020.9112409>
13. Jayanthi P, Thyagarajan KK (2013) Tamil alphabets sign language translator. In: 2013 fifth international conference on advanced computing (ICoAC), pp 383–388. <https://doi.org/10.1109/ICoAC.2013.6921981>
14. <https://images.app.goo.gl/kQmp2suEoq2YYoSH8>
15. https://upload.wikimedia.org/wikipedia/commons/thumb/c/c8/Asl_alphabet_gallaudet.svg/1200px-Asl_alphabet_gallaudet.svg.png

A Hexagonal Sierpinski Fractal Antenna for Multiband Wireless Applications



Mahesh Mathpati, Veerendra Dakulagi , K. S. Sheshidhara ,
Mohammed Bakhar, H. K. Bhalдар, A. A. Jadhav, and Dhanashree Yadav

1 Introduction

The concept of fractal was first developed by a scientist Benoit Mandelbrot in 1975. There are basically 2 types of fractal antenna viz. Sierpinski fractal and Koch fractal. As the days are passing the applications of antennas are increasing rapidly. The fractal structure uses self-similar concept in design which maximizes the effective length or increases the perimeter of antenna geometry. The key aspect of fractal lies on the iterations or the repetitions formed. Due to the iterations, fractal antennas can become compact, multiband and wideband and used in many wireless applications [1, 2]. Any patch antenna consists of 3 layers, the patch at top, middle substrate and at the bottom ground [3]. The antenna size depends on the operating frequency. In this paper a hexagonal sierpinski fractal antenna is designed for UWB applications. The Coplanar Waveguide (CPW) fed technique used to response the high frequency [4]. A modified sierpinski fractal-based microstrip antenna for ultrahigh frequency (UHF) radio frequency identification (RFID) can be designed by combining the techniques of corner cutting with fractal shape [5]. Sierpinski Carpet Fractal Antenna is designed at 2.4 GHz frequency by introducing C shaped slot at rectangular patch which supports a multiband characteristics [6].

A sierpinski gasket fractal multiband antenna can be used for Wi-Fi and cognitive radio applications with modified structure. The novel Microstrip triangular fractal

M. Mathpati · H. K. Bhalдар · A. A. Jadhav · D. Yadav
College of Engineering, Pandharpur, Maharashtra, India

V. Dakulagi (✉) · M. Bakhar
Guru Nanak Dev Engineering College, Bidar, Karnataka, India
e-mail: veerendra@ieee.org

K. S. Sheshidhara
Nitte Meenakshi Institute of Technology, Bengaluru, Karnataka, India
e-mail: shashidhar.ks@nmit.ac.in

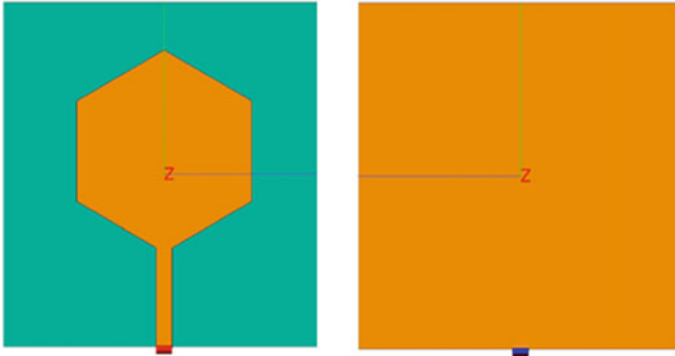


Fig. 1 Original hexagonal patch antenna design with full ground

antenna has multiband behavior which covers the frequency band from LTE, X-band (8–12 GHz), Ku-band (12–18 GHz), K-band (18–26.5 GHz), Ka-band (26.5–40 GHz) [7–11]. The array formation in the sierpinski fractal design leads to increase the bandwidth, resonant frequencies and gain of the antenna [12, 13]. The fractal antenna with multiple frequency band is always better than the different patch antennas for same frequencies [14]. Sierpinski fractal antenna with electromagnetic band gap structure helps to suppress the harmonics, improves the gain and bandwidth as compared with sierpinski fractal antenna without electromagnetic band gap [15]. The combination of planar metamaterial concept with compact ultra wide band sierpinski antenna helps to effectively increase the bandwidth of antenna [16]. The local optimization helps to improve the performance and miniaturization of an antenna [17]. Sierpinski antenna with triangular slots using midpoint geometry of triangle enhances the bandwidth when modified with circular shape [18, 19].

2 Antenna Design

The original antenna design is a microstrip antenna with hexagonal shaped patch. The antenna has been designed on the FR4 substrate with length 30 mm, width 28 mm, thickness 1.6 mm and dielectric constant 4.4. It has ground plane with length 30 mm, width 28 mm. The side of regular hexagon is 8.8 mm with a feed line length 9 mm and width 1.4 mm. The design is shown in Fig. 1.

3 Result and Discussion

The design and simulation of the antenna is done with the help of CAD FEKO software. The simulation results are obtained for reflection coefficient, total gain,

Table 1 The simulation results of the antenna with full ground and partial ground

Full ground	8.56 GHz	-13.56 dB
	10.80 GHz	-11.24 dB
	13.35 GHz	-23.47 dB
	18.27 GHz	-17.16 dB
Partial ground	5.05 GHz	-14.38 dB
	7.19 GHz	-13.61 dB
	9.29 GHz	-24.87 dB
	11.65 GHz	-13.51 dB
	13.63 GHz	-11.40 dB
	16.86 GHz	-20.25 dB

radiation pattern, voltage standing wave ratio (VSWR) and impedance of the designed antenna. The simulation results of the hexagonal patch antenna with full ground and the hexagonal antenna with partial ground are shown in Table 1 and the graphical representation of the reflection coefficient is shown in Fig. 2. From the simulation results, we can easily observe that the frequency bands as well as the bandwidths are improved for the partial ground plane.

The sierpinski fractal structure has been embedded in the design for further improvements in the results of an antenna. The simulation results viz., resonant frequencies, VSWR, impedance, reflection coefficient and total gain of Iteration 0, Iteration 1, Iteration 2 are shown in Table 2 and the graphical representation of frequencies and reflection coefficients for proposed antenna (Iteration 2) is shown in Fig. 3.

Fig. 2 Frequency and reflection coefficient relationship for full ground and partial ground

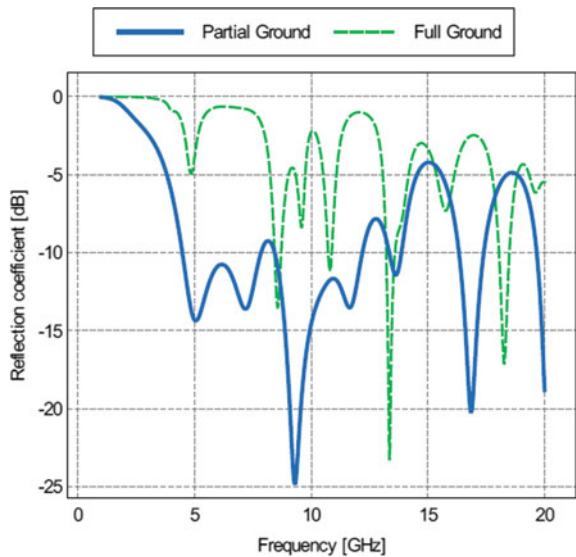
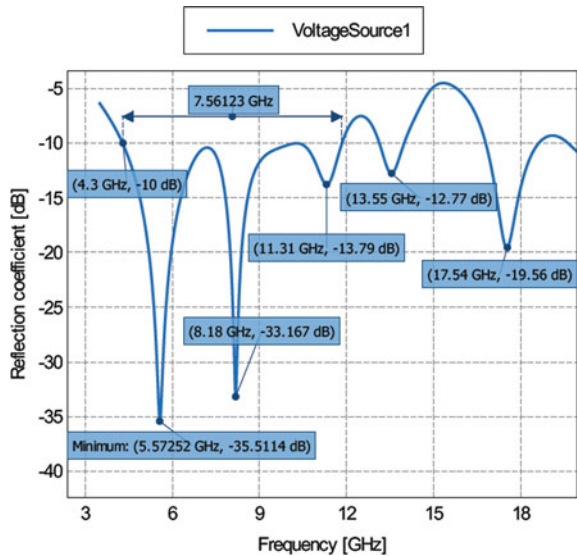


Table 2 Iteration wise simulation results of antenna parameter

	Resonant frequency (GHz)	VSWR	Impedance (Ω)	Reflection coefficient (dB)	Total gain (dBi)
Iteration 0	5.82	1.22	44.5	-20	4
	8.09	1.1	45.3	-26.17	8
	11.36	1.61	55.1	-12.67	8
	13.55	1.68	76.9	-11.98	7.5
	17.40	1.36	44.0	-16.81	6
Iteration 1	5.63	1.22	49.6	-30.24	4
	8.17	1.1	49.3	-44.52	8
	11.47	1.61	57.3	-12.40	8
	13.57	1.68	72.7	-13.63	7.5
	17.35	1.36	40.3	-15.96	5
Iteration 2	5.57	1.04	49.3	-35.51	4
	8.18	1.05	52.2	-33.17	6
	11.31	1.5	54.9	-13.79	8
	13.55	1.6	75.6	-12.77	7.5
	17.54	1.2	45.6	-19.56	6

Fig. 3 Frequency and reflection coefficient relationship for proposed antenna



From the simulation results, it has been observed that all the antenna parameters has acquired best results in Iteration 2 than that of Iteration 0 and Iteration 1. The proposed antenna has VSWR 1.04, 1.05, 1.5, 1.6 and 1.2 values, impedance 49.3 Ω , 52.2 Ω , 54.9 Ω , 75.6 Ω and 45.6 Ω values, reflection coefficients -35.51 dB, -33.17 dB, -13.79 dB, -12.77 dB and -19.56 dB values and total gain 4 dBi, 6 dBi, 8 dBi, 7.5 dBi and 6 dBi values for frequencies 5.57 GHz, 8.18 GHz, 11.31 GHz, 13.55 GHz and 17.54 GHz, respectively. The Bandwidth of an antenna has been increases effectively in Iteration 2. The maximum bandwidth for Iteration 0 is 7.472 GHz, for Iteration 1 is 5.6355 GHz and for Iteration 2 is 7.56123 GHz. The frequency bands for proposed antenna, i.e., for Iteration 2 are 4.3–11.9 GHz, 13.148–13.980 GHz and 16.83–18.60 GHz.

4 Conclusion

This paper gives analysis about the square shaped sierpinski fractal structure in hexagonal patch. The proposed antenna radiates at 3 different frequency bands viz. 4.3–11.86, 13.148–13.98 and 16.83–18.60 GHz with 5 resonant frequencies 5.57, 8.18, 11.31, 13.55 and 17.54 GHz, VSWR 1.04, 1.05, 1.5, 1.6 and 1.2 values, impedance 49.3, 52.2, 54.9, 75.6 and 45.6 Ω values, reflection coefficients -35.51 , -33.17 , -13.79 , -12.77 and -19.56 dB and total gain 4, 6, 8, 7.5 and 6 dBi for respective frequencies. It has been observed that the designed antenna gives best performance for Iteration 2 and it can be used for C-band (4–8 GHz), X-band (8–12 GHz), some part of Ku-band (12–18 GHz), Designed antenna is suitable for vehicle to everything (V2X), Dedicated Short Range Communications (DSRC) and Wireless Access in Vehicular Environments (WAVE) communications bands (5.850–5.925 GHz) applications.

References

1. Mathpati MS, Bakhar M, Dakulagi V (2021) Design and analysis of a fractal antenna using jeans material for WiMax/WSN applications. *Wireless Pers Commun.* <https://doi.org/10.1007/s11277-021-08418-y>
2. Bhaladar H, Gowre S, Mahesh, Dakulagi V (2021) Design of circular shaped microstrip textile antenna for UWB application. *IETE J Res.* <https://doi.org/10.1080/03772063.2021.1982416>
3. Kaur R, Singh H (2017) Review on different shape fractal antenna for different applications. *Int J Adv Res Comput Sci* 8(4):339–342
4. Bairy P, Ashok Kumar S, Shanmuganantham T (2017) Design of CPW fed hexagonal sierpinski fractal antenna for UWB band applications. In: *IEEE international conference on circuits and systems*, pp 107–108
5. Yin-kun W, Du L, Lei S, Jian-Shu L (2013) Modified sierpinski fractal based microstrip antenna for RFID. In: *IEEE international wireless symposium*
6. Sran SS, Sivia JS (2016) Design of C shape modified sierpinski carpet fractal antenna for wireless applications. In: *International conference on electrical, electronics, and optimization techniques*, pp 821–824

7. Prasanthi Jasmine K, Ratna Spandana S (2018) Design and performance analysis of sierpinski diamond fractal antenna for multi-band applications. In: SPACES, pp 85–89
8. Rajshree A, Sivasundarapandian S, Suriyakala CD (2014) A modified sierpinski gasket triangular multiband fractal antenna for cognitive radio. In: ICICES2014
9. Petkov PZ, Bonev BG (2014) Analysis of a modified sierpinski gasket antenna for Wi-Fi applications. In: 24th international conference Radioelektronika
10. Lizzi L, Massa A (2011) Dual-band printed fractal monopole antenna for LTE applications. *IEEE Antennas Wirel Propag Lett* 10:760–763
11. Baliarda CP, Borau CB, Rodero MN, Robert JR (2000) An iterative model for fractal antennas: application to the sierpinski gasket antenna. *IEEE Trans Antennas Propag* 48(5):713–719
12. Mathpati MS, Bakhar Md, Vidyashree D (2017) Design and simulation of sierpinski fractal antenna array. In: International conference on energy, communication, data analytics and soft computing, pp 2514–2518
13. Cao TN, Krzysztofik WJ (2018) Design of multiband sierpinski fractal carpet antenna array for C-band. In: 22nd international microwave and radar conference (MIKON), pp 41–44
14. Maharana MS, Mishra GP, Mangaraj BB (2017) Design and simulation of a sierpinski carpet fractal antenna for 5G commercial applications. In: IEEE WiSPNET 2017 conference, pp 1718–1721
15. MAA X, Mi S, Lee YH (2015) Design of a microstrip antenna using square sierpinski fractal EBG structure. In: IEEE 4th Asia–Pacific conference on antennas and propagation (APCAP), pp 610–611
16. Lamari S, Kubacki R, Czyzewski M (2014) Frequency range widening of the microstrip antenna with the sierpinski fractal patterned metamaterial structure. In: 20th international conference on microwaves, radar and wireless communications
17. Koziel S, Saraereh O, Jayasinghe JW, Uduwawala D (2017) Local optimization of a sierpinski carpet fractal antenna. In: ICIIS, pp 1–5
18. Ramprakash K, Shibi Kirubavathy P (2017) Design of sierpinski fractal antenna for wide-band applications. In: International conference on innovations in information, embedded and communication systems
19. Bal VK, Bhomia Y, Bhardwaj A (2017) Carpet structure of combination of crown square and sierpinski gasket fractal antenna using transmission line feed. In: International conference on electrical and computing technologies and applications (ICECTA)

Multi-level Hierarchical Information-Driven Risk-Sensitive Routing Protocol for Mobile-WSN: MHIR-SRmW



B. V. Shruti  and M. N. Thippeswamy 

1 Introduction

In the last few years, the exponential increase in wireless communication demands has revitalized academia-industry to achieve quality of service (QoS)-oriented communication systems(s) to meet major demands. Amongst the major wireless technologies, wireless sensor network (WSN) has been used in a diverse range of applications serving health care, industrial monitoring and control, surveillance systems, and numerous machine-to-machine (M2M) communication purposes. On the other hand, the rise in technologies like Internet of Things (IoTs) too have broadened the horizon for WSNs to serve different real-time decision purposes. The decentralized and infrastructure-less network characteristics make WSN a most sought technologies to serve real-time surveillance and communications even under disaster or natural calamity [1–4]. Despite of a significantly large application horizon, WSN has always been a challenging network due to resource constrained, greedy, and static network characteristics. On the contrary, in the last few years, IoT and M2M communication systems have sought mobile-WSN to serve QoS communication while guaranteeing energy efficiency [5]. Most of the classical WSN protocols apply reactive route management strategies, which often undergo limited performance due to lack of dynamism and inferior scalability over large network size [5]. The classical standard IEEE 802.15.4 applies reactive routing strategy, and lacks addressing topological variation, and its impact on communication efficiency. The change in network topology often results into network outage, link-loss, packet drop, congestion, etc. Consequently, it results into degraded performance of the overall network [6]. A

B. V. Shruti (✉)

Nitte Meenakshi Institute of Technology, Yelahanka, Bangalore 560064, India

e-mail: Shruthi.bv@nmit.ac.in

M. N. Thippeswamy

Ramaiah Institute of Technology, MSRIT, MSR nagar, Bangalore 560054, India

few researches indicate that the multihop transmission which is often employed over large network size undergo delay and packet loss that impacts over all QoS performance. On the other hand, M2M communication which undergoes communication with dynamic topology too requires a robust routing protocol or strategies to retain QoS performance. To alleviate such problems, introducing mobility in the network is of great significance. The use of mobility (say, mobile-WSN) can help reducing the number of hops in data gathering or distribution [6]. Similarly, it can broaden the scalability of the network to serve more applications demanding mobile communications. In this reference, in the last few years, a few researches have been done towards inculcating mobility with WSN. However, guaranteeing network reliability and QoS performance has remained challenge [1–7].

Exploring in depth one can find that medium access control (MAC) framework does have decisive role towards QoS assurance in WSNs. MAC protocols which often controls network adaptive activities to meet network demands, require intrinsic improvement, and mobility-adaptive scheduling capacity to ensure QoS-oriented communication over mobile-WSN [7]. Unlike PHY-centric routing decisions, MAC-based approaches help making optimal routing decision over diverse operation or network conditions [8]. Unlike standalone layer-based routing (i.e. PHY layer adaptive and system layer adaptive) decisions, the use of cross-layer architecture-based methods has yield superior performance even under mobile topology [7, 8]. In fact, PHY layer helps in controlling the rate of transmission and power management, while system layer information facilitates a large number of dynamic network information such as congestion, flooding probability, packet velocity, network link availability or link-loss probability, etc. Noticeably, these all parameters do vary over time in mobile-WSN communication [7]. Therefore, to alleviate such issues, applying (dynamic) network parameters adaptively towards routing can be of great significance. A MAC routing protocol applying the different cross-layer information can help achieving QoS performance in WSNs [JSMCRP] [10, 11]. Recent literatures also reveal that the strategic amalgamation of the routing protocol and MAC can help ensuring QoS performance [10–12]. In this reference, we developed a first of its kind dynamic network-driven MAC protocol named JSMCRP in [], which exploited both PHY layer information as well as the system layer information from network layer, application layer, and MAC layer to perform mobile-WSN communication. Unlike classical MAC protocols or the routing solutions, JSMCRP protocol was designed as a proactive network exploiting the congestion information from the MAC layer, link quality information from the network later, and data-specific priority information from the application layer to perform dynamic routing decision. Undeniably, unlike classical standalone parameter-based routing, JSMCRP protocol yields superior performance in terms of optimal resource utilization, higher throughput, low packet loss, etc., for both real-time traffic as well as non-real-time data. This approach exploited the single hop information (i.e. node information (cumulative congestion degree, packet velocity, dynamic link quality, and node rank information) pertaining to each neighbouring node towards the destination). Despite the fact that JSMCRP protocol yields better performance, it could not address the uncertainty in the link-reliability or outage caused due to malicious not or hardware failure. In other words,

JSMCRP applied one hop information to perform routing decision, where a neighbouring node with suitable node features was selected as the best forwarding node. However, it failed in addressing the problem which could be caused due to the failure in the subsequent forwarding node. On the other hand, iterative estimation of the node score for each forwarding node as per the JSMCRP criteria might result into exhaustive process and hence can impact energy consumption. To alleviate such problems, it is vital to consider both node transmission efficiency (i.e. energy efficiency and QoS constraints) as well as reliability (i.e. fault resilience) while performing QoS-centric routing decision in mobile-WSN.

Considering above problem and allied scopes, in this research paper, a state-of-the-art new and robust QoS-oriented routing protocol is developed. The proposed model can be called as an extension to our previous work, JSMCRP [9]. In this paper, a multi-level hierarchical information-driven risk-sensitive routing protocol is developed for mobile-WSN (MHIR-SRmW). As the name indicates, our proposed MHIR-SRmW protocol exploits dynamic network information as well as node behavioural aspects to perform node profiling and risk assessment, based on which it executes routing decision. At first, our proposed MHIR-SRmW protocol exploits node information such as IEEE 802.11 MAC information exchange, flooding information, link availability, and topological information such as link-index variation rate (LIVR) and cumulative congestion degree. The proposed risk profiling model helps MHIR-SRmW to segment the best suitable forwarding nodes for future transmission. However, realizing the fact that there can be the node death due to hardware malfunction, physical damage or even sudden malicious node attacks, and therefore, merely estimating best forwarding node can't help ensuring QoS provision. To address this problem, MHIR-SRmW considers a heuristic-driven multi-path availability-based recovery strategy (HMPARS). Our proposed MPARS model intends to design multi-path transmission or relaying strategy with minimum number of shared components in the forwarding paths. Noticeably, the proposed method considers single transmission path at a time. However, retains two additional relaying paths for recovery transmission in case of any node failure or node death. Moreover, when deciding the forwarding paths, our proposed MHIR-SRmW model guarantees that the three different forwarding paths have no shared component or connected component. Unlike classical Dijkstra shortest distance-based recovery strategy, MPARS model applies a heuristic concept that guarantees that the recovery paths don't have the shared component and considers only those nodes with optimal node profile value to perform data transmission. In case of any fault during dynamic routing, MHIR-SRmW protocol switches the relaying path without undergoing beaconing or node discovery, and thus saves energy as well as time. In MHIR-SRmW, at the network start phase, only our proposed model estimates supplementary recovery paths and saves in the proactive table. Once identifying any link-outage, it switches to another disjoint path and completes data transmission. In this manner, MHIR-SRmW protocol ensures higher reliability as well as QoS assurance, which makes it suitable for real-time WSN communication. The overall proposed routing protocol (i.e. MHIR-SRmW) was simulated using Network Simulator 2 (NS2), and performance was examined in terms of throughput, packet loss, delay, and energy consumption.

The other sections of this manuscript are divided as follows. Sect. 2 discusses the related work, which is followed by proposed method and its implementation in Sect. 3. Section 4 presents the results and discussion and the overall research conclusion and allied inferences are given in Sect. 5. References used in this manuscript are given at the end of the manuscript.

2 Related Work

This section discusses some of the key literatures pertaining to the WSN MAC protocols for QoS-centric communication. Authors in [8] proposed three MAC protocols named routing-enhanced MAC (RMAC), pipelined routing-enhanced MAC (PRMAC), and CLMAC to improve delay and reliability. To perform routing decision, authors applied merely the packet delivery rate and delay information to perform routing decision. Realizing congestion as a frequent problem in WSN, authors [10] proposed contention radio-based MAC (CR-MAC). Additionally, applying CR-MAC authors reframed a synchronous routing model named joint routing and MAC (JRAM-MAC) in [11]. However, authors [10–12] failed in addressing the fault probability and its impact on routing performance. A further improved model was developed in [13] which found that JRAM-MAC can be suitable in terms of energy and packet delivery. Even though, it could not address other QoS aspects like reliability under mobile topology and delay. In [14], authors proposed an adaptive operating cycle MAC protocol which focused on maintaining optimal trade-off between energy and QoS in wireless mesh network. Similarly, authors in [15] designed MAC and PHY layer information to perform routing in WSN. Authors applied shortest path information to perform QoS-centric routing in mobile network. In [16], authors designed a non-destructive interference MAC (NDI-MAC) model for constrained network routing. Author proposed a multi-channel pure collective Aloha (MC-PCA) MAC protocol [16]. However, it was highly computationally exhaustive. In [17], authors proposed routing-enhanced MAC (RM-MAC) protocol where authors focused on achieving timely data delivery.

Authors in [17] suggested to design a cross-layer routing concept towards WSN routing, yet it failed in addressing network dynamism and link vulnerability adaptive routing solution. In [18], a directed diffusion routing protocol (DDRP) was developed. Authors designed packet delivery rate analysis-based MAC routing protocol for WSN. In [19], a receiver-centric MAC (RC-MAC) was proposed where authors applied the different traffic conditions to perform routing decision. In [20], a MAC protocol named harvested energy adaptive MAC (HEMAC) was developed by using periodic listen and sleep mechanism with two frames pioneer (PION) and explorer (EXP). Authors applied delay as the decision variable to perform routing decision in WSN. In [21], authors proposed a joint MAC and routing protocol for Wireless Body Network (WBAN). To achieve it, authors designed a cross-layer routing concept by applying MAC and PHY layer information. To further improve MAC performance towards QoS-oriented WSN routing, different algorithms like

Hybrid Medium Access Control (H-MAC), Hybrid Sensor Medium Access Control (HSMAC), and Hybrid Medium Access Control (H-MAC) were proposed in [22]. These algorithms were developed to improve the performance of adaptive demand multi-path distance vector (AOMDV) routing purpose. Authors found that H-MAC-based AOMDV performs better than other methods. A few efforts like [23, 24] applied energy-aware routing protocol EAMP-AIDC by using adaptive individual duty cycle (AIDC) optimization model which applies residual energy as the decision variable to decide duty cycle [24]. A similar effort was made in [25] named adaptive energy efficient and rate adaptation-based MAC routing protocol (AEERA – MACRP). In [26], authors proposed an energy-aware MAC protocol routing MAC protocol (EARMP) that changes duty cycles to cope up with network demands.

Towards fault-resilient routing, authors [26] proposed location-based RMAC (RL-MAC) which exploited inter-node information to configure network with minimum contention window [26]. Unfortunately, authors could not address fault resilience and post-fault proactive routing decision. Authors [27] improved CSMA/TDMA MAC model with channel-adaptive named iQueue-MAC to retain QoS performance under burst traffic. As an enhanced solution, authors [28] proposed power-control and delay aware routing MAC (PCDARM) for multi-path transmission. To inculcate multihop transmission, authors applied hop extended pipelined routing MAC (HE-PRMAC) for WSNs. To ensure timely data transmission, authors [29] developed residual time-driven RD-MAC and depth-base routing MAC (DBR-MAC). In [30], authors suggested a cross-layer model using network layer and MAC layer to perform routing over WSNs. Similarly, authors in [31] proposed cross-layer protocol for multi-sink WSN (MS-WSN). In [32], buffer-reservation-based MAC was proposed for WSN. In [33], authors developed adaptive geographic any cast (AGA-MAC) protocol by solving sleep-delay problem of asynchronous preamble-based MAC. AGA-MAC protocol selects best forwarding node to complete transmission. However, it failed in addressing fault proneness under dynamic topology. Authors [34] exploited dynamic information from the different layers PHY, MAC, and network layers to perform WSN routing protocol. Contention-based cross-layer synchronous MAC protocol (CROP-MAC) was developed for WSN [24]. COP-MAC applied staggered sleep/wake scheduling, synchronization, and routing layer information to perform routing decision. Author [28] proposed a cross-layer model that dynamically switches the MAC behaviour between TDMA and CSMA. In [35], MAC-aware routing protocol for WSNs was proposed using two-hop information to make next hop routing. However, authors could not address the link-outage probability over run time to perform QoS-centric routing. Despite of the different efforts, most of the existing methods have failed to address network dynamism caused fault possibility while making QoS-centric routing. Moreover, no significant effort could address the possibility of iterative faults due to joint or shared node death in forwarding path. It can be considered as the key driving force behind this study.

3 System Model

This section primarily discusses the overall proposed JSMCRC protocol and allied implementation.

In this section, the overall proposed MHIR-SRmW protocol and allied implementation is discussed. To ensure QoS-oriented communication in mobile-WSN, our proposed MHIR-SRmW model encompasses two key steps. These are:

1. Multi-layered hierarchical dynamic information-based node profiling,
2. Heuristic-driven multi-path availability-based recovery strategy (HMPARS).

Being an extension of our previous work, JSMCRP [], in this research, we exploited dynamic node information from both PHY layer as well as system layers (in OSI, the layers above PHY are referred as System layer) to perform node profiling, which act as a decision variable to perform suitable forwarding node candidate estimation. Once identifying the best set of forwarding nodes, MHIR-SRmW executes MPARS which identifies the best multiple forwarding paths with no shared components. Here, we hypothesize that ensuring no shared component(s) in two disjoint paths can help alleviating iterative faults or link-loss over large dynamic network, especially under certain attack or physical damage conditions. In this reference, our proposed MPARS model applied particle swarm optimization algorithm that exploits node profile information along with the node connectivity information to identify a set of recovery paths which helps rerouting or forwarding the data under any node failure or link-outage. In this manner, it intends to guarantee QoS-centric transmission in WSN network. The detailed discussion of the overall proposed model is given in the subsequent sections.

3.1 Multi-layered Hierarchical Dynamic Information-Based Node Profiling

In sync with a dynamic network like mobile-WSN, it is always a case when all nodes undergo exceedingly high network dynamism, change in topology, swift and frequent link change and outage, congestion, etc. This dynamism often causes a network to undergo link-unavailability and hence results into transmission failure that not only impose retransmission but also mounts large redundant transmission cost, energy consumption, and delay. Eventually, such events result into QoS violation. The classical WSN protocols like reactive protocols often fail in addressing such challenges due to high delay and energy consumption. To alleviate it, in this work, we define MHIR-SRmW model as a proactive network management model. As the name indicates, we exploit the different dynamic network information from the different layers of the protocol stack to perform node profiling, which helps identifying the suitable set of nodes for further routing decision. Furthermore, in contemporary networks, the presence of malicious nodes or intruder to imposes significant (mischievous caused)

losses in terms of denial or service, flooding, or sometime irregular MAC information flow. This as a result force participating nodes to make wrong routing decisions and hence violates QoS performance by the network. Though, the classical researches have considered it as a distinct research problem. However, at the cost of increased complexity and cost. To alleviate this problem in this paper, we designed MHIR-SRmW as cumulative solution having capability to address joint MAC and routing problem while ensuring optimal QoS delivery. To achieve it, at first, MHIR-SRmW model obtains the different node information from the different layers to perform node profiling. It considers the following key parameters:

- IEEE 802.15.4 MAC information,
- Traffic overflow, and
- Topology information.

A brief of these parameters is given in the subsequent sections.

3.1.1 IEEE 802.15.4 MAC Information

In typical mobile-WSN, the nodes might undergo continuous topological changes, and hence, other dynamism like congestion, out-of-range problem, and frequent link-outage. Consequently, it might also cause packet loss at the MAC 802.15.4. In general, this kind of packet loss takes place because of the lack of MAC information at the sender node. The insufficient knowledge about the peer nodes often forces the participating nodes to undergo reduced packets and hence depleted performance. On the other hand, in case of intrusion, the anomalies can propagate false information that results into compromised transmission and hence violates the QoS performance. In numerous cases due to malicious nodes or even hardware malfunction, nodes can undergo nonlinear MAC transmission. Therefore, understanding such behaviour can help avoiding such malicious or fault-prone node(s) to participate forwarding path. Typically, in WSN, a transmitter node transmits packets as multicast and collects link-layer acknowledgement from the target receiver as well as other neighbouring node(s). In case of irregular probe signal and allied acknowledgement across peers might cause wrong routing decision and hence can impact QoS performance. In practice, to remain in network, each participating node in WSN requires transmitting a beacon message at certain predefined interval. However, it turns into energy as well as resource exhaustion. To alleviate this problem, in this work, we scheduled the beaoning interval at 10 ms, which is sufficient for a real-time WSN communication, without undergoing stale transmission. Thus, after each 10 ms interval, the participating nodes exchange their information including link quality, congestion information at the MAC which helps deciding the routing path(s). Here, our proposed MHIR-SRmW protocol receives the expected number of beacon message or packets from the neighbour node(s) and estimates the link quality dynamically to perform further relaying decision. Thus, identifying any irregularity in MAC information from any neighbouring node and higher packet loss pattern, our proposed MHIR-SRmW protocol classifies that node as the malicious node and excludes that

specific node from any future path formation. Moreover, in case a node doesn't responds the request beacon message (say, HELLO message), the node classifies that node as the malicious or misbehaving nodes, and thus excludes that for further routing purpose. Thus, in this approach, a participating node X examines the likelihood that a node can be able to transmit the data successfully to Z node or not. Here, we estimated the likelihood of successful transmission using (1).

$$P_M = \frac{\xi_{Rx(t_{i-1}, t_i)}}{\xi_{Exp(t_{i-1}, t_i)}} \quad (1)$$

In (1), the parameter ξ_{Rx} and ξ_{Exp} represent the total number of beacon message received, and the total number of expected beacons during the time-interval (t_{i-1}, t_i) .

In addition to the probability of successful transmission by a node, our proposed model estimated two other MAC parameters, first dynamic link quality and second the cumulative congestion probability. Similar to our previous work, JSMCRP [], in MHIR-SRmW protocol as well we estimated dynamic link quality of a node during the period (t_{i-1}, t_i) , using Eq. (2).

$$PDR_{ij} = \frac{P_{Rx}}{P_{Tx}} \quad (2)$$

In (2), the parameter P_{Rx} represents the total packets received, while P_{Tx} presents the total number of packets transmitted by i th node to the destination j th node. Additionally, our proposed MHIR-SRmW protocol model applied Eq. (3) to estimate the dynamic link quality of each participating node in the deployed network.

$$\beta_{DLQI} = \mu * \beta_{DLQI} + (1 - \alpha) * (PDR_{ij}) \quad (3)$$

In (3), β_{DLQI} represents the dynamic link quality of the i th node, while μ be the network coefficient that often varies in the range of 0–1. In addition to the transmission probability value and dynamic link quality at the MAC layer, the probability of congestion too can be estimated to perform node profiling. In this reference, our proposed MHIR-SRmW protocol exploited the information like the maximum buffer capacity and current buffer availability to assess congestion probability of a node. Recalling the fact that in mobile-WSN due to network dynamism or topological changes, there can be the iterative congestion at the participating nodes, we estimated cumulative congestion degree using Eq. (4) and (5).

$$CD_F = \frac{CD_{RPD} + CD_{NR2D}}{CD_{RPD_Max} + CD_{NR2D_Max}} \quad (4)$$

$$CD_r = \sum_{i=1}^N CD_{Fi} \quad (5)$$

In above derived model (4), the parameters, CD_{RPD} and CD_{RPD} represent the buffer available for the real-time traffic and the buffer available for non-real-time traffic, respectively. Similarly, CD_{RPD_Max} and CD_{NR2D_Max} are the maximum buffer available for the real-time data and the non-real-time data, respectively. The cumulative congestion of a node during the time-interval (t_{i-1}, t_i) is given by CD_r . Thus, applying above methods, our proposed MHIR-SRmW routing protocol estimated the key MAC information including successful transmission probability (1), dynamic link quality, (3) and cumulative congestion degree of a node (5). These MAC parameters are used as node profile variables for further computing.

3.1.2 Traffic Overflow

Due to continuous topological changes and being greedy in nature, mobile-WSN often undergoes the condition where a node might suffer abrupt payload rise. Such events can be undeniable for a normal node as well. However, there can be the malicious nodes which might intentionally uphold the data in buffer that can cause data flooding. Additionally, a malicious or malfunctional node might transmit burst data and thus can cause overflow. Similarly, due to improver transmission control at the PHY layer, there can be burst transmission causing traffic overflow or flooding. Such events eventually cause data drop and hence violates QoS performance. Considering this fact, in our proposed MHIR-SRmW protocol, we intend to consider such behavioural pattern as the variable to perform routing decision. Here, we hypothesize that a node with iterative flooding and irregular holding period can be defined as a malicious or malfunctional node. Thus, identifying such nodes, it can be secluded from any forwarding path formation.

In MHIR-SRmW protocol, the transmitting node monitors the load-traffic which behaves by the participating nodes and characterizes it as the normal node or the malicious node. Here, we obtained a parameter called queue length at the MAC layer and multicast it as ACK across the neighbouring nodes. Now, consider that i be the neighbouring node, while l_j be the j th sample value representing the queue length during time-interval (t_{i-1}, t_i) . Now, with queue length of L , we estimated the mean traffic load at the participating node as per (6).

$$T_{load_i} = \frac{1}{L} \sum_{j=1}^N l_j \quad (6)$$

Let l_{max} be the highest queue length of a participating node (at MAC layer), the total traffic density is estimated using (7).

$$T_{loadDens_i} = \frac{T_{load_i}}{l_{max}} \quad (7)$$

Thus, the probability of successful transmission can be obtained as per P_{Succ_i} following (8).

$$P_{\text{Succ}_i} = [1 - T_{\text{loadDens}_i}] \quad (8)$$

Since, the result of (8) is directly related to the transmission delay and energy consumption, and therefore, our proposed MHIR-SRmW protocol considered a node with minimum queue length for forwarding node selection. Thus, estimating above stated MAC information from IEEE 802.15.4 protocol stack of each node, MHIR-SRmW protocol executes heuristic concept to perform best forwarding path to guarantee QoS in WSN.

3.1.3 Topology Information

Recalling the fact that in mobile-WSN due to random movement pattern, there can be the situation of overhearing which might force the node(s) to make redundant communication and hence delay and/or QoS violation. Moreover, a malfunctional node or intruder can also create overhearing condition such as reply attack in WSN which can cause wrong transmission decision. To alleviate it, MHIR-SRmW protocol intends to deploy network in such manner that during data transmission to the next hop node, it doesn't create any overhearing condition in vicinity. If a node is able to overhear the packet forwarding from the one hop-distant node, it is labelled as the normal node. On the contrary, a node creating iterative beaconing is classified as malicious node. Thus, when a transmitter node is unable to overhear the retransmission of its packet when the destination node is unreachable because of the stale or repeated routing information, the corresponding forwarding node is identified as a malicious node. Thus, in reference to these information, MHIR-SRmW estimated a factor called trustworthiness for each participating node using Link-Index Change Rate (LICR), estimated as per (9) for i th node.

$$\eta_i = \gamma_i + \delta_i \quad (9)$$

In (9), the parameter γ_u states the rate of arrival, while δ_i presents the rate of link-outage by the i th node. Noticeably, the highest feasible rate of arrival γ_{i_Max} should be same as the rate of link-outage, and thus the highest link-outage (δ_{i_Max}) can be estimated as $\gamma_{i_Max} + \delta_{i_Max} = 2.\sigma_i$ [36]. Thus, the change in link quality or LICR can be estimated using (10).

$$\eta = \frac{\gamma_i + \delta_i}{2.\sigma_i} \quad (10)$$

In this manner, once estimating (10), MHIR-SRmW protocol re-estimated the likelihood of successful transmission using (11).

$$P_{\eta} = 1 - \eta \quad (11)$$

Thus, in reference to the above derived parameter, node with high link change LICR can be avoided during forwarding path selection. Now, once the proposed model has estimated above stated parameters such as the likelihood of successful transmission, dynamic link quality, cumulative congestion, change in link quality for each participating node, each node was characterized for its suitability to become the forwarding node. The risk analysis and labelling were performed using Eq. (12).

$$\text{Node}_{\text{sel}} = f \left[\left(\min_{i \in N} P_{\eta} \right), \left(\min_{i \in N} \text{CD}_r \right), \left(\max_{i \in N} P_{\text{Succ}_i} \right), \left(\max_{i \in N} \beta_{\text{DLQI}} \right) \right] \quad (12)$$

In MHIR-SRmW protocol, each transmitting node applies the node profile function (12) to estimate the suitability of a node to become the forwarding node.

A. **Heuristic driven multi-path availability-based recovery strategy (HMPARS)**

Though, the above stated model (12) can be more effective in reference to JSMCRP model []. However, so far could not address the sudden loss of link, as all the decision variables have been estimated for the time period of (t_{i-1}, t_i) . On the contrary, in real-time application environment, there can be the abrupt node death or link-outage and hence guaranteeing reliable recovery path is equally important. In some of the existing papers, authors have applied shortest distance path method(s) to define recovery path. However, it doesn't guarantee QoS or allied reliability in case of reiterating node death or link-outages towards the forwarding path. Practically, such events can be common due to physical damage or intrusion. Considering this fact, in this paper, our proposed MHIR-SRmW protocol introduces a state-of-the-art new and robust heuristic-driven multi-path availability-based recovery strategy (HMPARS) model that exploits above derived parameters to identify the suitable forwarding nodes and generates multiple recovery paths while ensuring no connected components.

Our proposed HMPARS model exploits above stated node information along with node availability to generate multiple disjoint paths for data transmission. In this mechanism, the second disjoint path remains on "Stand-By" condition and when a transmitter node realizes any abrupt link-outage or loss, it executes the recovery path without undergoing node discovery. Thus, it achieves timely and reliable data transmission without imposing addition computation, energy or resource exhaustion, and delay. The proposed HMPARS model at first exploits (12) to identify the most suitable nodes for recovery or forwarding path selection, and then exploits their link availability information to perform final forwarding path decision. By doing so, it not only alleviates the possibility of any future link-loss but also guarantees that the recovery paths remain active with minimum or not shared component to meet QoS communication demands. Once identifying and segmenting the set of suitable nodes (fulfilling the condition in (12)), our proposed HMPARS model estimates link availability of each node characterizing

the time for which the node remains connected to the peer nodes. In other words, it assesses whether the participating nodes can remain in connection during (t_{i-1}, t_i) to complete the transmission. Thus, a node with higher connectivity or availability would be considered for the recovery path formation. Moreover, it considers that the forwarding paths involving minimum distance (source to destination) and no connected component can be considered as the best forwarding path to ensure QoS. Here, we intended to achieve two disjoint paths with higher connectivity and no shared component. Thus, employing the suitable set of nodes (in reference to (12)), our proposed HMPARS model performs the best disjoint path selection while fulfilling the criteria of (12), high connectivity, and availability with no shared component. This problem resembles a typical NP-hard problem or convexity problem, which can be solved by any heuristic method. In this reference, we developed a particle swarm optimization (PSO) algorithm that exploits above parameters and identifies a set of forwarding paths with high connectivity (no link-loss or connectivity loss) and no shared component. In our proposed HMPARS model, we employed first-order approximation approach to obtain node or path unavailability, as the sum of unavailability of all connected nodes. In this work, we executed the proposed HMPARS model as Monte Carlo simulation that helped in calculating the dynamic topology and allied network conditions. It also helped in deploying the network as probabilistic network, and hence, we applied Bayesian network model to deploy mobile-WSN over defined geographical region.

B. Link Connectivity Estimation

Typically, in wireless communication, node connectivity states the likelihood that minimum one node path in between source and destination is present. In this reference, a sensor node n_0 can be connected to the recovery path only when it (i.e. n_0) is active in conjunction with minimum one path joining source to the destination. Towards QoS-oriented communication in WSN, our proposed MHIR-SRmW protocol hypothesizes that each node possesses two disjoint forwarding paths, with no shared component(s). Consider that the forwarding paths for n_0 be $\mathcal{P}_0, \dots, \mathcal{P}_{K-1}$, while $\overline{\mathcal{P}}_k$ be the connections with \mathcal{P}_k , then the connected path can be obtained as per (13).

$$C(n_0) = \mathbb{A}(n_0) \mathbb{A} \left(\bigcup_{k=0}^{K-1} \overline{\mathcal{P}}_k \right) \mathbb{A}(C) \quad (13)$$

In (13), $\mathbb{A}(\bigcup_{k=0}^{K-1} \overline{\mathcal{P}}_k)$ represents the path availability or the set of forwarding paths available for recovery. Despite the active terminal node, the connectivity of n_0 might undergo loss condition when the route $\overline{\mathcal{P}}_k$ fails. Thus, hypothesizing that the node and its allied link-outage are independent in nature, we estimate the link availability as per (14).

$$\mathbb{A}\left(\bigcup_{k=0}^{K-1} \overline{\mathcal{P}_k}\right) = 1 - \prod_{k=0}^{K-1} U_r(\overline{\mathcal{P}_k}) \quad (14)$$

Consider that the network encompasses $\{n_{k,0}, \dots, n_k, f_k\}$ sensor nodes and their corresponding link in path k be $\{e_{k,1,2}, \dots, n_k, f_{k-1}, f_k\}$ preconditioned at, $n_{k,0} = n_0, n_k = f_k C$, then link availability is estimated as per (15).

$$\begin{aligned} U_r(\overline{\mathcal{P}_k}) &= 1 - \mathbb{A}_r(\overline{\mathcal{P}_k}) \\ &= 1 - \prod_{i=1}^{f_{k-1}} \mathbb{A}_n(n_{k,i}) \prod_{j=0}^{f_{k-1}} \mathbb{A}_e(e_{k,j,j+1}) \end{aligned} \quad (15)$$

Now, employing above derived link availability model Eq. (13–15), the link connectivity for the transmitter n_0 is obtained using (16).

$$C(n_0) = \mathbb{A}(n_0)\mathbb{A}(C) \times \left(1 - \prod_{k=0}^{K-1} \left(1 - \prod_{i=1}^{f_{k-1}} \mathbb{A}_n(n_{k,i}) \prod_{j=0}^{f_{k-1}} \mathbb{A}_e(e_{k,j,j+1}) \right) \right) \quad (16)$$

Thus, employing above parameter MHPARS model executes PSO algorithm to identify the set of disjoint paths with maximum link availability using the nodes identified as suitable as per (12). In other words, in the proposed model, we apply (17) to identify the optimal set of nodes to become the forwarding member. However, once identifying the suitable nodes, MHPARS model exploits further link availability information using Eq. (13–16) and executes PSO to obtain the best set of disjoint forwarding paths with no shared component.

$$\begin{aligned} C(n_0) &= \prod_{j \in \Phi_n}^f \mathbb{A}_n(n_j) \prod_{k \in \Phi_e}^{f-1} \mathbb{A}_e(e_{k,k+1}) \\ &\times \left(1 - \left(1 - \prod_{i \in \Phi_{n,0}} \mathbb{A}_n(n_{0,i}) \prod_{j \in \Phi_{e,0}} \mathbb{A}_e(e_{0,j,j+1}) \right) \right) \\ &\times \left(1 - \prod_{i \in \text{int}\Phi_{n,1}} \mathbb{A}_n(n_{1,i}) \prod_{j \in \Phi_{e,1}} \mathbb{A}_e(e_{1,j,j+1}) \right) \end{aligned} \quad (17)$$

To identify the disjoint paths with no shared components, we hypothesize that the node connectivity with disjoint nature can be achieved by decoupling the shared component's importance and allied links. Consider, \mathcal{R}_0 and \mathcal{R}_1 be the two forwarding paths, then the allied link connectivity can be obtained as per (17). In (17), the parameters Φ_n and Φ_e state the sets of shared components. On the contrary, the set of unshared components be $\Phi_{n,i}$ and $\Phi_{e,i}$ over i th path. Now, applying first-order approximations in terms of unavailability, we obtain the connectivity loss as (16).

$$L(n_0) = 1 - C(n_0) \quad (18)$$

$$L(n_0) \approx \sum_{j f \Phi_n}^f U_n(n_j) + \sum_{k f \Phi_e}^{f-1} U_e(e_{k,k+1}) + \left(\sum_{i f \Phi_{n,0}} U_n(n_{0,i}) + \sum_{j f \Phi_{e,0}} U_e(e_{0,j,j+1}) \right) \\ \times \left(\sum_{i f \Phi_{n,1}} U_n(n_{1,i}) + \sum_{j f \Phi_{e,1}} U_e(e_{1,j,j+1}) \right) \quad (19)$$

Now, the probability that the unshared links or path which doesn't impact the retransmission probability or link-loss is obtained as per (20).

$$L(n_0) \approx \sum_{j f \Phi_n}^f U_n(n_j) + \sum_{k f \Phi_e}^{f-1} U_e(e_{k,k+1}) \quad (20)$$

Observing above derived models (19) and (20), it can be found that a QoS-oriented recovery path pair can be designed without applying any shared component. To achieve it, we applied PSO algorithm which exploited link availability (or link-loss probability) as the cost function (21) to perform disjoint recovery path estimation to guarantee QoS performance. Due to space constraints, the detailed discussion of PSO algorithm is not given in this paper.

In HMPARS, PSO intends to retain dual disjoint paths in such manner that it retains minimum hops, low link connectivity loss probability, and no shared component. PSO algorithm is executed overall all possible paths and process continues by adding a new hop sensor node (following (12)) with minimum link-loss probability and no shared component. This process continues till the likelihood of achieving superior path turns out to be very low. This process employs two iterative mechanism, path selection, and pruning. Once selecting the suitable forwarding path over certain iteration k , the other path(s) from S_k possessing low-cost function or poor link connectivity are pruned or removed. It helps not only in achieving QoS but also reduced computational costs and allied resource exhaustion. MHPARS model applied the cost function $c(\mathcal{P})$ for each possible path \mathcal{P} using (21).

$$\mathcal{P}^* = \arg \min_R c(\mathcal{P}) \quad (21)$$

Now, let $\bar{\mathcal{P}}$ be the forwarding path formed with zero link-unavailability. Then, the path R can be connected to the node n_f , and thus for any complete forwarding path $M_i \in S_k$, $L(\bar{\mathcal{P}}, M_i)$ be the connectivity loss for the source node n_0 . Thus, the mean connectivity loss is estimated using (22).

$$\tilde{L}(\mathcal{P}) = \frac{1}{N_c} \sum_{i=1}^{N_c} L(\bar{\mathcal{P}}, M_i) \quad (22)$$

This is the matter of fact that in case of dynamic networks like mobile-WSN the link-loss probability and so for the path \mathcal{P} . In this reference, the cost function as derived in (22) can be reframed as (23).

$$c(\mathcal{P}) = \tilde{L}(\mathcal{P}) + E(\mathcal{P}) \quad (23)$$

In above derived function (23), $E(\mathcal{P})$ is obtained based on the average loss caused per link across the path pairs. In other words, $E(\mathcal{P})$ is estimated as per (24).

$$E(\mathcal{P}) = \frac{1}{N_c} \sum_{i=1}^{N_c} E(\mathcal{P}, M_i) \quad (24)$$

where

$$E(\overline{\mathcal{P}}, M_i) = \frac{\tilde{L}(M_i)}{\lambda} d(n_{\mathcal{P}}, n_f) \quad (25)$$

Now, to estimate the distance values, graph theory is applied representing a graph matrix A possessing varied components a_{ij} where $a_{ij} = 1$ when the link between node i to node j is active. Otherwise, it follows $a_{ij} = 0$ and $a_{ii} = 1$. In this reference, it estimates a matrix $B(k)$, which is defined as (26).

$$B(k) = \mathbb{A}^k \quad (26)$$

In above expression (26), the parameter $B(k)$ possesses $b_{ij}(k)$ which is same as the total paths to reach j from i with hops lower than k . Thus, with $b_{ij}(k) = 0$, there would not be the other feasible path approaching j from i node in k -hops. In our proposed model, we estimated the distance between i to j node as the shortest path, which can be estimated as per (27).

$$d(i, j) = \min_{b_{ij}(k) > 0} \{k\} \quad (27)$$

The model derived above (27) states that $d(i, j)$ can have the minimum value of k -hops when $b_{ij}(k) > 0$. Thus, employing (19) and (27), with high link availability, our proposed HMPARS model obtained two disjoint paths with no shared components. In our proposed MHIR-SRmW protocol, once estimating the values for (25), all paths available in S_k are updated proactively. Thus, during transmission in case MHIR-SRmW protocol identifies any fault or sudden link-loss with "0" link connectivity, it switches to the available disjoint path and completes the transmission without undergoing node discovery and recovery path estimation. This as a result helped ensuring optimal QoS assurance for mobile-WSN.

4 Results and Discussion

In this paper, we developed a state-of-the-art new and robust QoS-oriented routing protocol that exploits cross-layer information in IEEE 802.15.4 protocol stack to ensure reliable transmission even under different topological conditions, including link-outage or continuous mobility. Realizing the up-surging significance of mobile-WSN in contemporary wireless communication techniques such as IoTs and M2M communication, we focused on both reliability as well as fault-resilient routing strategy so as to guarantee QoS in mobile-WSN. To achieve it, in this paper, MHIR-SRmW protocol was developed that encompassed two key functions, first, multi-layered hierarchical dynamic information-based node profiling, and second heuristic-driven multi-path availability-based recovery strategy (HMPARS). Conceptually, the proposed model hypothesizes that retaining only reliable nodes for forwarding path selection can help avoiding packet loss, while designing dual disjoint path with no shared components or nodes can help achieving optimal performance even under link-outage probability or attack condition. Unlike our previous work JSMCRP, the proposed MHIR-SRmW protocol exploits more significant features to perform node profiling followed by forwarding node selection and heuristic-driven dual disjoint forwarding path selection. To achieve it, at first, we obtained key MAC information including cumulative congestion of the participating nodes, dynamic link quality, and successful packet transmission probability. Applying these parameters, the nodes with the higher rank values were considered as the forwarding node candidates. Subsequently, our proposed MHIR-SRmW protocol executed HMPARS model that intends to identify or select two disjoint paths in such manner that the link (say, paths) possesses higher link availability (i.e. no link-loss probability) and minimum distance between source and destination. Additionally, it also intends to design only those two disjoint paths which don't have any common node or shared component(s). To achieve it, we applied PSO heuristic algorithm. In our proposed HMPARS model, PSO exploits link availability information with minimum hop while ensuring no shared component(s) to perform disjoint path selection. Here, we developed only two disjoint paths to ensure reliable communication. In this reference, in our proposed MHIR-SRmW protocol, a transmitter node at first identifies the set of best forwarding nodes using (12) and then executes HMPARS algorithm to select two disjoint paths with no shared component. During transmission, in case it identifies any node death or link-outage due to sudden link-loss or node death (caused due to physical damage or any possible intrusion attack), it selects the second alternative recovery path or the second disjoint path to transmit the remaining data without performing node discovery or forwarding path selection. In this manner, it reduces the delay and computational overhead. To select the alternate disjoint forwarding path for remaining data transmission, we applied logical AND function. The beaconing interval was maintained for 10 ms that at one hand reduces iterative beaconing and hence makes mobile-WSN communication more resource efficient. In this manner, our proposed MHIR-SRmW protocol intended to achieve reliable and QoS-oriented communication to meet mobile-WSN communication demands. The

overall proposed model was developed using Network Simulator platform, where the algorithms were developed in CPP programming language. The model was simulated over Ubuntu 14 version over the CPU armoured with 8 GB RAM. The experimental setup considered for the simulation is given in Table 1.

To assess relative performance by our proposed MHIR-SRmW protocol, we considered our previous work JSMCRP model [] as reference. Similar to the proposed MHIR-SRmW protocol, JSMCRP protocol employs Application Layer, Network Layer, MAC Layer, and PHY Layer to perform Network Adaptive MAC scheduling and Dynamic Routing Decision. JSMCRP employs Data Traffic Assessment, Prioritization and Scheduling (DTAPS), Proactive Network Monitoring and Knowledge (PNMK), Dynamic Congestion Index Estimation (DCIE), Adaptive Link Quality, Dynamic Packet Injection Rate (DPIR), and Cumulative Rank Sensitive Routing Decision (CRSRD) to perform routing decision. Thus, exploiting the dynamic values of the cumulative congestion degree, packet injection rate, and dynamic link quality information, our previous work JSMCRP performed best forwarding node selection and completed the transmission. The packet delivery rate (PDR) performance by

Table 1 Experimental setup

Parameter	Value
Number of nodes	100
Node density	10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 nodes (each autonomous simulation)
Network dimension	100×100
MAC	IEEE 802.15.4MAC
PHY	IEEE 802.15.4 PHY
Radio	100 m
Transmission rate (BPS)	10–512 p/s
Career frequency	2.5 GHz
Antenna	Omni directional
RF power amplifier	0.50
Link margin	45 dB
Gain factor	35 dB
Power density of radio channel	–130 dBm/Hz
Noise at the receiver	10 dB
BER performance	10^{-3}
Power consumption at the transmitter	98.2 milli-watts
Antenna gain	5 dB
Packet size	512 kb
PSO population size	20
Stopping criteria	Generations (100)
Fitness value	Link connectivity and minimum hops

JSMCRP was almost 99% that signifies its robustness towards QoS-oriented communication. However, JSMCRP model didn't consider any sudden link-outage probability and its consequence on the performance. On the contrary, the work proposed in this paper (i.e., MHIR-SRmW protocol) considers link-outage adaptive routing to guarantee QoS provision. To assess efficacy of the proposed MHIR-SRmW protocol as well as JSMCRP, we introduced predefined node-death scenario and accordingly their performance towards complete data transmission was examined. We estimated four key performance parameters, packet delivery rate, packet loss rate, latency, and energy consumption to assess relative performance by our proposed MHIR-SRmW protocol and previous work JSMCRP protocols. The simulated results and allied inference are given as follows.

Being a mobile topology network, mobile-WSN might undergo high congestion, link-outage probability, etc. Its severity might increase as per rise in network density. In other words, with a large number of autonomously communicating nodes, the likelihood of congestion and hence packet loss might increase. On the other hand, with the large number of nodes in a dense mobile-WSN network, the search space for HMPARS might have to process large data, and hence, it might be exhaustive especially in terms of delay, packet loss, and retransmission (hence high energy consumption). Considering this hypothesis, in this study, we examined whether our proposed MHIR-SRmW protocol alleviates above stated problem and yields timely data delivery while guaranteeing minimum delay, higher throughput, and minimum energy consumption. Achieving such performance might help accomplishing QoS performance. In this reference, we simulated the proposed MHIR-SRmW protocol as well as JSMCRP model independently with the different node density. Here, we simulated the proposed models with the different node density (i.e. 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 nodes). Moreover, we deployed random link-loss in simulation, even though the number of faults or faulty node considered (or deployed) was one. Thus, once simulating the proposed methods (i.e. MHIR-SRmW protocol and the previous work JSMCRP), the node death was identified in the forwarding path and subsequently the efficiency of the proposed model was examined towards QoS performance. To estimate the packet delivery rate (PDR), we applied the classical equation representing the ratio of the total received data to the total transmitted data. On the contrary, the packet loss was estimated as $(100 - \text{PDR} (\%))$. Towards energy estimation, we considered the classical energy model, where the key energy consumption took place at the time of node discovery and practice table estimation towards dual disjoint path estimation (with no shared component). Additionally, the energy exhaustion took place due to the power amplifier and the per-bit-transmission cost (mJ). The simulation parameters used are given in Table 1.

Figure 1 presents the packet delivery ratio (PDR) performance by our proposed MHIR-SRmW protocol and the previous contribution JSMCRP []. This is the matter of fact that the previous work JSMCRP had exhibited almost 99% of the successful packet delivery or PDR. However, there were no fault or node-death case during simulation. In other words, JSMCRP didn't consider link-outage probability and resulting recovery path estimation. In JSMCRP, we merely had estimated node parameters to decide best forwarding node and based on that a forwarding path was defined. On

the contrary, in case of fault presence or node death condition, JSMCRP is to identify the best suitable alternate path for recovery transmission. In JSMCRP, it can be achieved by only performing node discovery (post node death) and best forwarding path selection. This process can not only cause data drop but can also impose congestion, delay, and energy consumption. Therefore, JSMCRP might suffer packet losses. This fact is quite visible through the simulation results (Figs. 1 and 2). Observing the results, it can be found that the proposed MHIR-SRmW protocol which embodies the self-configuring dual disjoint forwarding path during node death achieves superior performance than JSMCRP. The depth assessment reveals that the proposed MHIR-SRmW protocol performs average PDR of 97.6%, while JSMCRP could achieve the average performance (i.e. PDR) of 86.8%, which is significantly lower than the proposed model. Though, one interesting outcome can be observed that even with the large number of nodes, both JSMCRP as well as the proposed MHIR-SRmW protocol retains near-stable PDR performance. This case be because of the proactive network management capability. The packet loss rate (PLR) too indicates (Fig. 2) that the proposed MHIR-SRmW protocol exhibits lower PLR than the JSMCRP mobile-WSN protocol.

In real-time communication across the WSN application environment, guaranteeing timely data transmission is inevitable. On the contrary, delay or latency might be caused due to higher computation, packet loss, retransmission etc. Though, above results confirm that the proposed MHIR-SRmW protocol exhibits lower PLR and hence low retransmission and hence can be expected to exhibit low latency. However, both JSMCRP as well as MHIR-SRmW protocol might have to execute alternate forwarding path selection process during link-loss. This process can be time-consuming. Realizing this fact, we examined our proposed MHIR-SRmW protocol as well as the previous work, JSMCRP in terms of their corresponding time-efficiency. To assess latency over a continuous channel access period of 60 s, we simulated our proposed model with 60 number of nodes. The simulation results with JSMCRP as well as the proposed MHIR-SRmW protocol is shown in Fig. 4. Observing the results in Fig. 4, it can easily be found that the proposed MHIR-SRmW protocol performs

Fig. 1 PDR Vs node density

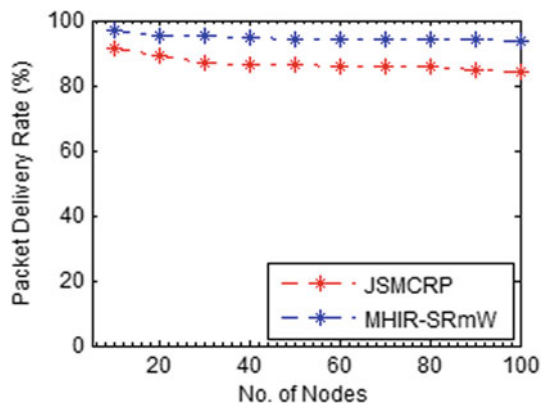
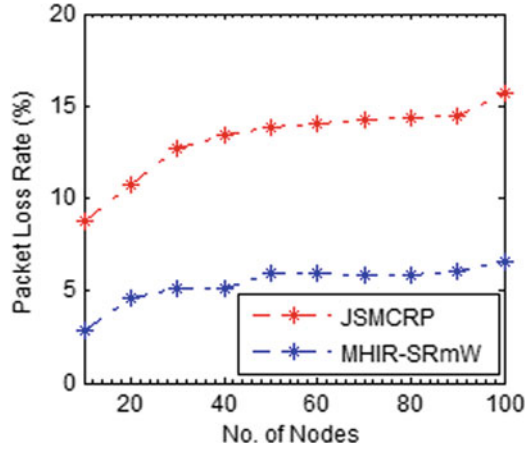


Fig. 2 PLR versus node density



significantly smaller delay in comparison with the previous work of JSMCRP. Here, the key reason can be the network discovery and best forwarding path formation cost in JSMCRP. On the contrary, our proposed MHIR-SRmW protocol achieves the dual disjoint forwarding path at the start of the network only, and hence once identifying any link-outage or node death, it switches to the disjoint forwarding path without undergoing node discovery or allied process. It makes our proposed MHIR-SRmW protocol more time efficient (Figs. 3 and 4). The results obtained (Figs. 3 and 4) reveals that though both JSMCRP as well as our proposed MHIR-SRmW protocol exhibits same delay at the start of simulation; however, once detecting the node death and link-loss, JSMCRP undergoes higher delay, due to data loss caused retransmission, node discovery process and another forwarding path estimation.

This is the matter of fact that the energy consumption is directly related to the retransmission or packet loss rate and computation. Though, MHIR-SRmW protocol

Fig. 3 Delay versus node density

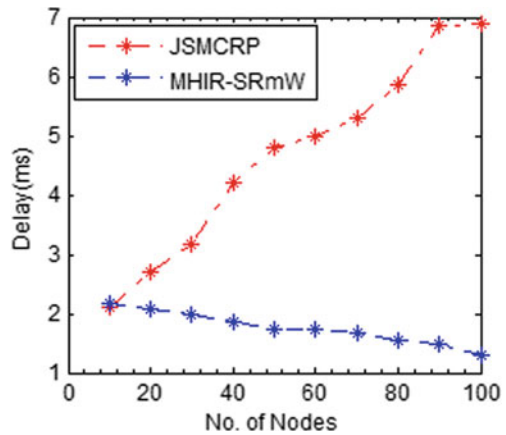
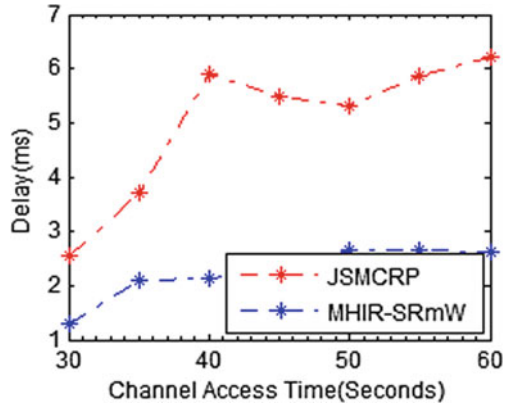
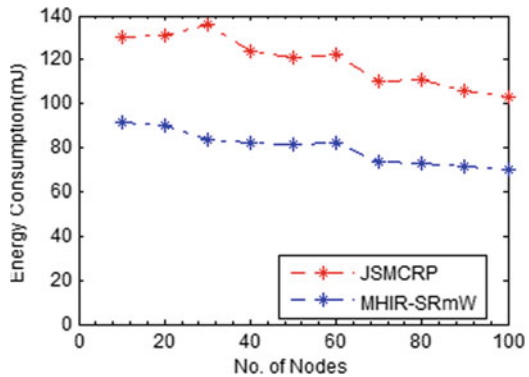


Fig. 4 Delay over channel access period (s)



employs more computing elements. However, its higher PDR performance enables it to reduce the energy consumption. In this reference, the result obtained (Fig. 5) reveals that the proposed MHIR-SRmW protocol exhibits less than 92 mJ power consumption even with high density network. On the contrary, despite of low-computational exhaustive, JSMCRP underwent higher power exhaustion or consumption due to retransmission. In reference to the PLR performance (Fig. 2), the retransmission probability is higher in case of JSMCRP, and hence, it undergoes higher power consumption (Fig. 5). Observing overall performance in terms of high packet delivery, low packet loss, low delay, and fault-resiliency, the proposed MHIR-SRmW protocol accomplishes QoS performance. The results obtained signify that the proposed model can be applied in real-time mobile-WSN application, where it can guarantee both network reliability as well as statistical performance to meet application's demands. The overall research conclusion and allied inferences are given in the subsequent section.

Fig. 5 Energy consumption versus node density



5 Conclusion

This paper proposed a state-of-the-art new and robust multi-level hierarchical information-driven risk-sensitive routing protocol for mobile-WSN (MHIR-SRmW). The proposed MHIR-SRmW protocol emphasizes on guaranteeing QoS under uncertain network conditions such as sudden physical damage of node(s), link-outage, etc., under dynamic topology. Moreover, it also intended to ensure that the nodes participating the forwarding path are reliable. In this reference, the proposed MHIR-SRmW protocol at first perform multi-layered hierarchical dynamic information-based node profiling by exploiting MAC layer information such as congestion probability, probability successful delivery, dynamic link quality, etc. It helps in segmenting the most suitable set of nodes to be considered by a transmitter node to complete data transmission. Here, the use of above stated parameters such as link quality, congestion probability, node's ability to serve successful data transmission, flooding behaviour, etc., ensures that the nodes selected towards forwarding path would be reliable enough to meet QoS-centric communication. On the other hand, the proposed MHIR-SRmW protocol executes heuristic-driven multi-path availability-based recovery strategy (HMPARS) to identify the set of disjoint paths to guarantee reliable transmission even under sudden node death or link-outage. The proposed HMPARS model exploits link connectivity and minimum hops as the cost function to form the dual disjoint paths for data recovery or data forwarding under link-outage. Additionally, HMPARS model forms dual disjoint paths while ensuring that the paths formed doesn't carry any shared components. Here, the key intend was to avoid any iterative fault or link-outage and ensure reliable transmission even under node death or complete path unavailability. It can be highly efficient under the condition where the nodes might undergo any external attack or physical damage. The performance comparison with the existing approaches like JSMCRP revealed that the proposed MHIR-SRmW protocol achieves superior (average) PDR of 97.6%, while retaining packet loss lower than 3%. On the contrary, due to lack of disjoint path and iterative forwarding path selection, JSMCRP protocol underwent higher packet loss (average 13.2% PLR) and lower PDR (86.8%) performance. Despite the proactive network management ability, JSMCRP underwent inferior performance due to lack of backup forwarding path. Since, MHIR-SRmW protocol identifies the backup or recovery disjoint path(s) at the time of network discovery only, it doesn't require executing node discovery again and again. This ability helped it to maintain low latency or delay. On the contrary, JSMCRP lacked this ability, which caused it to undergo higher latency or delay. The energy-performance which is highly related to the packet loss too affirms that due to lower packet loss, MHIR-SRmW protocol undergoes lower power exhaustion in comparison with JSMCRP protocol for mobile-WSN. The overall results confirm that the proposed MHIR-SRmW protocol is more capable to achieve QoS performance to meet contemporary communication demands. Though, both JSMCRP and MHIR-SRmW protocols intended to achieve QoS performance. However, could not address power transmission control, which can have the impact on overall efficiency including resource utilization, delay, etc. In future, efforts can be made towards

dynamic power management under uncertain or dynamic network conditions like mobile-WSN-oriented JSMCRP and MHIR-SRmW protocols.

References

1. Ehsan S, Hamdaoui B (2012) A survey on energy-efficient routing techniques with QoS assurances for wireless multimedia sensor networks. *Commun Surv Tutoriales IEEE* 14(2):265–278
2. Spachos P, Toumpakaris D, Hatzinakos D (2015) QoS and energy-aware dynamic routing in wireless multimedia sensor networks. In: 2015 IEEE international conference on communications (ICC), pp.6935–6940
3. Sen J Ukil A (2009) An adaptable and QoS-aware routing protocol for wireless sensor networks. In: 2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, pp 767–771
4. Khanke K, Sarde M (2015) An energy efficient and QoS aware routing protocol for wireless sensor network. *Int J Adv Res Comput Commun Eng* 4(7)
5. Lombardo L, Corbellini S, Parvis M, Elsayed A, Angelini E, Grassini S (2018) Wireless Sensor Network for Distributed Environmental Monitoring. *IEEE Trans Instrum Measure* 67(5):1214–1222
6. Boukerche A, Nelem Pazzi RW (2007) Lightweight mobile data gathering strategy for wireless sensor networks. In: 2007 9th IFIP international conference on mobile wireless communications networks, pp 151–155
7. de Araujo GM, Becker LB (2011) A network conditions aware geographical forwarding protocol for real-time applications in mobile wireless sensor networks. In: 2011 IEEE international conference on advanced information networking and applications, pp 38–45
8. Singh R, Rai BK, Bose SK (2016) A novel framework to enhance the performance of contention-based synchronous MAC protocols. *IEEE Sens J* 16(16):6447–6457
9. Shruti BV, Nagendrappa TM, Venkatesh KR (2021) JSMCRP: cross-layer architecture based joint-synchronous MAC and routing protocol for wireless sensor network. *ECTI Trans Electr Eng Electron Commun* 19(1):94–113.
10. Singh R, Rai BK, Bose SK (2017) A contention-based routing enhanced MAC protocol for transmission delay reduction in a multi-hop WSN. In: TENCON 2017—2017 IEEE Region 10 conference, Penang, pp 398–402
11. Singh R, Rai BK, Bose SK (2017) A joint routing and MAC protocol for transmission delay reduction in many-to-one communication paradigm for wireless sensor networks. *IEEE Internet Things J* 4(4):1031–1045
12. Arifuzzaman M, Dobre OA, Ahmed MH, Ngatched TMN (2016) Joint routing and MAC layer QoS-aware protocol for wireless sensor networks. In: 2016 IEEE global communications conference (GLOBECOM), Washington, DC, pp 1–6
13. Sefuba M, Walingo T (2018) Energy-efficient medium access control and routing protocol for multihop wireless sensor networks. *IET Wirel Sens Syst* 8(3):99–108
14. Chen S, Yuan Z, Muntean GM (2017) Balancing energy and quality awareness: a “mac-layer duty cycle management solution for multimedia delivery over wireless mesh networks. *IEEE Trans Vehicular Technol* 66(2):1547–1560
15. Haqbeen JA, Ito T, Arifuzzaman M, Otsuka T (2017) Joint routing, MAC and physical layer protocol for wireless sensor networks. In: TENCON 2017—2017 IEEE Region 10 conference, Penang, pp 935–940
16. Liu Y, Chen Q, Liu H, Hu C, Yang Q (2016) A non-destructive Interference based receiver-initiated MAC protocol for wireless sensor networks. In: 2016 13th IEEE annual consumer communications & networking conference (CCNC), Las Vegas, NV, pp 1030–1035
17. Liu Y, Liu H, Yang Q, Wu S (2015) RM-MAC: a routing-enhanced multi-channel MAC protocol in duty-cycle sensor networks. In: 2015 IEEE international conference on communications (ICC), London, pp 3534–3539

18. Mohapatra S, Mohapatra RK (2017) Comparative analysis of energy efficient MAC protocol in heterogeneous sensor network under dynamic scenario. In: 2017 2nd International conference on man and machine interfacing (MAMI), Bhubaneswar, pp 1–5
19. Khalil MI, Hossain MA, Haque MJ, Hasan MN (2017) EERC-MAC: energy efficient receiver centric MAC protocol for wireless sensor network. In: 2017 IEEE international conference on imaging, vision & pattern recognition (icIVPR), Dhaka, pp 1–5
20. Senthil T, Bifrin Samuel Y (2014) Energy efficient hop extended MAC protocol for wireless sensor networks. In: 2014 IEEE international conference on advanced communications, control and computing technologies, Ramanathapuram, pp 901–907
21. Lahlou L, Meharouech A, Elias J, Mehaoua A (2015) MAC-network cross-layer energy optimization model for wireless body area networks. In: 2015 International conference on protocol engineering (ICPE) and international conference on new technologies of distributed systems (NTDS), Paris, pp 1–5
22. Kalaivaani PT, Rajeswari A (2015) An analysis of H-MAC, HSMAC and H-MAC based AOMDV for wireless sensor networks to achieve energy efficiency using spatial correlation concept. In: 2015 2nd International conference on electronics and communication systems (ICECS), Coimbatore, pp 796–801
23. Bouachir O, Ben Mnaouer A, Touati F, Crescini D (2017) EAMP-AIDC—energy-aware mac protocol with adaptive individual duty cycle for EH-WSN. In: 2017 13th International wireless communications and mobile computing conference (IWCMC), Valencia, pp 2021–2028
24. Reddy PC, Sarma NVSN (2016) An energy efficient routing and MAC protocol for bridge monitoring. In: 2016 International conference on wireless communications, signal processing and networking (WiSPNET), Chennai, pp 312–315
25. Thenmozhi M, Sivakumari S (2017) Adaptive energy efficient and rate adaptation based medium access control routing protocol (AEERA—MACRP) for fully connected wireless ad hoc networks. In: 2017 8th international conference on computing, communication and networking technologies (ICCCNT), Delhi, pp 1–7
26. Cheng B, Ci L, Tian C, Li X, Yang M (2014) Contention window-based MAC protocol for wireless sensor networks. In: 2014 IEEE 12th international conference on dependable, autonomic and secure computing, Dalian, pp 479–484
27. Zhuo S, Wang Z, Song YQ, Wang Z, Almeida L (2016) A traffic adaptive multi-channel MAC protocol with dynamic slot allocation for WSNs. *IEEE Trans Mob Comput* 15(7):1600–1613
28. Rachamalla S, Kancharla AS (2015) Power-control delay-aware routing and MAC protocol for wireless sensor networks. In: 2015 IEEE 12th international conference on networking, sensing and control, Taipei, pp 527–532
29. Ananda Babu J, Siddaraju and Guru R (2016) An energy efficient routing protocol using RD-MAC in WSNs. In: 2016 2nd International conference on applied and theoretical computing and communication technology (iCATccT), Bangalore, pp 799–803
30. Wahid A, Ullah I, Khan OA, Ahmed AW, Shah MA (2017) A new cross layer MAC protocol for data forwarding in underwater acoustic sensor networks. In: 2017 23rd international conference on automation and computing (ICAC), Huddersfield, pp 1–5
31. Leao L, Felea V, Guyennet H (2016) MAC-aware routing in multi-sink WSN with dynamic back-off time and buffer constraint. In: 2016 8th IFIP international conference on new technologies, mobility and security (NTMS), Larnaca, pp 1–5
32. Seddar J, Khalifé H, Al Safwi W, Conan V (2015) A full duplex MAC protocol for wireless networks. In: 2015 international wireless communications and mobile computing conference (IWCMC), Dubrovnik, 2015, pp 244–249
33. Heimfarth T, Giacomini JC, de Araujo JP (2015) AGA-MAC: adaptive geographic anycast MAC protocol for wireless sensor networks. In: 2015 IEEE 29th international conference on advanced information networking and applications, Gwangju, pp 373–381
34. Akhtar AM, Behnad A, Wang X (2015) Cooperative ARQ-based energy-efficient routing in multihop wireless networks. *IEEE Trans Veh Technol* 64(11):5187–5197
35. Louail L, Felea V, Bernard J, Guyennet H (2015) MAC-aware routing in wireless sensor networks. In: 2015 IEEE international black sea conference on comm. and networking (BlackSeaCom), Constanta, pp 225–229

36. Wang X, Ding L, Bi D (2010) Reputation-enabled self-modification for target sensing in wireless sensor networks. *IEEE Trans Instrum Measure* 59(1):171–179

Target Classification Using CNN-LSTM Network with Reduced Sample Size in Surveillance Radar



Vinit R. Waingankar, Vijay Surya Vempati, Santhosh, and G. Malarkannan

1 Introduction

For any military force border surveillance is critical aspect to regulate line of control. Perimeter surveillance can be carried out by a modern radar [1, 2] for timely detection and tracking of targets. Target classification is vital in later stages of signal processing and has been carried out manually by operator by listening to doppler audios. This type of classification is tedious and varies from operator to operator. Artificial intelligence (AI) gained hype in the past decade due to the availability of larger datasets and faster hardware. The present work is a humble application of deep learning (a subdomain of AI) to automate the task of non-cooperative target classification based on radar backscattered signals.

There has been a lot of emerging research on target classification based on deep learning. Deep learning typically searches the data for patterns for predicting /classifying. A feed forward network lacks memory and is not suitable for prediction where time-dependency among data samples is present. They consider only present input sample for a prediction. For capturing time domain relationships, recurrent neural networks (RNN) come in handy as they have memory but do not have long-term memory [3].

V. R. Waingankar (✉) · V. S. Vempati · Santhosh · G. Malarkannan
Development & Engineering, Military Radars, Bharat Electronics Limited, Bangalore, India
e-mail: vinitrwaingankar@bel.co.in

V. S. Vempati
e-mail: vijaysuryavempati@bel.co.in

Santhosh
e-mail: santhoshwagle@bel.co.in

G. Malarkannan
e-mail: malarakannang@bel.co.in

They also suffer from what is known as vanishing/exploding gradient descent [4]. To overcome the above problem an ameliorated version of RNN, long-short term memory [5] is used. On the other hand, convolutional neural networks (CNN) are extensively used for image-classification, because of advantage of automatic-feature extraction [6].

Most target classification utilizing deep learning techniques have been implemented in various kinds of radars, namely, high-resolution range profile (HRRP) radars, micro-doppler radar, passive radar, synthetic aperture radar (SAR) based, and forward scattering radar (FSR). In these works, radar data generally has large number of features and undergoes various transformations after acquisition. For training any model, large amount of data is the primary requirement, and this is typically done by data simulating practical data [7–10]. With the advent of faster hardware, many classification algorithms are based on RNN or its enhanced version, LSTM to extract time-dependency information. LSTM has become an integral part for target classification in time-series data. The aim of present work is on maintaining degree of accuracy in classification as the input sample size to the LSTM network decreases.

In this work, target classification task is performed by utilizing the radar back-scattered signals also called as radar echoes of a ground surveillance radar. For training the model, time-series parameters under consideration are RCS, doppler frequency and velocity, generated by the mono-static radar for four different target classes. For testing, the block diagram of auto-classification steps is shown in Fig. 1. The echoes acquired by radar antenna follows a chain of processing before reaching the expert system as shown typical to any radar. The echo signals from the present radar consists of four parameters namely range, power, doppler bin number and velocity of target. The expert system has various stages consisting of CNN, LSTM and FC layers before predicting the target class. This time-sequence radar data after pre-processing is reshaped to be sent to a CNN network for extracting features, following classification by LSTM network. The rest of the paper is organized in the following sections.

Section 2 describes the proposed methodology. Section 3 generation of database followed by experimental results in Sect. 4. The proposed work is concluded in Sect. 5.

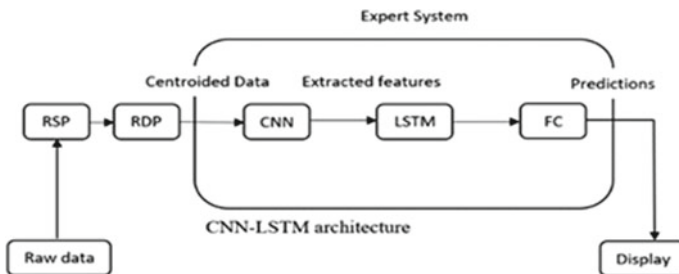


Fig. 1 Block diagram of auto-classification

2 CNN-LSTM Methodology

Convolutional neural networks (CNN) are type of neural network designed for handling image data. They operate directly on raw data instead of domain specific or handcrafted features. The model then learns to automatically extract features called as representation learning and features are extracted regardless of how they occur, known as transform or distortion invariance. This ability of CNN to learn and automatically extract features can be applied to time-series problems.

The proposed model makes use of CNN for feature extraction and LSTM for predictions. Raw radar data is provided to radar signal processor (RSP) which extracts target signal from noise and is given as input to radar data processor (RDP) used for track association, filtering, and computing velocity based on change in position of target. Based on the track ids, reports are clubbed together in a set of 8 (sample size) and provided to the input layer. Each report consists of three parameters, namely, RCS, doppler, and velocity. Convolution layer is followed by max pooling for reducing the number of inferences include the most significant features. The structures are then flattened to one-dimensional vector to be used as single input time step to the LSTM layer. As the next report arrives, the first report is discarded in first-in-first-out (FIFO) fashion and the process is repeated.

Figure 2 shows the architecture used for classification. The CNN-LSTM consists of architecture similar to LSTM architecture in Fig. 3. As quoted earlier the CNN acts as feature extractor from radar signals, and forwards it to LSTM network.

3 Data Modeling

The parameters under consideration for training our classifier are RCS of target, doppler frequency, and velocity.

3.1 Radar Cross Section of Target

Every target is characterized by a radar cross section (RCS) which accounts for the amount of reflected energy. The Institute of Electrical and Electronics Engineers (IEEE) dictionary of electrical and electronics terms [11] defines RCS as a measure of the reflective strength of a target. The mathematical relation of RCS is given by Eq. (1).

$$\sigma = \lim_{R \rightarrow \infty} 4\pi R^2 \frac{|E^{\text{scat}}|^2}{|E^{\text{inc}}|^2} \quad (1)$$

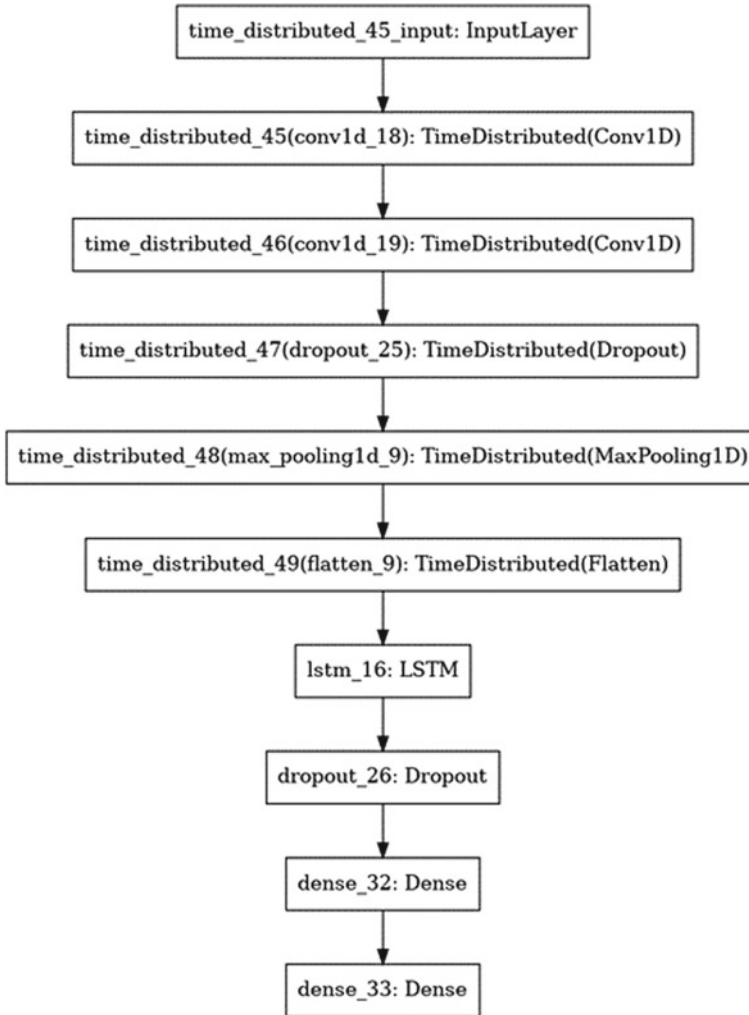
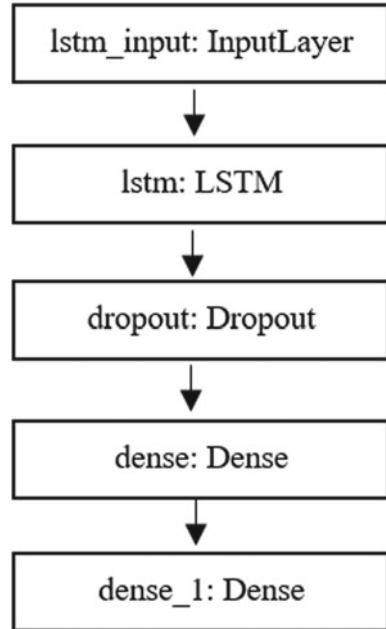


Fig. 2 CNN-LSTM

Probability of detection for moderately complex targets is sensitive RCS which in turn fluctuates with respect to radar-target geometry, target vibration, and radar frequency. For the capturing these complexities, statistical modeling of RCS is used for practical purposes. Swerling models [12] are probabilistic models of RCS proposed for dwell-to-dwell and pulse-to-pulse. For present work, SW-I and SW-III are under consideration. SW-I and SW-III are given by Eq. (2) and Eq. (3) respectively. Figure 4 shows the linear scale RCS distributions of a particular target under consideration. Based on congruence with experimental data for various targets SW-III model was used for RCS distribution.

Fig. 3 LSTM



$$p(\sigma) = \begin{cases} \frac{1}{\sigma} \exp\left[-\frac{\sigma}{\sigma}\right], & \sigma \geq 0 \\ 0, & \sigma < 0 \end{cases} \tag{2}$$

$$p(\sigma) = \begin{cases} \frac{4\sigma}{\sigma^2} \exp\left[-\frac{2\sigma}{\sigma}\right], & \sigma \geq 0 \\ 0, & \sigma < 0 \end{cases} \tag{3}$$

3.2 Velocity Modeling

Velocity is defined as rate of change of displacement with respect to time. The maximum velocity for ground moving targets is restricted by physical limits and mostly by terrains where the radar is deployed. Based upon the limitations, velocities the four targets under consideration are shown in table 1.

From Fig. 5, it can be seen that the velocity samples can attributed to Gaussian distribution. For velocity samples with time, we have considered a Gaussian distribution with mean values varying from 0.5 to 15 m/s.

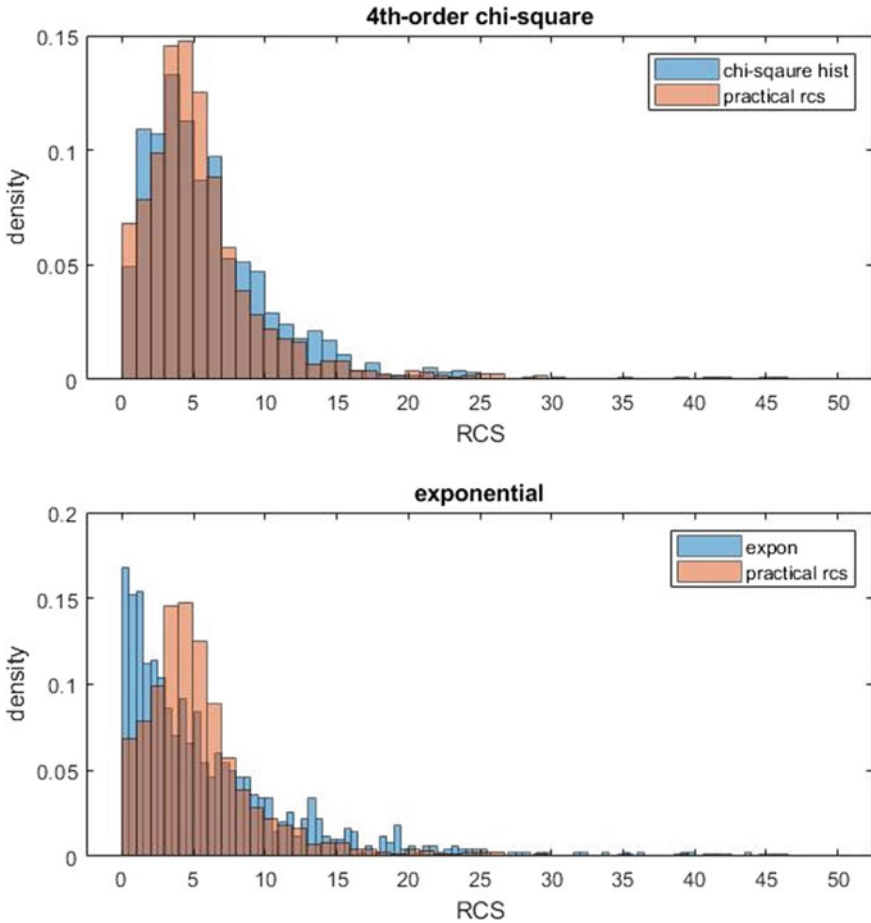


Fig. 4 RCS histogram of practical data correlating with Swerling models

Table 1 Maximum velocity of each target class

Class	Maximum velocity (kmph)
A single person	20
Group of people	20
Vehicle	90
Heavy vehicle	90

3.3 Doppler Frequency Bin

In the present radar, pulse-doppler processing is used to find doppler shift using radial velocity component. The doppler frequency shift (f_d) is given by

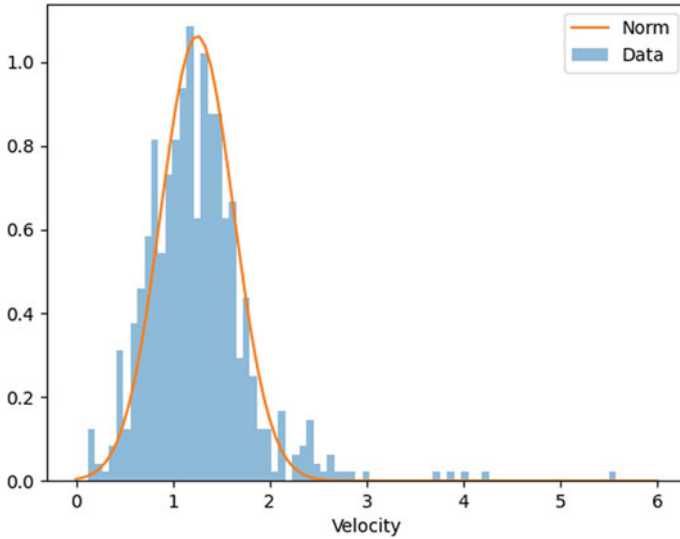


Fig. 5 Histogram of velocity samples of a walking person fitting to a Gaussian pdf of mean 1 m/s

$$f_d = \frac{2v}{c} f = \frac{2v}{\lambda} \quad (4)$$

The speed of electromagnetic wave c , radial velocity of target v , f and λ being operating frequency and wavelength respectively. The maximum frequency shift is bound between $(-\frac{\text{PRF}}{2}$ to $+\frac{\text{PRF}}{2})$. The f_d is assigned a particular doppler bin number. Based on practical data for the four targets doppler data is generated synthetically based on maximum radial speed achievable. For doppler samples with time, we have considered a Gaussian distribution with mean values varying from calculated from Eq. (4) for various velocities in the range.

3.4 Data Set

A single time sample consists of RCS, doppler bin number, and velocity. Input data to CNN is generated by stacking radar echoes one after the other with respect to time to form a block. The number of samples used for making a 2D matrix here are 8, 16, and 32 for three model configurations. Arranging data such a way preserves the time dependency. This form of data is then fed to CNN layers followed by LSTM layers finally producing the output.

4 Experiments

A sequence of observations can be treated as one-dimensional image that the CNN can read and distill the most salient features. Radar data being time dependent, LSTM architecture is used for classification. The extracted features are flattened into one-dimensional time-series data and fed to the LSTM layer. The LSTM output is fed to a hidden dense layer with 30 nodes and ‘Relu’ activation function. The output layer is a fully connected dense layer with 4 nodes and ‘Softmax’ activation function.

Figures 6 and 7 compare the training accuracy and loss of two architectures with three different sample sizes. It was observed that the training accuracy in LSTM architecture reduced significantly with reduction in sample size, whereas the training accuracy and loss varied by a very small amount with CNN-LSTM architecture. The training and test accuracy are provided in Table 2.

The predictions of individual targets will be given in terms of confidence level as shown in Fig. 8, with row number representing track id and column number representing class. The predictions are displayed on the radar display as shown in Fig. 9, with track id, class number and confidence level. A user defined threshold is set (65% in this case) and any predictions below the threshold level will be considered as unidentified.

The predictions are displayed on the radar display as shown in Fig. 9, with track id, class number, and confidence level. A user-defined threshold is set (65% in this case) and any predictions below the threshold level will be considered as unidentified.

Figure 10 shows the classification outputs of two models for a vehicle moving away from the radar. It is observed that for a sample of size 8 time steps, CNN-LSTM is consistently classifying the target, compared to that of LSTM.

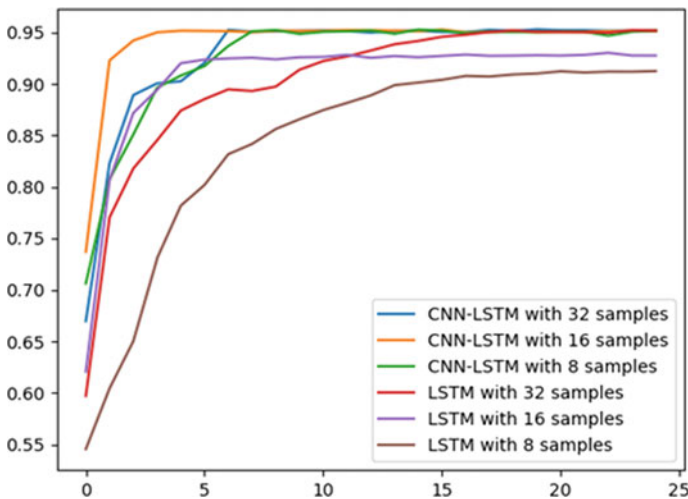


Fig. 6 Accuracy versus epoch graph

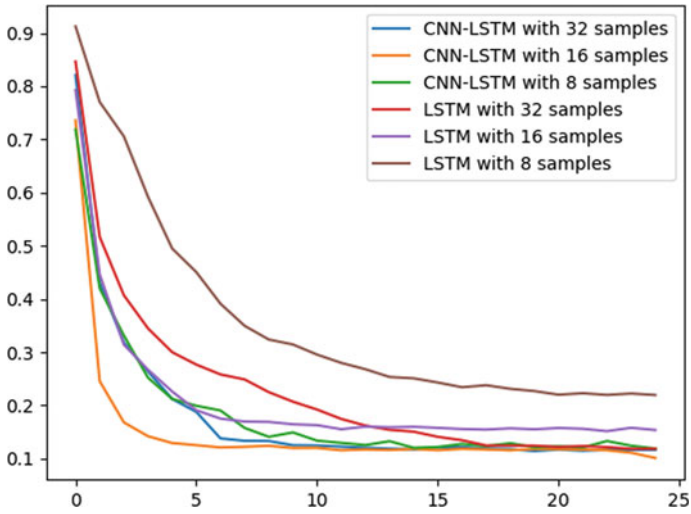


Fig. 7 Loss versus epoch graph

Table 2 Comparison of CNN-LSTM and LSTM architecture with different sample size

Sample size	Architecture	Accuracy %		Training loss
		Training	Test	
8	CNN-LSTM	96.332	93.269	0.119
	LSTM	90.044	86.343	0.224
16	CNN-LSTM	96.428	96.25	0.095
	LSTM	93.573	91.629	0.169
32	CNN-LSTM	96.436	97.501	0.107
	LSTM	96.559	96.744	0.118

	0	1	2	3
0	5.49409e-07	0.44496	0.000210337	0.554829
1	0.0799556	0.230227	0.622849	0.0669686
2	0.000140829	0.909224	0.0668032	0.0238316
3	0.999966	1.77782e-09	5.09351e-10	3.39807e-05
4	0.000133993	0.907062	0.0672412	0.0255626
5	4.90345e-07	0.410772	0.000142586	0.589084
6	0.00763164	0.582926	0.346037	0.0634054
7	0.999967	1.7457e-09	5.12702e-10	3.267e-05

Fig. 8 Predictor output

```
1539 0 with 91.66 % confidence
1540 1 with 99.45 % confidence
1541 3 with 93.74 % confidence
1542 2 with 80.76 % confidence
1543 0 with 77.59 % confidence
1544 2 with 85.14 % confidence
1545 3 with 93.67 % confidence
1546 1 with 99.46 % confidence
1547 0 with 86.33 % confidence
1548 3 with 84.14 % confidence
1549 2 with 80.50 % confidence
1550 1 with 99.54 % confidence
1551 0 with 91.25 % confidence
1552 1 with 99.39 % confidence
1553 unidentified 0.58250654
1554 2 with 78.16 % confidence
1555 unidentified 0.5645936
1556 2 with 83.64 % confidence
1557 3 with 96.25 % confidence
1558 1 with 99.51 % confidence
1559 0 with 74.16 % confidence
1560 3 with 91.56 % confidence
```

Fig. 9 Output on console

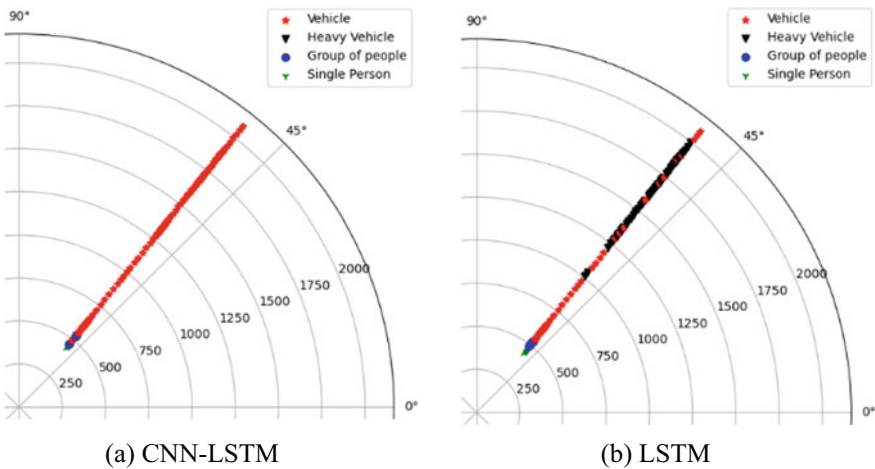


Fig. 10 PPI Screen displaying classifier results for a moving vehicle from 400 to 1900 m at an azimuth of 50 degree approximately a CNN-LSTM and b LSTM

5 Conclusion

The results show that the proposed model can be effectively employed in any radar for surveillance. For sample size of 8 timestamps, the combination of CNN-LSTM gave better accuracy compared to LSTM. Only 5.75% of the overall targets were misclassified within subclassification. In this study we believe that we have developed

an expert system for classification of ground targets in radar combining CNN for feature extraction and LSTM for classification. The proposed model showed better performance with test accuracy of 93.27 compared to 86.34% of LSTM alone. Further research will be focused on classification with increased number of classes.

Acknowledgements The authors would like to thank General Manager, Military Radars, BEL for their valuable help, encouragement and motivation during the implementation of the work described in this paper.

References

1. Skolnik M (2001) Introduction to radar systems, 3rd edn. McGraw Hill, New York
2. Richards MA, Holm WA, Scheer JA (eds) (2010) Principles of modern radar Vol. I: basic principles. SciTech Publishing 2010
3. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
4. Nir Arbel, How LSTM networks solve the problem of vanishing gradients, Posted on December 21 (2018). <https://medium.datadriveninvestor.com/how-do-lstm-networks-solve-the-problem-of-vanishing-gradients-a6784971a577>
5. Christopher Olah, Understanding LSTM Networks, Posted on August 27 (2015). <http://colah.github.io/posts/2015-08-Understanding-LSTMs>
6. Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. *Int Conf Eng Technol (ICET)* 2017:1–6. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
7. Kouba E, Rogers S, Ruck D, Bauer K (1993) Recurrent neural networks for radar target identification. *Proc SPIE Int Soc Opt Eng.* <https://doi.org/10.1117/12.152541>
8. Jithesh V, Sagayaraj MJ, Srinivasa KG (2017) LSTM recurrent neural networks for high resolution range profile-based radar target classification. In: 2017 3rd International conference on computational intelligence & communication technology (CICT), pp 1–6. <https://doi.org/10.1109/CICT.2017.7977298>
9. Sehgal B, Shekhawat HS, Jana SK (2019) Automatic target recognition using recurrent neural networks. *Int Conf Range Technol (ICORT)* 2019:1–5. <https://doi.org/10.1109/ICORT46471.2019.9069656>
10. Wan J, Chen B, Xu B et al (2019) Convolutional neural networks for radar HRRP target recognition and rejection. *EURASIP J Adv Signal Process* 2019:5
11. Jay F (ed) (1984) IEEE standard dictionary of electrical and electronic terms, ANSI/IEEE Std 100–1984, 3d edn. IEEE Press, New York
12. Swerling P (1960) Probability of detection for fluctuating targets. *IRE Trans Inf Theory* 1(6):269–308

Iron Oxide Nanoparticle Image Analysis Using Machine Learning Algorithms



Parashuram Bannigidad , Namita Potraj ,
Prabhuodeyara Gurubasavaraj , and Lakkappa Anigol 

1 Introduction

Nanoparticles popularly abbreviated as NPs have shown their prominent applications in various fields. These particles are synthesized from various metals and metal oxides. The application of the nanoparticles varies depending on the architecture and method of synthesis. Iron Oxide magnetic NPs with appropriate surface chemistry are prepared using various methods such as; physical, chemical, and biological is proposed by Attarad et.al. [1]. The prominent features of Iron Oxide NPs like; low-slung toxicity, superparamagnetic properties, and simple separation methodology have dominated biomedical applications for protein immobilization, thermal therapy, MRI, and drug delivery. The nanoparticles obtained from these methods show different morphological characteristics such as; shape, size, regularity, circularity, porosity, and density. The challenge lies in obtaining uniform size and shapes of the nanoparticles such as; nanorod, self-oriented flowers, spheres, nanohusk, and nanocubes. Synthesizing customized nanoparticles to suit the required application and characterizing them will play a vital role in the world of nanotechnology. For instance, to deliver the precise amount of drug to the targeted area, appropriately scaled nanoparticles should be synthesized and have to be analyzed whether the synthesized nanoparticle is acceptable or not. The characterization using the good old traditional methods is tedious and time-consuming. General-purpose software like ImageJ also requires human intervention and thus has motivated the proposed study to develop an automated tool using digital image processing techniques.

P. Bannigidad · N. Potraj (✉)
Dept. of Computer Science, Rani Channamma University, Belgaum, Karnataka 591156, India
e-mail: namitapotraj@gmail.com

P. Gurubasavaraj · L. Anigol
Dept. of Chemistry, Rani Channamma University, Belgaum, Karnataka 591156, India

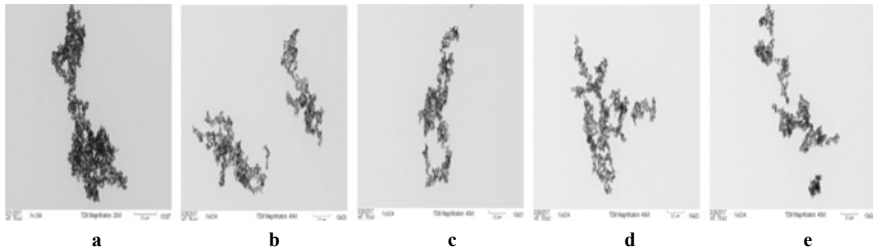


Fig. 1 TEM images of Iron Oxide Nanoparticles synthesized at varying temperatures using chemical (a at 300 °C, b at 300 °C, c at 500 °C, d at 500 °C) and biological (e at 300 °C) methods

The TEM images a, b, c, d are obtained from the chemical method and image e is obtained from the biological method of the Iron Oxide Nanoparticles used in the study are presented in Fig. 1.

The essence of synthesis and characterization of Iron Oxide Nanoparticles is contributed by many researchers in their respective studies. Cuenya [2] proposed that the physical synthesis methods have less control over the size of NPs. Ling Li et. al. [3] proposed the physical and chemical synthesis process of nanoscale Iron-based materials and their environmental applications. Toyokazu Yokoyama [4] explained the size-based effects on the thermal, electromagnetic, optical, and morphological properties of nanoparticles. Aryn et.al. [5] briefed how continuous supercritical hydrothermal synthesis provides the control and scalability of magnetic Iron Oxide Nanoparticles. Lazar et.al. [6] explained particle shape analysis of silica-coated Iron Oxide Nanoparticle clusters using computational methods. Nene et.al. [7] explained different applications of Iron Oxide Nanoparticles such as curing cancer, drug delivery, antifungal activity, antibiotic activity, imaging, and cellular labeling. The Iron Oxide Nanoparticles are used in drug delivery Control in 3-D MRI was proposed by Mohammed et. al. [8]. Bannigidad et.al. [9] have studied the characterization of nano-membrane engineering of Anodic aluminum oxide template. Ethan [10] has described the functionalizing of Iron Oxide Nanoparticles for controlling the movement of immune cells. Bannigidad et al. [11] investigated the impact of time on Al_2O_3 nanopore structures using automated devices on FESEM images. Characterization of various nanoparticles is a vibrant process that can be useful in understanding the computational results of nanoparticles that are carried out by using various image processing techniques. Ismail et al. [12] proposed diverse image segmentation techniques; thresholding, Hidden-MarkovRandom-Field Model-Expectation Maximization (HMRF-EM), Gaussian Mixture Model—Expectancy Maximization (GMM-EM) to segment Iron Oxide Nanoparticles. Vidyasagar et.al. [13] proposed the influence of anodizing time on the porosity of nanopore structures. Reem et.al [14] used some preprocessing techniques namely; noise filtering, and background subtraction, and applied *K*-means clustering algorithms to segment Iron Oxide Nanoparticles from 3-dimensional MRI Images. Bannigida et al. [15] proposed several segmentations techniques; *K*-means, active contour, global thresholding, region growing, and watershed to characterize nanoporous membrane. In this study, an effort is made

Table 1 The details of chemicals used in obtaining the TEM nanoparticles and the temperature maintained during the synthesis process

Image	Chemicals used in the synthesis of the nanoparticles	Temperature (°C)
a	Prepared by Iron Nitrate	300
b	Prepared by FeCl ₂ , 2H ₂ O	300
c	Prepared by Iron Nitrate	500
d	Prepared by FeCl ₂ , 2H ₂ O	500
e	Prepared by Iron Nitrate using plant extract	300

to characterize the Iron Oxide Nanoparticle synthesized from both chemical and biological methods.

2 Materials and Methods

The dataset in the study consists of various TEM images of Iron Oxide Nanoparticles, and are synthesized and prepared by using two different methods, i.e. chemical and biological. The details of chemicals used in obtaining the Iron Oxide Nanoparticles TEM images and the temperature maintained during the synthesis process are given in Table 1.

3 Proposed Method

The objective of the study is to develop an algorithm to analyze the Fe₂O₃ nanoparticles in the TEM images synthesized at varying temperatures using chemical (a at 300 °C, b at 300 °C, c at 500 °C, d at 500 °C) and biological (e at 300 °C) methods. The features of the nanoparticles in the TEM images used in this study include; finding the number of nanoparticles, calculating the size (area), perimeter, major axis, minor axis, porosity, circularity, and interparticle distance. These features are defined as:

- Area: Total number of pixels in an extracted nanoparticle.
- Major Axis: Total number of pixels along the length of the largest axis in a nanoparticle.
- Minor Axis: Total number of pixels along the length of the smallest axis in a nanoparticle.
- Porosity: Total particle size/Total Area*100
- Circularity: Minor axis/Major axis
- Interparticle distance; The distance d , between two particles whose coordinates are (x_1, y_1) and (x_2, y_2) is

$$d = \sqrt{[(x_2-x_1)^2 + (y_2-y_1)^2]}$$

where two points whose coordinates are (x_1, y_1) and (x_2, y_2)

The algorithm described in the proposed work is given below:

- Step 1. Input Iron Oxide Nanoparticle TEM image.
- Step 2. Apply image preprocessing techniques; gamma correction to enhance the quality of the image.
- Step 3. Apply Gaussian Mixture Model-Expectancy Maximization (GMM-EM) segmentation technique to binarize the given input image.
- Step 4. Remove unwanted background and noise from the segmented image
- Step 5. Extract the individual particles and label them on the segmented image.
- Step 6. Compute the geometric features, i.e., area(size), porosity, circularity, interparticle distance, and average area, and store them in the database.
- Step 7 Compare and interpret with manual results.
 - Categorize the nanoparticles based on the following condition.
 - if the area is between is 0–50 nm then extract, count, and display the nanoparticles
 - else if the area is between is 51–100 nm then extract, count, and display the nanoparticles
 - else if the area is between is 101–150 nm then extract, count, and display the nanoparticles
 - else if the area is between is 151–200 nm then extract, count, and display the nanoparticles
 - end
 - Store all the nanoparticles in the database.

4 Experimental Results and Discussion

For the purpose of experimentation, the total of 137 TEM images of Iron Oxide Nanoparticles obtained from two different synthesis methods, i.e. chemical and biological are considered. Images a, b, c, and d are obtained from chemical, and image E is obtained from the biological method. The features of each image are extracted using MATLAB R2018a software on Intel(R) Core™ i5-10210U CPU@1.60 GHz system. In computing the features of the TEM images of Iron Oxide Nanoparticles the significant challenge is the overlapping of particles in each image. The experimentation is carried out by applying basic preprocessing operations such as resizing and noise removing from the original images (Fig. 2. (a)). To extract individual particles from every image, the image segmentation technique named Gaussian Mixture Model—Expectancy Maximization (GMM-EM) is used to segment the nanoparticles from each TEM image (Fig. 2. (b)). The various segmentation techniques, namely; Gaussian Mixture Model—Expectancy Maximization (GMM-EM), *K*-means, and Fuzzy *C*-Means clustering methods are tried, out of these segmentation methods the Gaussian Mixture Model—Expectancy Maximization (GMM-EM)

segmentation technique yields better results and flawless segmentation of nanoparticles. These nanoparticles are labeled (Fig. 2. (c)) and stored as a knowledge base. Further, the geometric features namely; area (size-wise in the range of 0–50, 51–100, 101–150, and 151–200 nm.), average area(%), porosity (%), average circularity (%), and average interparticle distance (nm) are computed from all TEM images of Iron Oxide Nanoparticles A, B, C, D, and E which have been synthesized using chemical and biological methods at two different temperatures (300 °C and 500°C) as depicted in Table 2.

Table 3 shows the overall percentage (area-wise) of nanoparticles that are in the range of 0–50 nm, extracted from both synthesis methods; chemical and biological at different temperatures. The smallest size of Iron Oxide Nanoparticles is suitable for most of the applications; hence the chemists consider the nanoparticles which are in the range of 0–50 nm. So considering this strategy, we have calculated the percentage of the nanoparticles based on the extracted feature value; area among these images. The area-wise percentage of images a, b, c, and d is 43.39%, 62.06%, 57.14%,

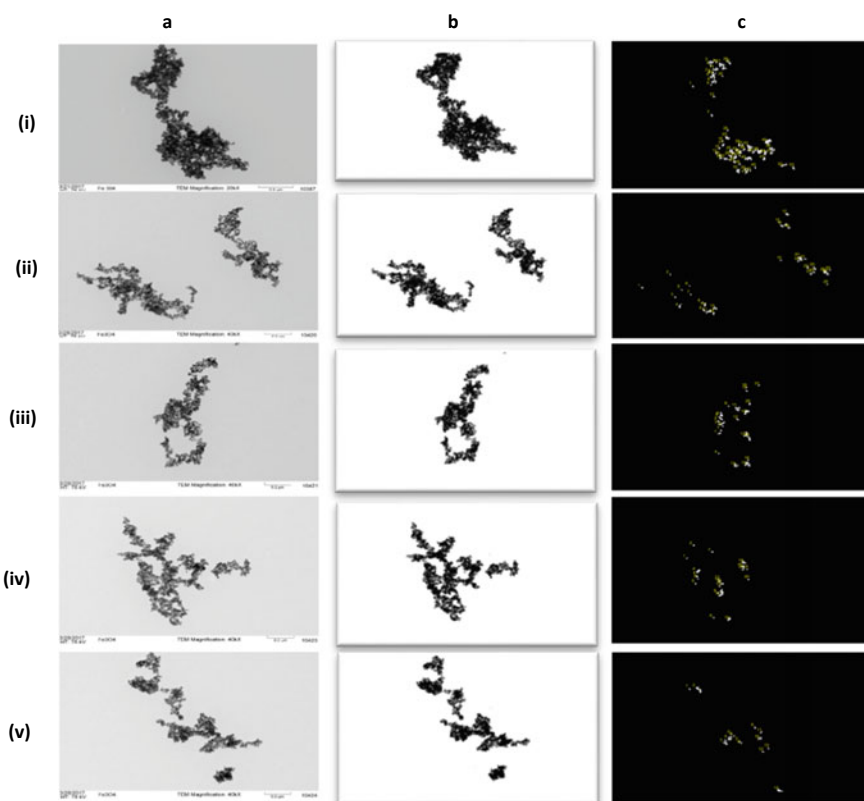


Fig. 2 a Original TEM images of iron oxide nanoparticles, b Segmented images using GMM-EM technique c Labeled nanoparticles

Table 2 Details of computed geometric features; area(size-wise in the range of 0–50 nm, 51–100 nm, 101–150 nm, and 151–200 nm), porosity (%), average circularity (%), and average interparticle distance (nm) for TEM images of Iron Oxide Nanoparticles A, B, C, D, and E

Synthesis Methods	Image With different temperatures (°C)	Number of particles				Avg. area (%)	Porosity (%)	Avg circularity (%)	Average interparticle distance (nm)
		Area in range 0–50 nm	Area in range 51–100 nm	Area in range 101–150 nm	Area in range 151–200 nm				
Chemical method	a (300)	23	13	11	6	81.21	16.9784	57.26	66.1755
	b (300)	18	7	4	0	53.12	5.0766	51.36	54.6966
	c (500)	12	3	5	1	68.35	8.2366	56.08	54.0291
	d (500)	14	4	1	1	48.95	3.6730	62.70	52.1772
Biological method	e (300)	9	1	3	1	60.12	4.7216	54.39	59.2356

Table 3 The overall percentage of nanoparticles that are in the range of 0–50 nm, extracted from both synthesis methods; chemical and biological

Synthesis methods	TEM images	Temperature (°C)	Area (size in range 0–50 nm) in percentage
Chemical method	a	300	43.39
	b	300	62.06
	c	500	57.14
	d	500	70.00
Biological method	e	300	64.28

and 70.00%, respectively, which are synthesized using the chemical method and the area-wise percentage of image e is 64.28%, which is synthesized using the biological method. The images which are synthesized at 500°C, i.e. image c and image d; image d has the highest area-wise percentage of 70.00%. And images which are synthesized at 300°C temperature, i.e., image a, b, and e, image e has the highest area-wise percentage of 64.28%. Since the biological synthesization method is universally accepted as it uses no toxic chemicals, is cost-effective, and is environmentally friendly, it is proposed that the biological method is s for synthesizing the Iron Oxide Nanoparticles TEM images.

The computed features of image a are interpreted and analyzed in comparison with manual results obtained from chemical experts, based on these values the other results of images b, c, d, and e are interpreted.

5 Conclusion

In this study, the TEM images (a, b, c, d, and e) of Iron Oxide Nanoparticles are synthesized using chemical and biological methods at 300 °C and 500 °C temperature. The motive behind the present work is that the traditional characterization techniques are time-consuming and are not economically cost-effective. Hence, an effort is made to automate a tool that can determine the size of TEM images of Iron Oxide Nanoparticles. The Gaussian Mixture Model—Expectancy Maximization (GMM-EM) segmentation technique is applied and diverse features, namely; size (area), perimeter, major axis, minor axis, porosity, circularity, interparticle distance, and average area are computed from TEM images of Iron Oxide Nanoparticles. It is observed that the area-wise percentage of images a, b, c, and d is 43.39%, 62.06%, 57.14%, and 70.00%, respectively, which are synthesized by using the chemical method and the area-wise percentage of image e is 64.28%, by using the biological method. The biological synthesization method is universally accepted as it uses no toxic chemicals, is cost-effective, and is environmentally friendly. The proposed results are analyzed and compared with manual results obtained from the chemical experts and are found to be a good performance.

Acknowledgements The authors are grateful to the ‘KSTEPS, DST, GOVT. OF KARNATAKA’ for providing financial assistance and sanctioning Ph.D. fellowship to carry out this research work. Authors are also grateful to the reviewers for their valuable suggestions that helped in improving this manuscript.

References

1. Ali A, Zafar H, Zia M, ul Haq I, Phull AR, Ali JS, Hussain A (2016) Synthesis, characterization, applications, and challenges of iron oxide nanoparticles. *Nanotechnol Sci Appl* 9:49–67
2. Cuenya BR (2010) Synthesis and catalytic properties of metal nanoparticles: size, shape, support, composition, and oxidation state effects. *Thin Solid Films* 518(12):3127–3150
3. Li L, Fan M, Brown RC, Van Leeuwen J, Wang J, Wang W, Song Y, Zhang P (2006) Synthesis, properties, and environmental, applications of nanoscale iron-based materials: a review. *Crit Rev Environ Sci Technol* 36:405–431
4. Hosokawa M, Nogi K, Naito M, Yokoyama T (2008) Basic properties and measuring methods of nanoparticles. *Nanoparticle technology handbook*, pp 3–48
5. Teja AS, Koh P-Y (2009) Synthesis Growth, properties, and applications of magnetic iron oxide nanoparticles. *Prog Cryst Charact Mater* 55:2245
6. Kopanja L, Kralj S, Zunic D, Loncar B, Tadic M (2016) Core–shell superparamagnetic iron oxide nanoparticle (SPION) clusters: TEM micrograph analysis. *Part Des Shape Anal* 42(9):10976–10984
7. Ajinkya N, Yu X, Kaithal P, Luo H, Somani P, Ramakrishna S (2020) Magnetic iron oxide nanoparticle (IONP) synthesis to applications: present and future. *Materials* 13:4644
8. Almijalli M, Saad A, Alhussaini K, Aleid A, Alwasel A (2021) Towards drug delivery control using iron oxide nanoparticles in three-dimensional magnetic resonance imaging. *Nanomaterials* 11:1876–1888
9. Bannigidad P, Udoshi J, Vidyasagar CC (2020) Automated characterization of aluminum oxide nanopore fesem images using machine learning algorithms. *Int J Adv Sci Technol* 29(03):6932–6942
10. White EE, Pai A, Weng Y, Suresh AK, Van Haute D, Pailevanian T, Alizadeh D, Hajimiri A, Badie B, Berlin JM (2015) Functionalized iron oxide nanoparticles for controlling the movement of immune cells. *Nanoscale* 7(17):7780–7789
11. Bannigidad P, Udoshi J, Vidyasagar CC (2018) Effect of time on Aluminium FESEM nanopore images using fuzzy inference system. *Recent Trends Image Process Pattern Recogn* 1037:397–405
12. Ismail HJ, Barzinjy AA, Hamad SM (2019) Analysis of nanopore structure images using MATLAB software. *Eurasian J Sci Eng* 4(3):84–93
13. Vidyasagar CC, Bannigidad P, Muralidhara HB (2016) Influence of anodizing time on porosity of nanopore structures grown on flexible TLC aluminium films and analysis of images using MATLAB software. *VBRI, Adv Mater Lett* 1:71–77
14. Alanazi RS, Saad AS (2020) Extraction of iron oxide nanoparticles from 3 dimensional MRI images using K-mean algorithm. *J Nanoelectron Optoelectron* 15:1–7
15. Bannigidad P, Udoshi J, Vidyasagar CC (2019) Characterization of Aluminium oxide nanoporous images using different segmentation techniques. *Int J Innov Technol Exploring Eng (IJITEE)* 8(12):2491–2497

Bankruptcy Prediction Using Bi-Level Classification Technique



Abhinav Antani , B. Annappa , Shubham Dodia ,
and M. V. Manoj Kumar 

1 Introduction

Bankruptcy is a financial state for any firm or person where they are unable to pay their debt. Financial investors, banks, government, and money lenders seek an efficient method to determine the bankruptcy status of the firm. Prediction of bankruptcy will help all the stakeholders of the company. Because of this reason, intensive research regarding the prediction of bankruptcy has been going on. For example, Altman [1] uses multivariate discriminant analysis to obtain a z -score which is used to classify bankrupt and non-bankrupt companies. Carton [2] gave measures to determine the performance of the organization. Parameters are divided into different categories such as profitability measures, growth-based measures, and market-based measures. Bankruptcy prediction can be treated as a classification problem. Machine Learning (ML) can be used to classify bankrupt and non-bankrupt companies. Altman, Kimura, and Barboza [3] presented ML models to predict bankruptcy based on Carton's [2] and Altman z -score [1] parameters. The main aim of this work is to improve the prediction performance of bankruptcy using ML algorithms. The features that have been used to train the ML algorithms are the financial ratios that are available for public access. The key observation that has been made is that adding the indicators of the organization's performance has resulted in an improved performance for

A. Antani · B. Annappa · S. Dodia (✉)

Department of Computer Science and Engineering, National Institute of Technology Karnataka,
Bangalore, Surathkal, India
e-mail: shubham.dodia8@gmail.com

B. Annappa

e-mail: annappa@ieee.org

M. V. Manoj Kumar

Department of Information Science Engineering, Nitte Meenakshi Institute of Technology,
Bangalore, Karnataka, India

bankruptcy prediction [3]. The indicator that is used as a feature in this work is Tobin's Q . The key contributions of this work are listed as follows:

1. A new feature set including both organizational indicators and market-based measures are used to predict bankruptcy.
2. A heterogeneous bi-level classification technique is introduced to perform bankruptcy prediction.
3. An improvement in the results have been observed for the proposed work.

The remainder of the paper is organized as follows. Section 2 delves into previous research. Section 3 explains the methodology. Section 4 digs into the details of results and analysis. Section 5 brings the project to a conclusion.

2 Previous Work

Financial investors, banks, governments, and money lenders seek to know the status of companies and they want to know their bankruptcy status. Because of this reason extensive research has been going on predicting bankruptcy.

2.1 Altman Z-score

In year 1968, Edward Altman [1] gave us financial ratios and discriminant analysis method to predict the bankruptcy. The goal of Altman [1] is to evaluate the analytical quality of ratio analysis. This research aims to assess the analytical quality of ratio analysis. Final discriminant function uses 5 financial ratios shown in Eq. 1.

$$Z = 0.012 X_1 + 0.014 X_2 + 0.033 X_3 + 0.006 X_4 + 0.999 X_5 \quad (1)$$

where, X_1 = liquidity, X_2 = profitability, X_3 = productivity, X_4 = leverage ratio, X_5 = asset turnover and Z = altman z-score. If z-score is less than 1.8 then firm is going to be bankrupt in near future. If z-score is between 1.8 and 2.9 then Altman defined this as gray area. Firms in gray area are likely to go bankrupt if they are not monitored properly. If z-score is greater than 2.9 then firm is in no danger.

2.2 Inclusion of Tobin's Q as Feature

Altman [1] used 5 financial ratios to obtain z-score which is used to predict bankruptcy. These ratios are liquidity, profitability, productivity, leverage ratio, and asset turnover. ML model [3] have used 11 financial ratios as features to predict bankruptcy. Out of these 11 financial ratios, 5 are Altman z-score parameters. And

other six parameters are from Carton's study [2] such as return on equity, operational margin, market to book ratio, growth in sales, growth in employees, and growth in assets. In this work, use of Tobin's Q as a market-based measure [2] along with Altman [1] and Carton's [2] parameters is proposed.

Tobin's Q : The Q ratio or Tobin's Q is the market value of the company divided by the replacement cost of its assets [7]. It represents the relationship between the market value of the company and intrinsic values. It can be used to determine whether a firm is overvalued or undervalued. The formula of Tobin's Q is stated below:

$$\text{Tobin's } Q = \frac{\text{Market value of the Firm}}{\text{Replacement Cost of Firm's Assets}} \quad (2)$$

If the value of Tobin's Q is between 0 and 1, the asset's replacement cost is greater than the firm's market value. It indicates that the company is under-valued. If the value of Tobin's Q is greater than one, the firm's market value is greater than its replacement cost. It means that firm is overvalued. To summarize, the previous works on bankruptcy prediction used only Altman parameters as features. However, the inclusion of organizational indicators such as Tobin's Q has displayed improvement in the performance. Therefore, in this work, we aim to develop a heterogeneous classification system to predict bankruptcy.

3 Proposed Methodology

3.1 Data Set

Any ML model is driven by its data. The data set used for this work is collected from Kaggle¹. This data set has around 92 k records with 558 records of bankrupt firms. These firms are US-based firms whose information is publicly available.

Data set ranges from the year 1971 to 2017. It has 13 columns, one is for class label and the rest are features for the model. Features and their formula are mentioned in the Table 1. After missing value imputation and data balancing 80% of the data is used to train the model while the rest 20% is used to test the model.

Where, TA = Total Asset, EBIT = Earning before interest and tax, TD = Total Debt, MVS = Market value of share, and NI = Net Income. Features and formula are from the study [1] and [2].

¹ <https://www.kaggle.com/shuvamjoy34/us-bankruptcy-prediction-data-set-19712017>

Table 1 Features used in the model

Name	Formula
Liquidity	Net Working Capital \div TA
Profitability	Retained Earnings \div TA
Productivity	EBIT \div TA
Leverage	MVS * number of shares \div TD
Asset Turnover	Sales \div TA
Operational Margin	EBIT \div Sales
Return on Equity	NI \div common stock holder equity
Growth in Sales	$(Sales2 - Sales1) \div Sales1$
Growth in Employee	$(Emp2 - Emp1) \div Emp1$
Growth in Asset	$(Asset2 - Asset1) \div Asset1$
Market to book ratio	Market Value \div Book Value
Tobin Q	Market Value \div Replacement cost of asset

3.2 Missing Value Imputation

Data needs to be cleaned before giving it as an input to ML models. Missing value imputation is one of the stages of data cleaning. This data set contains missing values. Missing data values are usually encoded as NaN, blank, or place- holders. In this data set, missing values are encoded with blank. In this work, the performance of the 3 missing value imputation techniques are compared. Missing value imputation by mean, median, and K -nearest neighbor imputation ($K = 15$). A comparison of their result is shown in the results section.

3.3 Data Balancing

Balancing the data set is an important step while making a prediction using ML [8]. The problem which is addressed in this study occurs rarely. In the real world, more companies are non-bankrupt than bankrupt. This data set has around 92 k records and out of these 92 k records, only 558 records are of bankrupt companies. Other records are of non-bankrupt companies. In this data set 99% of the records are of non-bankrupt companies and only 1% records are of bankrupt companies. In this study, Near-Miss downsampling technique is used to balance the data set. It selects examples from the majority class based on the Euclidean distance to minority class examples. It stops when majority class samples are equal to minority class samples. After using the Near-Miss Algorithm, both the classes (bankrupt and non-bankrupt) have 558 records each.

3.4 The Model: Bi-Level Classification

Figure 1 illustrates the architecture of the bi-level classification used in this work. Here, meta-model uses the prediction done by the base models as features along with the training data to give the final prediction. Training data samples used as input in meta-model are different than training data samples which are used as inputs in base models. First of all training data is given as input to all the base models. The base model uses the input data as features and gives its prediction. These predictions are stored as level-one predictions. These level-one predictions are given as input to the meta-model. The final prediction is given by using these level-one predictions and training data meta-model.

In this bi-level classification technique, two single-level classifiers; decision tree, and linear support vector machine, four ensemble techniques; AdaBoost, gradient boost, bagging with random forest estimators, and bagging with AdaBoost estimators are used as base heterogeneous classification models. The decision tree is used with a Gini index and it has a maximum depth of 5. SVM is used with a linear kernel. Adaboost and gradient boost are used with 50 estimators. Gradient boost uses logistic regression as the loss function. Bagging with random forest and bagging with AdaBoost uses 10 estimators. Predictions of these heterogeneous classification models are stored as level 1 predictions. And they are given as input to the meta-algorithm.

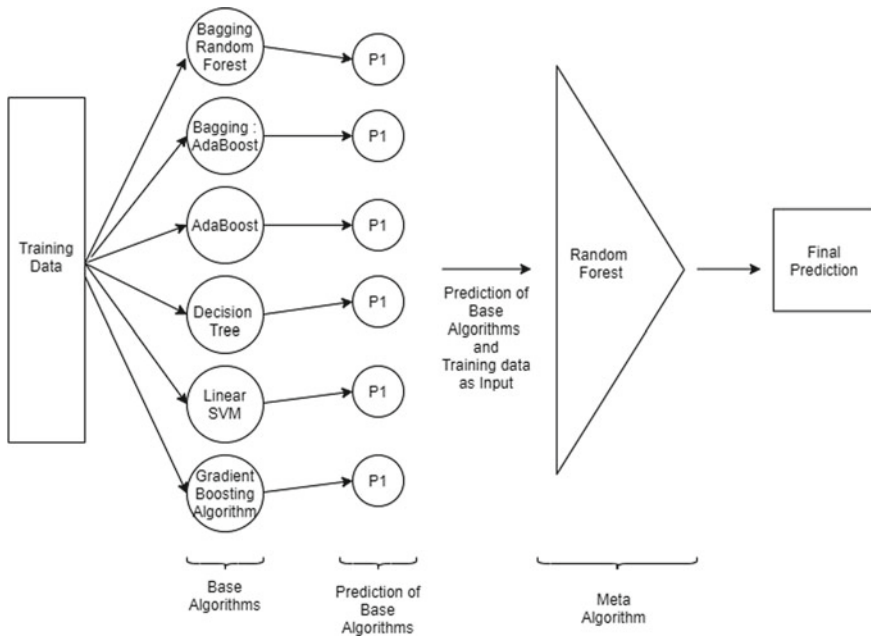


Fig. 1 Diagram of Bi-level classification

In this bi-level classification model to predict bankruptcy, Random Forest is used as a meta-algorithm. It uses training data samples plus predicted class labels from the base models as a feature. Prediction made using this meta-model is the final prediction of the bi-level classification.

4 Analysis and Results

In this section, results and comparison among different classification models is shown. Accuracy is not necessarily an appropriate measure to evaluate ML model used for classification. To have fair evaluation of the model, true positive, true negative, false positive, and false negative can be calculated. Based on their values recall and precision can be calculated. From recall and precision $F1$ -Score can be calculated. In this study $F1$ -score is used to judge the performance of model.

4.1 Comparison of Model

As data set which is used by the Altman is not available, to compare the performance of current model with studies of Altman, kimura and barboza [3] four independent models with different set of features have been made. Importance of including Tobin's Q as parameter can be shown using these results. Below are the four feature set used.

Type 1: Altman z -score [1] (5 features) liquidity, profitability, productivity, leverage, asset turn over.

Type 2: Altman z -score [1], and Carton [2] (11 features): liquidity, profitability, productivity, leverage, asset turn over, return on equity, operational margin, market to book ratio, growth in sales, growth in assets, growth in employee.

Type 3: Altman z -score [1], Carton [2], and Tobin's Q (12 features): liquidity, profitability, productivity, leverage, asset turn over, return on equity, operational margin, market to book ratio, growth in sales, growth in assets, growth in employee, tobin's Q .

Type 4: Altman z -score [1], Carton [2] (Except Market to book ratio), and Tobin's Q (11 features): liquidity, profitability, productivity, leverage, asset turn over, return on equity, operational margin, growth in sales, growth in assets, growth in employee, tobin's Q .

Altman [1] uses the Type 1 feature set. Barboza, Kimura, and Altman [3] use the Type 2 feature set. In this study best results are achieved with Type 4 feature set. To compare the performance of the developed model with that of prior studies, the Random Forest algorithm is evaluated with the available data set and Type 2 features. The Bi-level classification algorithm is tested with the same data set and Type 4 features.

4.2 Results

Figure 2 displays comparison of missing value imputation techniques. Y axis is *F1*-score and X axis is different classification algorithms for different missing value imputation techniques. Figure 2 is graphical representation of Table 2. From Fig. 2 it is clear that for all classification algorithms, missing value imputation by KNN imputation is performing better than other missing value imputation techniques. Value of *K* is kept as 15 in missing value imputation technique.

Figure 3 is performance of various classification models when different feature set are used. It is graphical representation of Table 3. Type 1, Type 2, Type 3, and Type 4 are different feature set described in Sect. 4.1. Y axis has *f1*-score. And x axis has classification models when they use different feature set. From the results in Table 3 and graphical representation of data in bar graph in Fig. 3 it is evident that using indicators of organizational performance along with Altman *z*-score parameters as features

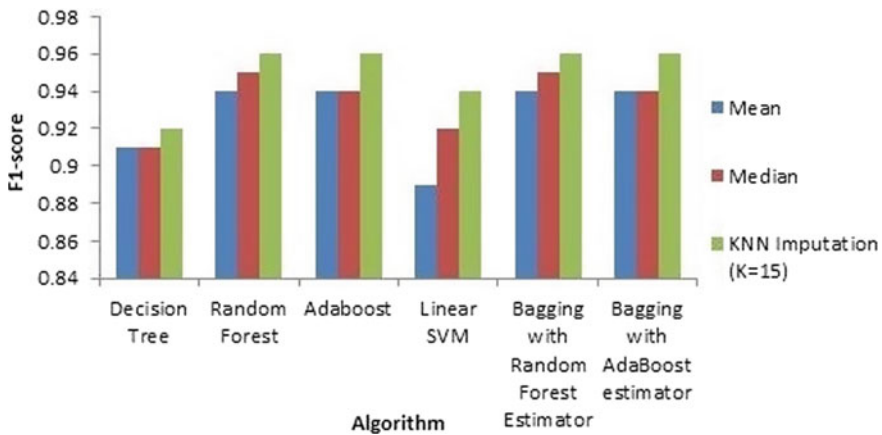


Fig. 2 Comparison on missing value imputation

Table 2 Comparison of missing value imputation

Missing Value Imputation: Comparison using <i>F1</i> -score			
Algorithm	Mean	Median	KNN Imputation
Decision tree	0.91	0.91	0.92
Random forest	0.94	0.95	0.96
AdaBoost	0.94	0.94	0.96
Linear SVM	0.89	0.92	0.94
Bagging with random forest	0.94	0.95	0.96
Bagging with AdaBoost	0.94	0.94	0.96

improves the performance of ML model. Best results by an individual classification model are achieved when Tobin’s Q is replaced as market-based parameter.

Table 4 illustrates comparison of different classification algorithms. For this comparison Type 4 feature set is used. From the able we can note that bi-level classification technique is performing better than other classifiers.

Table 5 illustrates performance of bi-level classification technique when used with different features. Type 1, Type 2, Type 3, Type 4 are feature set from 4.1. From the representation it is clear that bi-level classification technique algorithm is giving best performance when type 4 feature is used. Accuracy and f1-score both are highest when type 4 feature set is used.

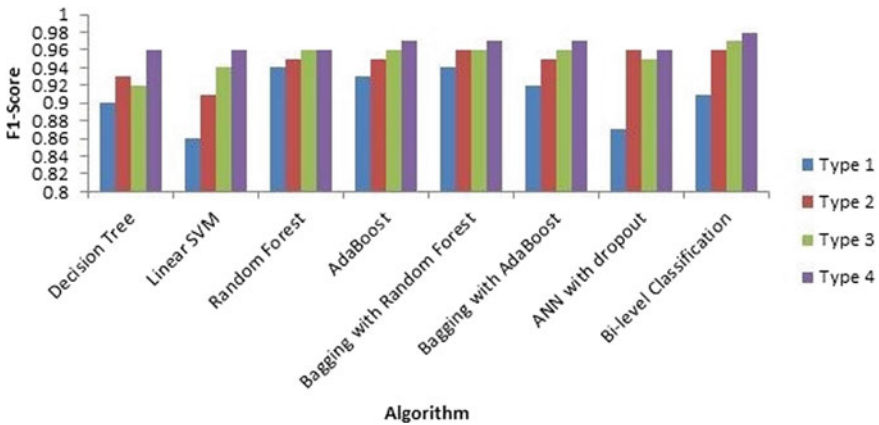


Fig. 3 Comparison of different feature set

Table 3 Performance of classifiers with different feature sets

Performance comparison of Feature set using $F1$ -score

Algorithm	Type 1	Type 2	Type 3	Type 4
Decision tree	0.90	0.93	0.92	0.96
Linear SVM	0.86	0.91	0.94	0.96
Random forest	0.94	0.95	0.96	0.96
AdaBoost	0.93	0.95	0.96	0.97
Bagging with random forest	0.94	0.96	0.96	0.97
Bagging with AdaBoost	0.92	0.95	0.96	0.97
ANN with dropout	0.87	0.96	0.95	0.96
Bi-level classification	0.91	0.96	0.97	0.98

Table 4 Comparison with other classifiers

Algorithm	Accuracy	F1-score scaled to 100
Decision tree	96.85	96
Random forest	96.42	96
AdaBoost	97.7	97
Linear SVM	95.5	96
Bagging with random forest	96.42	97
Bagging with AdaBoost	96.4	97
ANN with dropout	96.45	96
Bi-level classification	97.8	98

Table 5 Performance of Bi-level classification

Feature set	Accuracy	F1-score scaled to 100
Type 1	91.1	91
Type 2	96	96
Type 3	96.4	97
Type 4	97.8	98

5 Conclusion

Bankruptcy is a state in which a person or a company is unable to pay their debts. In this study, along with Altman *z*-score and indicators of organizational performance an indicator of corporate governance is added as a feature. Tobin’s *Q* is included as a feature in this study. Using missing value imputation by KNN imputation and near-miss algorithm to balance the dataset, bi-level classification model achieved accuracy of 97.8% with an *F1*-score of 0.98. Future studies should extend this work by including more indicators of corporate governance as feature.

References

1. Altman EI (1968) Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *J Finance* 23(4):589–609
2. Carton RB (2004) Measuring organizational performance: an exploratory study (Doctoral dissertation, University of Georgia)
3. Barboza F, Kimura H, Altman E (2017) Machine learning models and bankruptcy prediction. *Expert Syst Appl* 83:405–417
4. Fen Y, P’ng Y (2019) Tobin’s *Q* and its determinants: a study on Huawei technologies Co., Ltd
5. D. Tarliman: The Corporate Scandal and the Probability of Bankruptcy: A Case Study of Mylan NV. Available at SSRN 3385217, (2019).

6. Wolfe J, Sawaia ACA (2003) The Tobin Q as a company performance indicator. In: *Developments in business simulation and experiential learning: proceedings of the annual ABSEL conference* 30
7. Fu L, Singhal R, Parkash M (2016) Tobin's Q ratio and firm performance. *Int Res J Appl Finance* 7(4):1–10
8. Veganzones D, S'everin E (2018) An investigation of bankruptcy prediction in imbalanced datasets. *Decision Support Syst* 112:111–124

Infant Brain MRI Segmentation Using Deep Volumetric U-Net with Gamma Transformation



Gunda Sai Yeshwanth , B. Annappa , Shubham Dodia ,
and M. V. Manoj Kumar 

1 Introduction

Studies show that most brain abnormalities form during the first year of brain development. The baby connectome project was started to identifying the factors that contribute to healthy brain development. The University of North Carolina heads this project [1]. The Medical Image Computing and Computer-Assisted Intervention Society, abbreviated as MICCAI, is a society that creates awareness among the world research community about the present problems in the medical field by hosting conferences and organizing competitions with deeply concerning problem statements. (For reference: Iseg termed after infant segmentation [2].) This competition deals with the segmentation of 6-month-old children's brains into three parts: white matter, gray matter, and cerebrospinal fluid. The University of North Carolina backs this competition as part of the baby connectome project. After thinking very profoundly about resolving brain abnormalities, researchers have come to one understanding that most of the brain functions develop in their first year of growth. So researchers are trying to study brain growth during 6–8 months, called the isointense phase.

There are three phases in the brain development of a 1-year-old. 0–6 months is infantile phase and 6–8 months is an isointense phase, and above eight months are

G. S. Yeshwanth · B. Annappa · S. Dodia (✉)

Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, India

e-mail: shubham.dodia8@gmail.com

B. Annappa

e-mail: annappa@ieee.org

M. V. Manoj Kumar

Department of Information Science Engineering, Nitte Meenakshi Institute of Technology, Bangalore, Karnataka, India

called adult phase [3]. In the infantile phase, nothing is much grown, so segmentation is not possible, and in the adult phase, brain segmentation is possible and not very difficult. However, the major problem is to segment the brain into defined three parts in the isointense phase as most of the white matter and gray matter are overlapping, and there are not many magnetic resonance imaging (MRI) scans available for the models to learn. Iseg dataset is the biggest available dataset with ten patients' MRI images for training. And other ten is for testing. There are two MRI scans, T1 and T2 images, for a single person [2].

The image segmentation technique had greatly helped in analyzing images. Even before deep learning algorithms have done image segmentation, there have been many methods used to perform the same task [4–6]. Deep learning models resulted in state-of-the-art performance [2]. Image segmentation means categorizing each pixel value in the image into a set of classes. Image segmentation has numerous applications in all domains like medical, technology of self-driving cars, satellites. Some algorithms came into existence to facilitate this segmentation to be as accurate as possible.

Segmentation of an image using deep learning uses neural networks. U-net is perhaps the most famous architecture in the field of image segmentation. U-net uses autoencoder architecture which has expanding and contracting paths; upsampling the image to increase the spatial resolution of the image and downsampling to decrease the image's spatial resolution. Pooling layers, either max or min, can do downsampling, and either the upsampling layer or transpose layer can do upsampling [7]. The main aim in this work is to develop a segmentation algorithm to perform segmentation for child brain MRI scan. It becomes a challenging task to segment the brain of children belonging to age group of less than 1 year old. Therefore, in this work, a deep learning architecture is introduced to perform brain segmentation on MRI images. The key contributions of this work are listed below:

1. A three-dimensional U-Net architecture is proposed for image segmentation in brain MRI scan.
2. The proposed method is compared with state-of-the-art image segmentation techniques to analyze the performance.
3. The proposed method resulted in a good performance as compared to other state-of-the-art image segmentation methods.

The rest of the paper is organized as follows: Existing brain image segmentation techniques are discussed in Sect. 2. The details of the dataset used to develop the segmentation technique are given in Sect. 3. The detailed description of the proposed method is presented in Sect. 4. The results obtained for the proposed method, and the future possibilities are given in Sect. 5. Finally, the summarization of the work is presented in Sect. 6.

2 Literature Review

Kendall et al. [1] have shown that 3D segmentation of a volumetric image is better than using 2D segmentation. The work also studied ensemble methods working on the Iseg-2017 dataset and proposed a dense network for brain MRI segmentation. Gao et al. [8] proposed a fully conventional neural network to address the segmentation of the infant's brain. Instead of giving explicit images, the work used coarse and dense feature maps to learn some of the regions of an image and used a transformation module to combine all the layers. Wu et al. [4] dealt with MRI images of 1-year-old, two-year, and 3-year-old. They gathered about 95 MRI images and used registration and atlases methods to learn the patterns. The work also added brain contrast probability maps to give the model more information. Weinberger et al. [5] after observing when connections are very near, the results of convolutional neural networks are good, the work developed a new model called dense net in which every layer gets inputs from all layers which are before it. Jiang et al. [6] proposed a segmentation correction algorithm; due to less contrast difference between gray matter and white matter in both T1 and T2 images, there will be misclassifications, so it is more common that models result in an error so the work proposed a method which will be able to correct some of those errors.

Lei et al. [9] proposed a 3D convolution neural network using pyramid dilation while downsampling and a special type of attention network while upsampling. This model resulted in an excellent dice value of 0.90 for White matter and stood top1 in Iseg 2019 challenge. Lienkamp et al. [10] proposed a three-dimensional U-net which replaces all 2D counterparts of U-net paper into 3D counterparts. Moreover, the work was specially designed to learn from the sparsely annotated images. Annotating only some part of the image the model should be able to learn. The work found it very useful in many applications.

Atlas generally means a predefined map that is present for reference. Similarly, even in image segmentation, the atlases method uses some reference atlas of the previous patient. Atlases are generated based on previous patient's data. There can be one atlas or multiple atlases. Parametric models are those models which do not have the freedom to learn anything the model wants. Models work based on some prior restrictions and assumptions, and the model will learn in that manner.

Based on the review of the existing systems, it is observed that the 3D deep learning models perform better in segmenting the infant brain as in comparison with the 2D deep learning models. The results have exhibited in improved performance in the dice score when the MRI images are trained on 3D deep learning image segmentation methods. Therefore, in this paper, a deep learning algorithm is proposed for segmentation of infant brain MRI images as deep learning proved to be very effective in segmentation than atlas-based methods and parametric-based methods due to the freedom it gives the model to learn very complex equations. It is also seen that for volumetric images 3D segmentation works better than 2D convolutions [11]. U-net-based architecture is used in this work as it is specifically meant for medical image segmentation.

3 Dataset

The dataset of this work is not public as this problem statement is part of the MICCAI challenge.¹ However, details of the dataset are as follows. There are ten patients MRI images of $144 * 192 * 256$ pixels. For every patient, both T1 and T2 MRI scans are given. The validation set also contains ten patient's data with the same configuration as training images. T1-weighted MR images were acquired with 144 sagittal slices: TR/TE = 1900/4.38 ms, flip angle = 7° , resolution = $1 \times 1 \times 1 \text{ mm}^3$; T2-weighted MR images were obtained with 64 axial slices: TR/TE = 7380/119 ms, flip angle = 150° , resolution = $1.25 \times 1.25 \times 1.95 \text{ mm}^3$. Along with training data, the segmentation output of each training sample with the below information.

0: background (everything outside the brain), 1: cerebrospinal fluid (CSF), 2: gray matter (GM), and 3: white matter (WM).

Access to the dataset is only possible by participating in the competition. Here the link for information is about the dataset and competition page Dataset information.

4 Proposed Methodology

This paper proposes a 3D U-net-based deep learning architecture for segmentation of infant brain MRI images; Fig. 1 depicts the architecture. The proposed model uses five downsampling layers and five upsampling layers similar to U-net [7]. Kernel size chosen is three as per default U-net architecture [11]. The dimensions of filters used and input dimensions are mentioned in Fig. 1. This U-net uses Leaky ReLU instead of just ReLU as ReLU has dead points; once the value goes below zero, the neurons are dead and cannot be used for learning, so to save the network from stagnating, Leaky ReLU is used [12]. The alpha value is set as default which is 0.3. Adam optimizer is used with a default learning rate of 0.001. The loss function that is chosen is dice loss [13]. Dice is also one of the metrics of evaluation of Iseg. Batch normalization is used after every convolution layer as normalized data always performs better. Dropout layers of value 0.6 are added after every upsampling and downsampling layer to reduce over-fitting. Gamma transformation with a value of 0.9 is applied to all the images.

4.1 Preprocessing the Data

The dataset has images in hdr format. Those images are converted from hdr to nii format.

¹ <https://iseg2019.web.unc.edu/>.

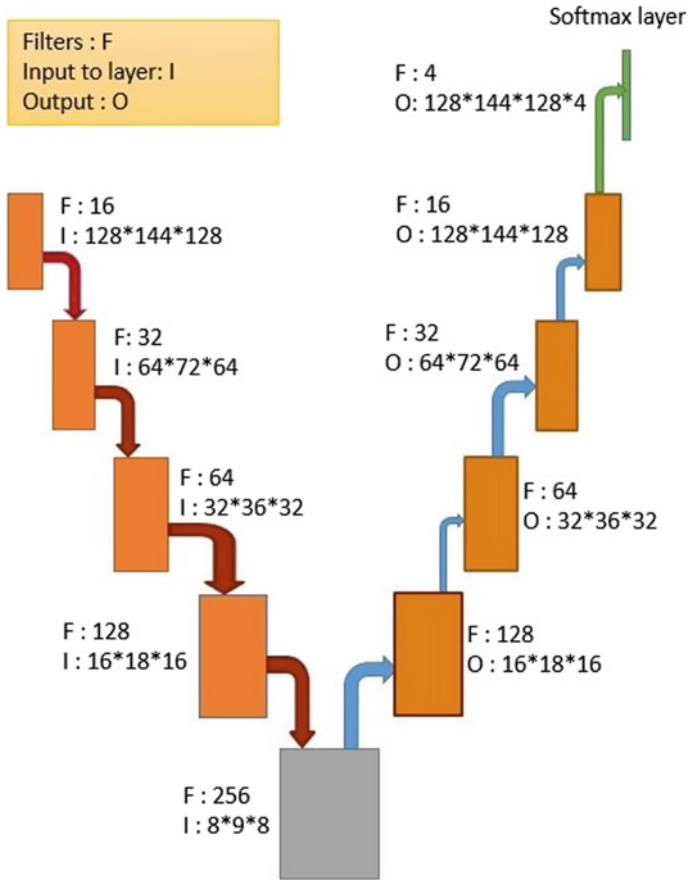


Fig. 1 U-net architecture

4.2 Gamma Transformation

As images of T1 and T2 have significantly less contrast difference between gray matter and white matter, some image contrast transformations are applied for the model to learn better. Gamma transformation has shown that it can improve the performance of the model from previous studies [10]. Gamma in the range of 0.9–1.1 is applied to make sure that the originality of the image is not lost. The original MRI image and the gamma-transformed image are shown in Fig. 2a, b. Model improved dice after doing gamma transformation. So, these images were shown to present the difference between original MRI and gamma transformed MRI.

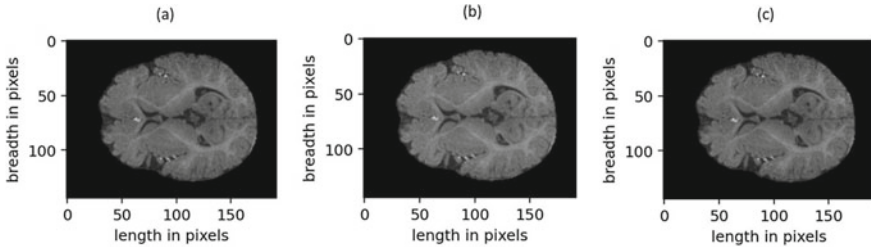


Fig. 2 MRI images **a** original image, **b** gamma transformed image, and **c** piecewise transformed image

4.3 Piecewise Transformation

Piecewise transformation [14] is one of the image contrast change transformation methods. Here, different pixel value ranges are multiplied by different ranges, so that the difference between a certain range of pixels from other ranges is visible. The piecewise transformed image is shown in Fig. 2c.

5 Results and Discussion

This section mainly focuses on the discussion of results for the various experiments carried out. The U-net architecture, along with gamma transformation, resulted in 85.64 in white matter, 88.24 in gray matter, and 93.75 in cerebrospinal fluid. The model is trained for 2000 epochs with Adam optimizer. The performance of the proposed model on the validation set is shown in Table 1. All the values are dice values. The first column represents dice values of CSF segmentation followed by GM and WM segmentation. Table 2 shows the experiments based on U-net architecture. The architecture column represents what modifications are done to the U-net to get the dice value in the second column. The architecture format is as follows: the number of epochs + what images is used for training + batch size. Some architectures used piecewise and gamma transformations. In this column, high represents the increased capacity of U-net which starts from 32 filters, while the previous U-net starts from 16 filters.

The format followed to represent architecture column is same as Table 2 for Tables 3, 4 and 5. In Table 3, experiments are performed with augmentation techniques such as zooming, rotation. In Table 4, experiments of attention U-net are mentioned. Table 5 experiments related to attention U-net with augmentation are mentioned. The patchwise results in Table 6 are based on using $16 * 16 * 16$ patch size. Each of them is trained for 150 epochs. Segmentation image of white matter, gray matter, and cerebrospinal fluid is shown in Fig. 3.

Table 1 Validation results of proposed model in terms of dice

ID	CSF dice	GM dice	WM dice
11	0.943963	0.898107	0.87005
12	0.931463	0.865996	0.831487
13	0.946864	0.892952	0.871021
14	0.930118	0.876744	0.853067
15	0.944716	0.89604	0.867549
16	0.939883	0.884636	0.870405
17	0.937428	0.882884	0.866403
18	0.946895	0.882597	0.865278
19	0.940778	0.882393	0.867755
20	0.932929	0.867366	0.821604
21	0.932132	0.878341	0.85386
22	0.925424	0.882624	0.84831
23	0.935371	0.880758	0.847644
Mean	0.93753569	0.88241831	0.85649485
Std	0.00693551	0.00955553	0.01586224

Table 2 U-net-based experiments without augmentation

Architecture	Dice (in %)
1000 + (T1,T2) + 1	78.84
2500 + (T1,T2) + 2	80.03
2500 + T1 + 2	81.90
4000 + T1 + 2	82.39
1250 + T1 + 1 + piecewise	83.61
900 + T1 + 1 + gamma	83.37
1250 + T1 + 1 + gamma	84.38
2000 + T1 + 1 + piece + highcap	85.10
150 + T1 + 1 + piecewise + patch(64)	76.99
150 + T1 + 1 + gamma + patch(64)	78.57
150 + (T1,T2) + 1 + gamma	80.2
150 + T1 + 1 + overlap-patch(64)	78
2000 + T1 + 1 + gamma	85.32
1000 + T1 + 10 training images	99.28

Table 3 U-net-based experiments with augmentation

Architecture	Dice (in %)
1000 + (T1,T2)	77.60
1500 + T1	78.59
1000 + high	81.38

Table 4 Attention U-net without augmentation

Architecture	Dice (in %)
1500 + (T1,T2)	79.73
500 + weighted cross entropy loss	75.70
2000 + T1 + piece	83.94
2000 + T2 + piece	81.90

Table 5 Attention U-net with augmentation

Architecture	Dice (in %)
1000 + T1 + 1	81.63
1500 + T2 + 1	80.20
1500 + T1 + 1 + high	79.95
2500 + T1 + 2	77.53

Table 6 Patchwise approach results

Batch-size	Dice (in %)	Loss
4	60.54	0.97635
8	76.40	0.98356
16	78.06	0.98459
32	78.63	0.98439
64	78.64	0.98485
243	78.07	0.98466
486	77.22	0.98369
972	74.13	0.98297
1728	70.67	0.98097

5.1 Comparison with Other Models

Comparison of the proposed model with other models that participated in the Iseg2019 challenge is shown in Table 7; names of the models are chosen as team names of these models as they did not give any specific name based on their approach. All the values are dice values. All the models are taken from [2]. All these models are part of the Iseg competition and trained on the same dataset. In Table 7 although the proposed model is not the best model to perform this segmentation task, it is better than some models. This paper also includes various experimental results, so that the limitations of these approaches presented in this paper can be understood directly without experimentation.

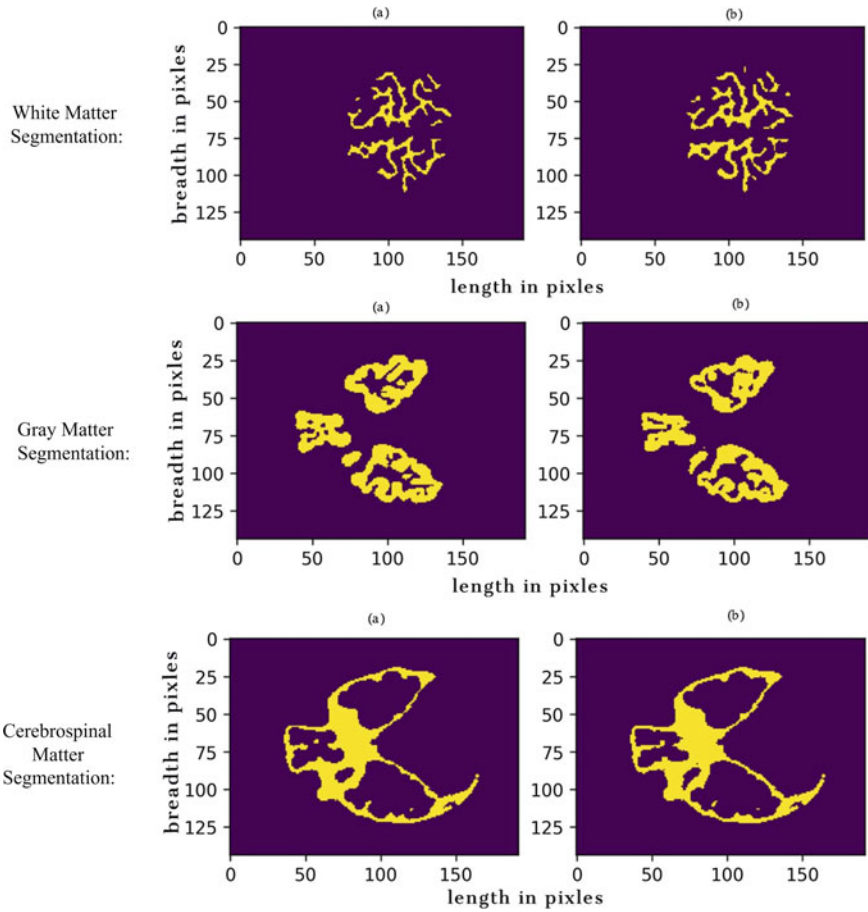


Fig. 3 White matter segmentation, gray matter segmentation, and cerebrospinal matter segmentation, where **a** actual output, **b** predicted output

6 Conclusion

In this paper, a 3D U-net architecture is proposed for accurate segmentation of brain MRI images, and also the importance of deep learning methods in solving segmentation tasks is discussed. Various experimental results based on U-net and attention U-net were presented. The model proposed has a dice of 85 for the white matter, which is less than 90, which is the highest dice value for white matter, as shown in the comparison table. The model could not retain the information of relevant features in the process of upsampling and downsampling; this can be considered a limitation of this model. Future research can be concentrated on two crucial aspects: improving the segmentation: improving the MRI image quality, and identifying some other technique to extract features without losing much information.

Table 7 Comparison with other models

Model	White matter	Gray matter	Cerebrospinal fluid
Proposed model	0.85	0.88	0.93
Q111111	0.90	0.92	0.95
Fight autism	0.90	0.92	0.94
BIG	0.89	0.90	0.95
SLHC MICCAI	0.88	0.90	0.94
World seg	0.85	0.88	0.89
UBCOO1	0.81	0.86	0.87
Legend	0.84	0.86	0.90
Climb mountains	0.83	0.86	0.92
Taitantian	0.82	0.85	0.91

References

- Melbourne A, Cardoso MJ, Kendall GS, Robertson NJ, Marlow N, Ourselin S (2012) NeoBrainS12 challenge: adaptive neonatal MRI brain segmentation with myelinated white matter class and automated extraction of ventricles I–IV. In: Proceedings of the MICCAI grand challenge: neonatal brain segmentation, pp 16–21
- Sun Y, Gao K, Wu Z, Li G, Zong X, Lei Z, Wang L (2021) Multi-site infant brain segmentation algorithms: the iSeg-2019 challenge. *IEEE Trans Med Imaging* 40(5):1363–1376
- Wang L, Nie D, Li G, Puybareau É, Dolz J, Zhang Q et al (2019) Benchmark on automatic six-month-old infant brain segmentation algorithms: the iSeg-2017 challenge. *IEEE Trans Med Imaging* 38(9):2219–2230
- Shi F, Yap PT, Wu G, Jia H, Gilmore JH, Lin W, Shen D (2011) Infant brain atlases from neonates to 1- and 2-year-olds. *PLoS ONE* 6(4):e18746
- Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
- Xue H, Srinivasan L, Jiang S, Rutherford M, Edwards AD, Rueckert D, Hajnal JV (2007) Automatic segmentation and reconstruction of the cortex from neonatal MRI. *Neuroimage* 38(3):461–477
- Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham, pp 234–241
- Nie D, Wang L, Gao Y, Shen D (2016) Fully convolutional networks for multi-modality iso-intense infant brain image segmentation. In: 13th international symposium on biomedical imaging (ISBI). IEEE, pp 1342–1345
- Lei Z, Qi L, Wei Y, Zhou Y (2019) Infant brain MRI segmentation with dilated convolution pyramid downsampling and self-attention. *arXiv preprint arXiv:1912.12570*
- Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O (2016) 3D U-net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham, pp 424–432

11. Kamnitsas K, Ledig C, Newcombe VF, Simpson JP, Kane AD, Menon DK, Glocker B (2017) Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med Image Anal* 36:61–78
12. Xu J, Li Z, Zhang M, Liu J (2020) Reluplex made more practical: leaky ReLU. In: 2020 IEEE symposium on computers and communications (ISCC), pp 1–7
13. Sørensen T (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons
14. Hamann B, Chen JL (1994) Data point selection for piecewise linear curve approximation. *Comput Aided Geom Des* 11(3):289–301

Analysis of Deep Learning Architecture-Based Classifier for the Cervical Cancer Classification



R. Chandrababha and Seema Singh

1 Introduction

The fourth most ubiquitous cancer affecting women worldwide is cervical cancer [1]. Every year among women, cancer diagnosis is greater than 500,000 and over 300,000 deaths occur globally. There is a lack of screening and HPV vaccination programs in economically developing countries leading to 90% of incidence cases of cervical cancers. However, in affluent countries, both the incidence and mortality have been reducing due to the introduction of proper screening programs [2]. In most cases, before the disease could become clinically marked, the cervical cancer experiences a long asymptomatic phase. This may lead to death also. Hence, through the process of regular screening, the early and timely detection of cancer is possible to prevent the cancer progression [3].

The continual screening is very much essential among women to enable the doctors to identify cancer at an early stage before it could reach the final stage. The Pap smear test (smear collected from uterine cervix and it is stained) is used as a screening method to detect cervical, but the alertness within the public about the screening test is limited. Every three years, a routine cancer screening must be done, and for every five years, a Pap smear with an HPV DNA test is recommended as a screening method [4].

Figure 1a shows the abnormal cells with the enlarged nucleus as one of the features in identifying the cancer cells, and Fig. 1b depicts the normal cells without any

R. Chandrababha (✉)

Electronics and Communication Engineering, BMS Institute of Technology and Management,
Bangalore, India

e-mail: chandra@bmsit.in

S. Singh

Electronics and Telecommunication Engineering, BMS Institute of Technology and Management,
Bangalore, India

e-mail: seemasingh@bmsit.in

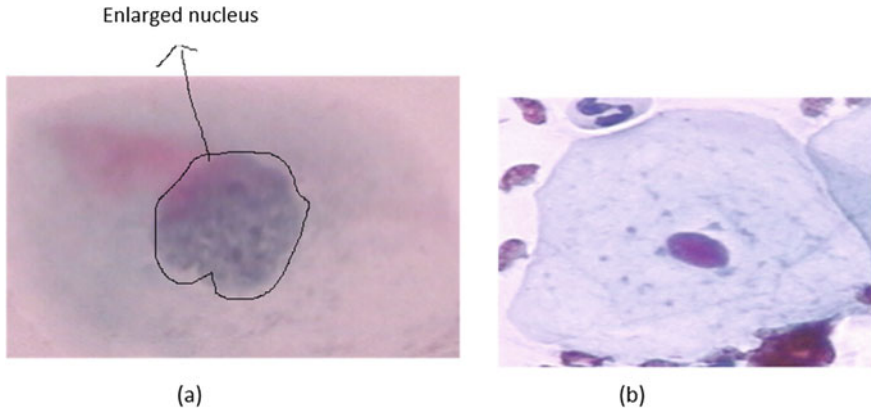


Fig. 1 Standard dataset: **a** abnormal cell and **b** normal cell

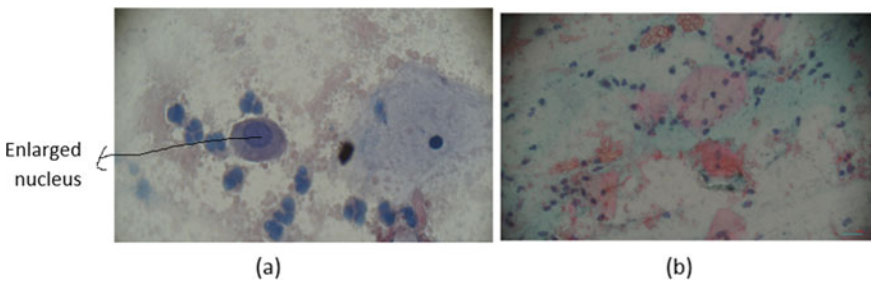


Fig. 2 Real clinical data: **a** abnormal cell and **b** normal cell

changes in the nucleus size. Figure 2 shows the clinical data. Figure 2a shows the abnormal cells with an enlarged nucleus, and Fig. 2b depicts the normal cells without any changes in the nucleus size.

To assist the clinical providers, in providing the report for the subsequent patient management, the general classification of cancer images “normal” from “abnormal” is recommended [5].

The manual screening procedure is time-consuming and not economical, and also, it is error-prone since the manual procedure may drain the paramedical workers. Hence, there is a requirement for an automatic diagnosis system that can technically support the clinicians that can reduce cost, time, and expertise needed for cervical cancer screening.

Therefore, a system must be designed to make a faster decision at a faster rate in all aspects. The automated system must adapt itself to fresh and complicated cases of the cervical cancer. With the active participation of medical paramedical health workers and the recent advent of technology, an automated diagnosis system is to be formulated and implemented.

The tremendous growth in artificial intelligence (AI) has given a supporting technical hand to doctors in providing personalized health care to patients, enhancing the efficiency of doctors in making a diagnosis of any diseases at an early stage [6].

2 Literature Review

Cervical cells can be classified into cancerous (abnormal) and non-cancerous (normal) cells. In this task of classification, the techniques related to image processing, machine learning, and deep learning are applied [7–10].

Iwai and Tanaka [7] presents a method of the segmentation techniques using image processing to detect the cells with nuclear enlargement and color density. The true positive and true negative were 97% and 55%, respectively. Sokouti et al. [11] discusses the attributes of the normal, the precancerous, and the cancerous cell images that are mined using image processing techniques. The cells were classified as cancerous cells, normal cells, and precancerous cells. The unique linear plot result was 100% accuracy. Ashok and Aruna [12]. presents the diagnosis using Support Vector Machine classifier. The thresholding segmentation was performed. Both the texture and shape features were considered for the classification. Mutual information, sequential floating forward search, sequential forward search, and random subset feature methods were applied to extract the features. The sensitivity of 0.98, the accuracy of 0.98, and the specificity of 0.975 were achieved. Su et al. [13] proposes an automatic system with image segmentation with cascaded two-level classifier considering 28 features with respect to morphology and texture. A higher identification rate of 95.642% was achieved with integrated two-level classifier. Sharma et al. [14] discusses a system to classify the cancer cells. Edge detection was applied to extract the feature like area and perimeter. Normalization is performed through the min–max method. Applying the KNN classifier (fivefold validation), the cell was classified with accuracy rate of 82.9%. Kumar et al. [15] presents a system to classify the cells using the K-nearest neighborhood. The features considered for the classification were gray level, texture features, color gray level, color-based features, Law's Texture Energy-based features, Tamura's features, and wavelet features. The accuracy, specificity, and sensitivity of 0.92, 0.94, and 0.81 were achieved, respectively. Singh et al. [16] proposes an image processing and watershed algorithm for feature removal, and a neural network classifier is applied for the classification; the accuracy achieved was 79%. Mustafa et al. [17] presents an image analyzer feature which was applied in feature extraction. The extracted features considered were perimeter, red, blue, green, intensity2, intensity1, and saturation. The Hierarchical Hybrid Multi-layered Perceptron is used as a classifier. The accuracy achieved was 94.29%. Chen et al. [18] presents a Personal Computer-based Cellular Image Analysis system to extract the features like nuclear size, nuclear to cytoplasm's ratio, nuclear shape, and nuclear texture. The classifier used was Support Vector Machine classifier (2-cluster, 4-cluster)—filter method and wrapped method. The accuracy achieved was 97.16%.

Shin et al. [19] discusses the convolution neural network model analysis which can be adapted to design a high-performance automated system for medical imaging tasks. Segmentation deep learning techniques were applied in the segmentation process. The author presents region of interest detection with deep learning based on convolution neural network. Deep learning algorithms provide superior performance than traditional machine learning for the classification of cervicography images into different classes [20]. The faster region convolutional neural network system [21] was applied in the classification task. The sensitivity of 99.4%, specificity of 34.8%, and 0.67 of the area under the curve were obtained. Wu et al. [22] presents an intelligent simple convolutional neural network (CNN) that is trained and tested by original image group (3012 datasets) and augmented image group (108,432) datasets. The works discuss that through augmentation the classification accuracy was improved.

From the above survey, conventional machine learning and deep convolution neural networks have influenced the classification of cervical cells. Classification can be performed by extracting the features and then followed by a classifier. Deep convolution neural network classifies directly without any manual feature extraction with image processing. However, validation of the deep learning models is limited only for the standard benchmark dataset.

In this work, the pre-trained architectures are considered. The model is fine-tuned through transfer learning. The trained network is validated and tested for the standard benchmark dataset and for the real-time clinical data also. Comparative analysis to find better architecture and training parameters which is also verified with clinical data.

3 Methodology

In this work, the input images from both standard and real-time clinical dataset are considered. The orientation of the samples is changed through the data augmentation process. Through the adaption of transfer learning process, the pre-existing deep convolution neural networks are fine-tuned according to our work. A very few important hyper-parameters are fine-tuned. The tuned architectures are trained, validated, and tested for the dataset. The considered architectures are trained with hardware Processor Intel E-2224G CPU with 3.50 GHz, 64-bit operating system, and x64-processor. The software platform used is MATLAB R2021—academic use. Figure 3 discusses the proposed methodology.

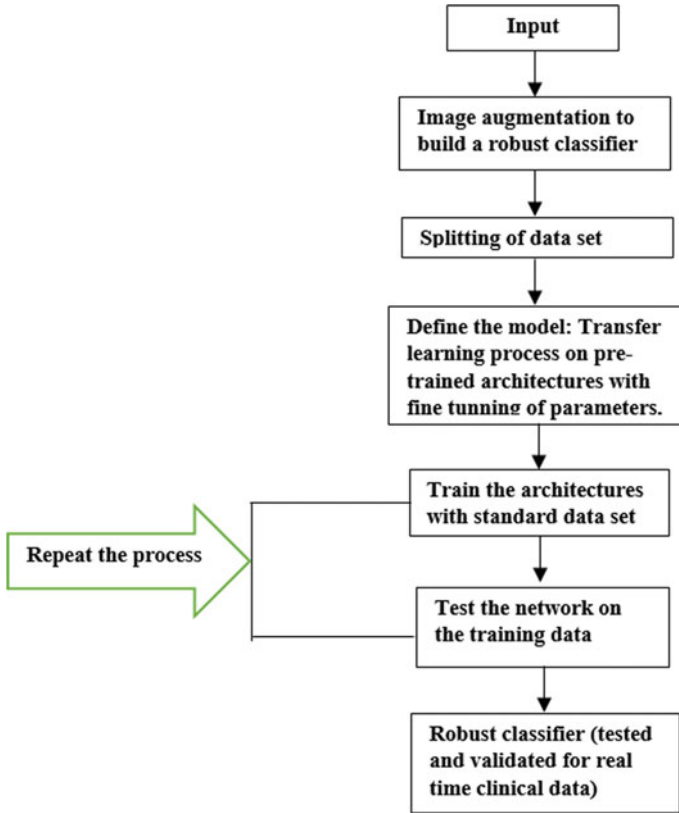


Fig. 3 Methodology of the proposed work

3.1 Image Dataset

Standard datasets (HERlev Dataset)

The image dataset for the work was collected from the HERlev Dataset (standard) for Pap smear images and real-time clinical data. The HERlev Dataset (standard) set comprises of total images of 917 cell images; out of this, 242 belong to normal classes and 675 belong to abnormal classes.

Real-time clinical dataset

The real-time clinical data is captured from the Labomed LX 300 trinocular microscope. Real-time clinical data comprises total images of 10 cell images; out of this, 4 belong to normal classes and 6 belong to abnormal classes.

Data augmentation

The orientation of the samples is restricted in the dataset. Hence, in order to have a robust architecture which can classify any type of orientation of samples data augmentation techniques are applied. The different data augmentation techniques are image rotation, image reflection, and image translation.

Deep learning-based architectures

For image analysis and classification, deep learning networks are very appropriate that results in successful image classification [23]. Generally, a deep learning networks consists of (a) convolutional layer, (b) activation layer, (c) pooling layer, and (d) fully connected layer [4].

For the proposed work, the deep learning architectures GoogleNet and Alexnet are considered.

The GoogleNet model consists of 22 deep layers and 27 pooling layers with 9 inception modules stacked up linearly. The outputs of inception modules are connected to pooling layer [24].

The Alexnet architecture consists of five convolution layers, and last 3 are fully connected. All the outputs of convolutional and fully connected layers are connected to a softmax activation layer which produces 1000 class labels [25].

For medical applications, building a model from scratch is unpractical due to limited clinical data and the requirement of computational resources. Hence, the concept of transfer learning arises [24, 26, 27].

3.2 A Fine-Tuned Classifier Models

Transfer learning is similar to the way humans may relate their information of an assignment to enable the learning of another assignment [28, 29]. The fine-tuning of an already existing pre-trained network is termed transfer learning. In this process, the last fully connected layers of the models are replaced with the new classification layer. The new classification layer comprises two nodes at the output representing the two classes of cancer.

The hyper-parameters considered for the fine-tuning are network learning rate-0.0001, stochastic gradient descent momentum, momentum of 0.9, and epoch of 06 and 30. An epoch refers to one cycle through the full training dataset.

The splitting of data for training, testing, and validation is 60:20:20. The number of normal images considered for the testing, training, and validation are 145, 48, and 48, respectively. The number of abnormal images considered for the testing, training, and validation is 405, 135, and 184, respectively.

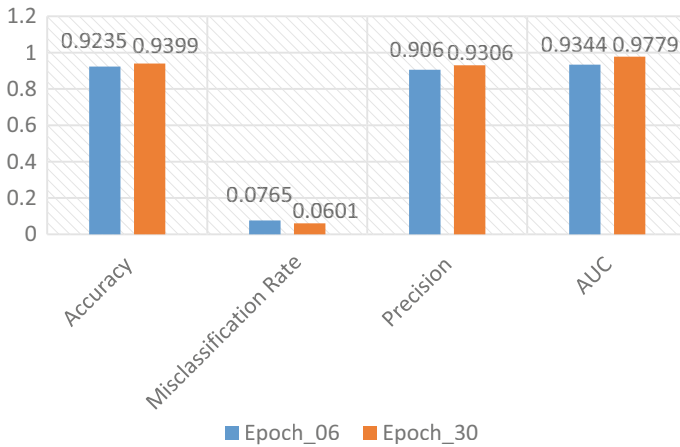


Fig. 4 Classifier A performance for epochs 06 and 30 with HERlev Dataset (standard)

4 Results and Discussion

Analysis is presented with respect to variations in dataset, architectures, and fine-tuned parameters which are validated with the clinical data and HERlev Dataset (standard). An accuracy (training, validation, testing), error rate or misclassification rate, precision, and area under the receiver operating characteristic curve are considered in the work to evaluate and analyze the performance of the trained models.

4.1 Classifier A-Alexnet

Figure 4 displays the performance metric of the classifier A for two different epochs which shows that with increase in number of epochs, the model is trained better. The other measuring parameters will also improve. The classifier incorrectly predicting the abnormal class is decreased by the network as the number of iterations increases.

For larger epochs, the model is robust in identifying cancer and non-cancerous cells with decreased misclassification rate for the test dataset.

4.2 Classifier B-Google Net

From Fig. 5, the Classifier B behavior for the two levels of epoch is almost constant with not much variations.

As the iterations (number of epochs) increase in the training phase, the model’s area under curve is increased by 0.04.

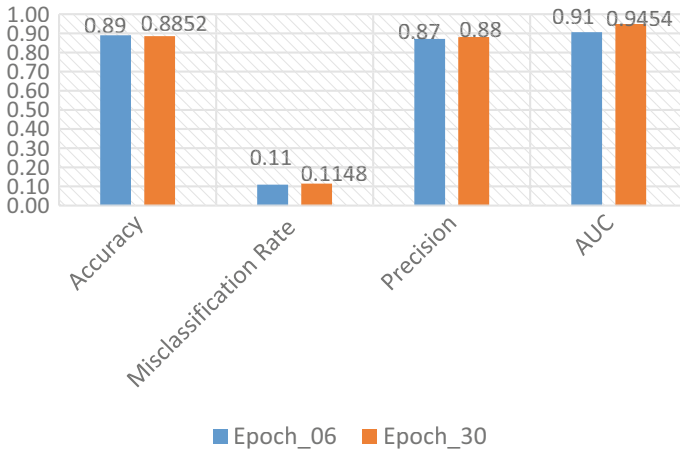


Fig. 5 Classifier B confusion matrix parameters for different epochs with HERlev Dataset (standard)

4.3 Comparison Measure of the Classifiers

As in Fig. 6, the Classifier B performance parameters have less prediction rate compared to Classifier A.

The Classifier A performance is showing a good response on the test dataset. The accuracy, precision, and AUC are 0.9508, 0.9565, and 0.981, respectively.

Fig. 6 Illustrates the performance of Classifier A and Classifier B architectures for the HERlev Dataset (standard)

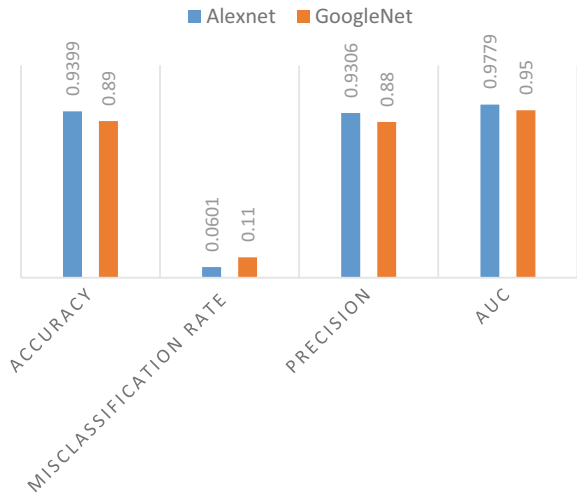
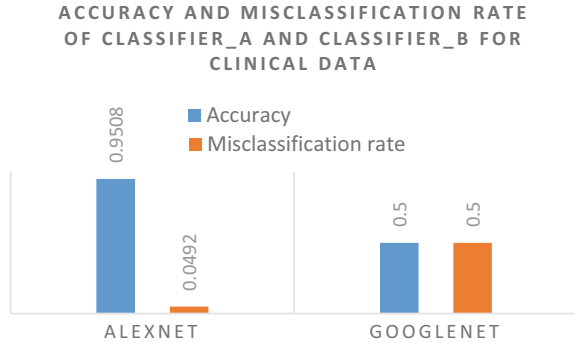


Fig. 7 Measure of accuracy and misclassification rate for the real-time clinical data



4.4 Evaluating the Classifiers for the Real-Time Clinical Dataset

From results, the classifier performance is better when the network is trained for the 30 epochs. The trained model with 30 epochs is further considered for the validation with the real-time clinical data.

Figure 7 displays that the Classifiers A and Classifier B are predicting cancer with an accuracy of 0.9508 and 0.5, respectively.

5 Conclusion

The incidence and mortal rates of cervical cancer have to be reduced among women. A very efficient automated system is essential for the classification of cancer into non-cancerous and cancerous cases. Artificial intelligence is already on terms in providing a technical hand (automated system) to health care. Deep learning architectures are in the classification of cervical cancer to build an automated system.

In this work, the existing architectures Alexnet and GoogLeNet are fine-tuned through the transfer learning process for the classification task. The two classification classes considered for this work are two classes, i.e., normal and abnormal. The fine-tuned pre-existing models are trained by fine-tuning hyper-parameters such as learning rate stochastic gradient descent momentum and number of epochs. The HERlev Dataset (standard) set alone is applied for the training and validating phase, whereas for the testing phase, both the HERlev Dataset (standard) and real-time clinical dataset are applied. There is 1% variation in the performance of GoogLeNet for the epochs 6 and 30. The performance of the pre-trained Alexnet is increased by 4–8% for the epoch 30 compared to epoch 6. For 30 epoch, an accuracy, precision, and AUC are 0.9399, 0.9306, and 0.9779, respectively.

Further, the two pre-trained classifiers that are trained, validated, and tested with 30 epochs are considered for the new non-trained real-time clinical test data. The

Alexnet architecture executes better with clinical data compared to the GoogleNet architecture with accuracy of 0.9508 and misclassification rate of 0.0492.

In the future, the architecture robustness in classification can also be enhanced. Further, the work can be extended for the multiclass classification and verifying for the different datasets.

References

1. Jemal A, Center MM, DeSantis C, Ward EM (2010) Global patterns of cancer incidence and mortality rates and trends. *Cancer Epidemiol Biomark Prev* 19(8):1893–1907
2. Cohen PA, Jhingran A, Oaknin A, Denny L (2019) Cervical cancer. *Lancet* 393(10167):169–182. [https://doi.org/10.1016/S0140-6736\(18\)32470-X](https://doi.org/10.1016/S0140-6736(18)32470-X)
3. Canavan TP, Doshi NR (2000) Cervical cancer. *Am Fam Physician* 61(5):1369–1376
4. Saslow D, Solomon D, Lawson HW, Killackey M, Kulasingam SL, Cain J, Garcia FA, Moriarty AT, Waxman AG, Wilbur DC, Wentzensen N, Downs LS Jr, Spitzer M, Moscicki AB, Franco EL, Stoler MH, Schiffman M, Castle PE, Myers ER, ACS-ASCCP-ASCP Cervical Cancer Guideline Committee, American Cancer Society, American Society for Colposcopy and Cervical Pathology, American Society for Clinical Pathology (2012) Screening guidelines for the prevention and early detection of cervical cancer. *CA Cancer J Clin* 62(3):147–172. <https://doi.org/10.3322/caac.21139>. Epub 2012 Mar 14. PMID: 22422631; PMCID: PMC3801360
5. Nayar R, Wilbur DC (2017) The Bethesda system for reporting cervical cytology: a historical perspective. *Acta Cytol* 61(4–5):359–372. <https://doi.org/10.1159/000477556>. Epub 2017 Jul 11. PMID: 28693017
6. Reddy S, Allan S, Coghlan S, Cooper P (2020) A governance model for the application of AI in health care. *J Am Med Inform Assoc* 27(3):491–497. <https://doi.org/10.1093/jamia/ocz192>
7. Iwai, Tanaka T (2017) Automatic diagnosis supporting system for cervical cancer using image processing. In: 2017 56th annual conference of the society of instrument and control engineers of Japan (SICE), pp 479–482. <https://doi.org/10.23919/SICE.2017.8105610>
8. William W, Ware A, Basaza-Ejiri AH, Obungoloch J (2018) A review of image analysis and machine learning techniques for automated cervical cancer screening from pap-smear images. *Comput Methods Programs Biomed* 164:15–22. <https://doi.org/10.1016/j.cmpb.2018.05.034>
9. Hussain E, Mahanta LB, Das CR, Talukdar RK (2020) A comprehensive study on the multi-class cervical cancer diagnostic prediction on pap smear images using a fusion-based decision from ensemble deep convolutional neural network. *Tissue Cell* 65:101347
10. Supriyanto E, Pista NA, Ismail L, Rosidi B, Mengko T (2011) Automatic detection system of cervical cancer cells using color intensity classification
11. Sokouti B, Haghypour S, Tabrizi AD (2012) A pilot study on image analysis techniques for extracting early uterine cervix cancer cell features. *J Med Syst* 36(3):1901–1907. <https://doi.org/10.1007/s10916-010-9649-y>
12. Ashok B, Aruna D (2016) Comparison of feature selection methods for diagnosis of cervical cancer using SVM classifier
13. Su J, Xu X, He Y, Song J (2016) Automatic detection of cervical cancer cells by a two-level cascade classification system. *Anal Cell Pathol* 2016:1–11. Article ID 9535027. <https://doi.org/10.1155/2016/9535027>
14. Sharma M, Singh S, Agrawal P, Madaan V (2016) Classification of clinical dataset of cervical cancer using KNN. *Indian J Sci Technol*
15. Kumar R, Srivastava R, Srivastava S (2015) Detection and classification of cancer from microscopic biopsy images using clinically significant and biologically interpretable features. *J Med Eng* 2015:457906. <https://doi.org/10.1155/2015/457906>

16. Singh S, Tejaswini V, Murthy RP, Mutgi A (2015) Neural network based automated system for diagnosis of cervical cancer. *Int J Biomed Clin Eng*
17. Mustafa N, Mat Isa NA, Mashor MY, Othman NH (2007) New features of cervical cells for cervical cancer diagnostic system using neural network. In: *International symposium on advanced technology*
18. Chen YF, Huang PC, Lin KC, Lin HH, Wang LE, Cheng CC, Chen TP, Chan YK, Chiang JY (2014) Semi-automatic segmentation and classification of pap smear cells. *IEEE J Biomed Health Inform* 18(1):94–108. <https://doi.org/10.1109/JBHI.2013.2250984>
19. Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans Med Imaging* 35(5):1285–1298. <https://doi.org/10.1109/TMI.2016.2528162>
20. Park YR, Kim YJ, Ju W et al (2021) Comparison of machine and deep learning for the classification of cervical cancer based on cervicography images. *Sci Rep* 11:16143. <https://doi.org/10.1038/s41598-021-95748-3>
21. Tan X, Li K, Zhang J et al (2021) Automatic model for cervical cancer screening based on convolutional neural network: a retrospective, multicohort, multicenter study. *Cancer Cell Int* 21:35. <https://doi.org/10.1186/s12935-020-01742-6>
22. Wu M, Yan C, Liu H, Liu Q, Yin Y (2018) Automatic classification of cervical cancer from cytological images by using convolutional neural network. *Biosci Rep* 38(6):BSR20181769. <https://doi.org/10.1042/BSR20181769>
23. Krizhevsky A, Sutskever I, Hinton G (2012) ImageNet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst*. <https://doi.org/10.1145/3065386>
24. Han J, Kamber M, Pei J (2012) Classification: basic concepts, chap 8. In: Han J, Kamber M, Pei J (eds) *Data management systems, data mining*, 3rd edn. The Morgan Kaufmann series. Morgan Kaufmann, pp 327–391. ISBN 9780123814791. <https://doi.org/10.1016/B978-0-12-381479-1.00008-3>. <https://www.sciencedirect.com/science/article/pii/B9780123814791000083>
25. Szegedy C, Liu W, Jia YQ, Sermanet P, Reed S, Anguelov D (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Boston, MA, pp 1–9
26. Han J, Kamber M, Pei J (2012) Classification: advanced methods, chap 9. In: Han J, Kamber M, Pei J (eds) *Data management systems, data mining*, 3rd edn. The Morgan Kaufmann series. Morgan Kaufmann, pp 393–442. ISBN 9780123814791. <https://doi.org/10.1016/B978-0-12-381479-1.00009-5>. <https://www.sciencedirect.com/science/article/pii/B9780123814791000095>
27. Chandraprabha R, Hiremath S (2021) Computer processing of an image: an introduction. In: *Handbook of research on deep learning-based image analysis under constrained and unconstrained environments*. IGI Global, pp 1–22
28. Mikołajczyk, Grochowski M (2018) Data augmentation for improving deep learning in image classification problem. In: *2018 international interdisciplinary PhD workshop (IIPHDW)*, pp 117–122. <https://doi.org/10.1109/IIPHDW.2018.8388338>
29. Ying X (2019) An overview of over fitting and its solutions. *J Phys Conf Ser* 1168:022022
30. Jantzen J, Dounias G (2006) Analysis of pap-smear image data
31. Chandraprabha R, Singh, S (2016) Artificial intelligent system for diagnosis of cervical cancer: a brief review and future outline. *J Latest Res Eng Technol*
32. Jones OT, Calanzani N, Saji S, Duffy SW, Emery J, Hamilton W, Singh H, de Wit NJ, Walter FM (2021) Artificial intelligence techniques that may be applied to primary care data to facilitate earlier diagnosis of cancer: systematic review. *J Med Internet Res* 23(3):e23483. <https://doi.org/10.2196/23483>. PMID: 33656443; PMCID: PMC7970165

Covid Vaccine Adverse Side-Effects Prediction with Sequence-to-Sequence Model



Shyam Zacharia  and Ashwini Kodipalli 

1 Introduction

The situations in the entire world changed in recent times because of the outbreak of COVID-19 pandemic. Originating in Wuhan, the disease started spreading drastically and was declared a pandemic. The increase in the infections lead to the increased deaths, increased lay-offs, increased crisis, and decreased human population. There were many measures and actions taken by the World Health Organization (WHO) to prevent the spread of this disease. These measures included wearing masks, washing hands, and maintaining social distance between the people. Global immunization scheme is also one of the safest and reliable measures of preventing disease spread. The vaccines are administered for various age groups in all the countries and have proved the efficacy. Nevertheless, these vaccines have side effects along with their preventive nature among different masses of populations. The vaccine side effects are not only being temporarily harmful but have also proved fatal in some cases. The preventive measures of the history of previous pandemics elaborate on the seriousness of any pandemic situations. The previous pandemics were eradicated because of the administration of vaccines along with preventive measures. One such example of complete eradication of the disease is polio. In 1988, more than 350,000 children were paralyzed with polio [1]. With the effect of immunization, there are hardly few polio cases globally. Not only polio, but also eradication of smallpox was achieved with the increase in vaccination [2]. The current work is aimed at predicting the side

S. Zacharia
British Telecom, Bangalore, India
e-mail: shyam.zacharia@bt.com

A. Kodipalli (✉)
Department of Artificial Intelligence and Data Science, Global Academy of Technology,
Bangalore, India
e-mail: dr.ashwini.k@gat.ac.in

effects of COVID-19 vaccine among different sets of population mainly based on age and gender.

The organization of the paper is as follows: Sect. 2 describes the literature survey; Sect. 3 describes the methodology; Sect. 4 describes the results. Section 5 concludes the paper.

2 Literature Survey

In the previous works, the use of machine learning and deep learning has shown significant results in the field of automation. Menni et al. conducted a survey on the side effects of vaccination after eight days of vaccination in China by considering the data recorded in a symptom study app. BNT162B2 and ChAdOx1 vaccines gave the reduced risk of SARS-Cov2 infection after 12 days of vaccination [3]. Leng et al. selected the vaccines based on the seven different attributes, among which vaccine decision, effectiveness, side effects, and number of doses are important attributes. The discrete choice models were based on Bayesian information criteria (BIC) and Alkaline information criteria [4].

Riad et al. based on a survey for a month on the Czech healthcare workers vaccinated with Pfizer and BioNTech vaccine, found the common side effects of vaccine to be pain at the injection site, fatigue, muscle pain, chills, and joint pain. The chi-squared test and ANOVA were used with a significance level of <0.05 [5]. Alam et al. used deep learning-based technique for analyzing the COVID-19 vaccine response from the twitter data. To achieve this, data were downloaded from Kaggle. The data characters were detokenized, and performance is checked by using long short-term memory (LSTM) and bi-directional LSTM. It was found that both the models gave good performance [6].

Zaman et al. collected the information in several vaccination centers among different age groups and applied deep learning models to predict the side effects due to vaccine. Serious conditions such as cardiovascular diseases and diabetes are considered as the main variables for predicting the side effects. The deep learning techniques applied are artificial neural network (ANN), LSTM, and GRU were applied and found that GRU gave the best accuracy [7]. Muneer et al. predicted the mRNA vaccine degradation using deep learning techniques for which the data were obtained from Kaggle, and two models were proposed which is deep hybrid neural network models—GCN-GRU and GCN-CNN. MCRMSE score of 0.22614 and 0.34152 and GCN-GRU pre-trained model achieved AUC score of 0.938 [8].

Jarynowski et al. surveyed the effects of Sputnik V vaccine in Russia through social media analysis in which natural language processing is used to extract the text from telegram groups. BERT technique of natural language processing was used to perform multi-label classification which resulted in AUROC value of 0.991. Fever, pains, chills, fatigue, and lot such symptoms were recorded [9]. Aryal and Bhattarai proposed two models based on machine learning approach which is Naïve Bayes and deep learning approach which is LSTM. The twitter data were extracted, and

preprocessing was done. It was found that the accuracy of LSTM model is higher than the machine learning model by 7% [10].

Kerr et al. prepared a textual question and answer format for the analysis of the efficacy and side effects of vaccines. Based on the responses obtained, information tests are conducted. Participants were randomized by Qualtrics randomization tool and showed that there is no effect of increasing the vaccinations but by individual perspective [11]. Sen et al. prepared a graphical visualization of the analysis of spread of COVID-19 pandemic situation with the help of current pandemic situation using machine learning regression algorithms and obtained satisfying results. However, the models are to be explored to obtain better results with the help of deep learning techniques [12]. Ashwini et al. [13] using LSTM predicted the number of possible cases for the next 10 days.

Majority of the literatures referred are using deep learning algorithms for sentiment analysis and survey-based approach to analyze the COVID vaccine symptoms. In the present work, the analysis of vaccine side effects is given importance.

3 Methodology

3.1 Dataset Used

The Vaccine Adverse Event Reporting System (VAERS) was created by the Food and Drug Administration (FDA) and Centers for Disease Control and Prevention (CDC) to receive reports about adverse events that may be associated with vaccines in US [14]. Covid-19-related VERS data have been used in the current experiment.

Table 1 provides a detailed description of the data provided in each field of the VAERSDATA.CSV file.

Total distinct count of VAERS_ID records were 7462. We could see the distribution of different side effects with respect to age and gender in the data. For example, we can observe that death adverse effect is high in old ages as shown in Fig. 1.

Table 1 Description of the dataset

Header	Type	Description of contents
VAERS_ID	Num (7)	VAERS identification number
AGE_YRS	Num	Age in years
CAGE_YR	Num	Calculated age of patient in years by considering exact number of months also
SEX	Char (1)	Sex
SYMPTOM_TEXT	Char (32,000)	Reported symptom text

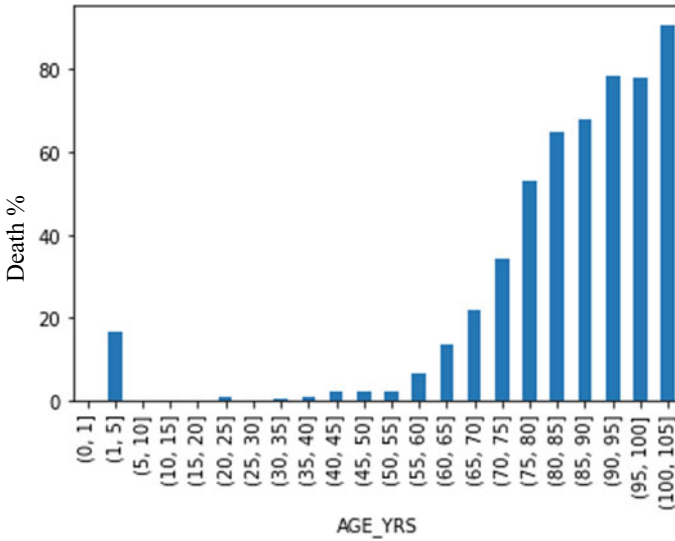


Fig. 1

3.2 Structured Data to Sequence Learning Problem Formulation

Our aim was to predict adverse side-effect symptoms based on age and sex. For age and gender input, the below side effects were reported:

- COVID-19, Chills, Death, Fatigue, Headache, Pain

To process this structured data, in Seq2Seq model, we had to convert the data into input sequence (encoder input) and output sequence (decoder output) format. So, we formulated the data into format as shown in Table 2.

Once we formulated 7462 records into format, the total distinct records came to 184. Then, we split the data into train and test category with Table 3 record count.

Table 2 Seq2Seq model

Encoder input		Decoder output
AGE	SEX	ONE-HOT ENCODED SIX SIDE EFFECTS

Table 3 Train and test data

Train	Test
147	37

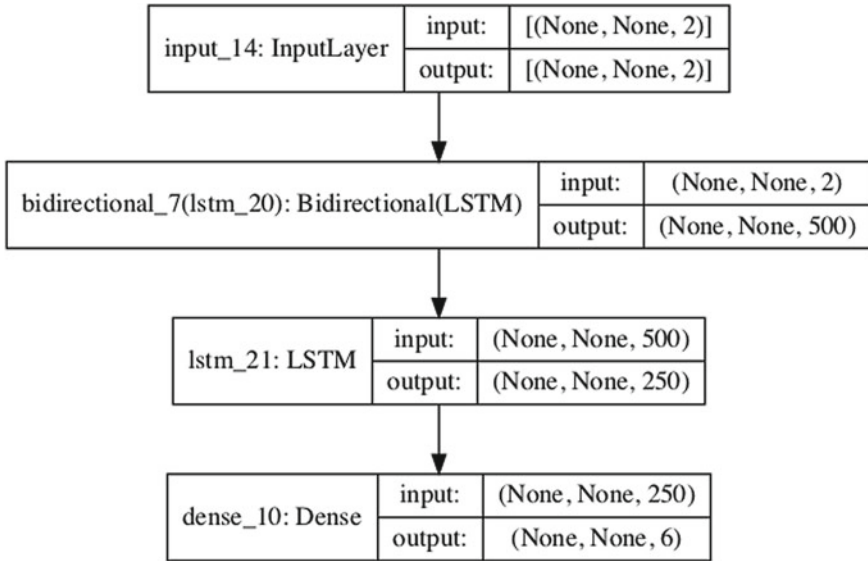


Fig. 2 Architecture of the model

3.3 The Seq2Seq Model

Sequence-to-sequence model, with LSTM, maps the input sequence to a fixed sized vector and maps the vector to another target sequence [14]. Due to this capability, Seq2Seq models are widely using in sequence learning tasks such as machine translation [15], speech recognition [16], and video captioning [17].

In the input layer, we are giving age and gender as input. Bidirectional LSTM layer was used as encoder layer. Encoder output fed into LSTM-based decoder layer. We added dense layer at the end of sequence-to-sequence layer to capture the adverse symptoms. Sigmoid activation function used in this output layer. The model is having architecture as shown in Fig. 2.

The output of the decoder layer fed into dense layer and generated 6-dimensional vector output to predict the symptoms. The model used binary cross-entropy loss function with Adam optimizer.

4 Result

We have obtained 88% micro-average accuracy with the abovementioned LSTM-based Seq2Seq model. Please find the accuracy matrix for the result our multi-label classification output as shown in Fig. 3.

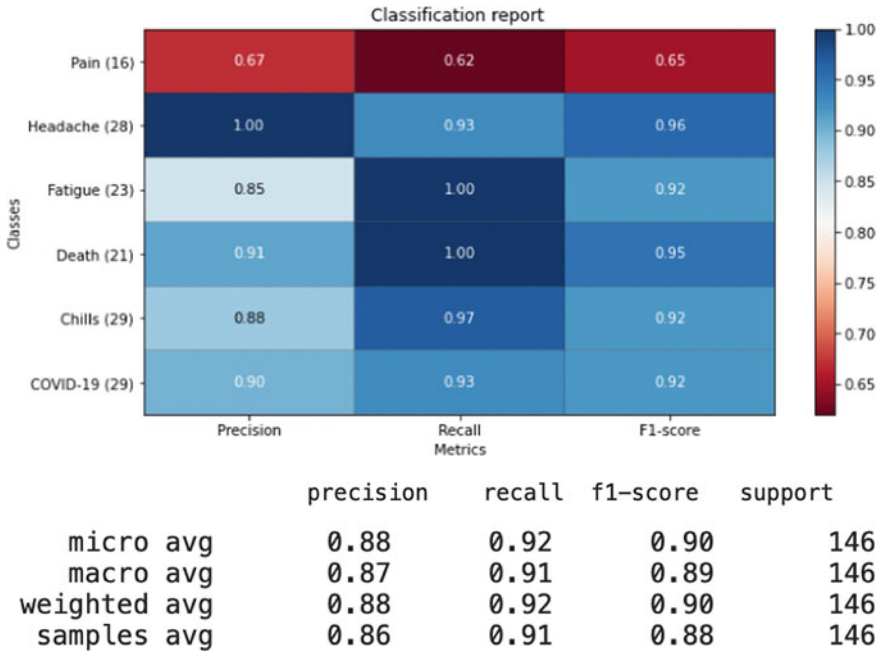


Fig. 3 Classification report

We have also tried with linear activation function in the last layer and loss function with mean squared error. But could only obtain the micro-average accuracy as 85%. Could observe that, after sigmoid transformation, the multi-label one-hot encoded values were having values to near zero or 1, and binary cross-entropy yields higher accuracy at this time.

5 Conclusion

Due to the COVID-19 pandemic, there was tremendous change in the multiple strata of life such as clinical practice, regular life, work from home culture, online classes to students, economy, policing, and crime control. In order to contain the outbreak of COVID-19, many vaccines are discovered. Each of the vaccine underwent lot of preclinical and clinical trials before regulatory approach. Still a few adverse events due to side effects of vaccine are reported in the world. Therefore, development of an efficient methodology to predict the possible adverse side effects of vaccine is customary. In this direction, the present study predicts the adverse side effects of COVID-19 vaccine using deep learning computational models. The proposed methodology when tested with VAERS data provided 88% of accuracy using LSTM. The developed methodology can help in predicting the side effects

due to COVID-19, which can be used by the healthcare department to select a country/region/race/lifestyle category/comorbidities/age/gender-specific vaccine to minimize the side effects. Also, observed that, model formulated with the final layer output with sigmoid activation function along with binary cross-entropy loss function improved the accuracy level.

References

1. Bigouette JP, Wilkinson AL, Tallis G, Burns CC, Wassilak SGF, Vertefeuille JF (2021) Progress toward polio eradication—worldwide, January 2019–June 2021. *Morb Mortal Wkly Rep* 70(34):1129
2. Moss B, Smith GL (2021) Research with variola virus after smallpox eradication: development of a mouse model for variola virus infection. *PLoS Pathog* 17(9):e1009911
3. Menni C, Klaser K, May A, Polidori L, Capdevila J, Louca P, Sudre CH et al (2021) Vaccine side-effects and SARS-CoV-2 infection after vaccination in users of the COVID symptom study app in the UK: a prospective observational study. *Lancet Infect Dis*
4. Leng A, Maitland E, Wang S, Nicholas S, Liu R, Wang J (2021) Individual preferences for COVID-19 vaccination in China. *Vaccine* 39(2):247–254
5. Riad A, Pokorná A, Attia S, Klugarová J, Koščik M, Klugar M (2021) Prevalence of COVID-19 vaccine side effects among healthcare workers in the Czech Republic. *J Clin Med* 10(7):1428
6. Alam KN, Khan MS, Dhruva AR, Khan MM, Al-Amri JF, Masud M, Rawashdeh M (2021) Deep learning-based sentiment analysis of COVID-19 vaccination responses from twitter data. *Comput Math Methods Med* 2021
7. Zaman FU, Siam TR, Nayen Z. Prediction of vaccination side-effects using deep learning
8. Muneer A, Fati SM, Akbar NA, Agustriawan D, Wahyudi ST (2021) iVaccine-deep: prediction of COVID-19 mRNA vaccine degradation using deep learning. *J King Saud Univ-Comput Inf Sci*
9. Jarynowski A, Semenov A, Kamiński M, Belik V (2021) Mild adverse events of Sputnik V vaccine in Russia: social media content analysis of telegram via deep learning. *J Med Internet Res* 23(11):e30529
10. Aryal RR, Bhattarai A (2021) Sentiment analysis on covid-19 vaccination tweets using Naïve Bayes and LSTM. *Adv Eng Technol Int J* 1(1):57–70
11. Kerr JR, Freeman ALJ, Marteau TM, van der Linden S (2021) Effect of information about COVID-19 vaccine effectiveness and side effects on behavioural intentions: two online experiments. *Vaccines* 9(4):379
12. Sen S, Thejas BK, Pranitha BL, Amrita I (2021) Analysis, visualization and prediction of COVID-19 pandemic spread using machine learning. In: *Innovations in computer science and engineering*. Springer, Singapore, pp 597–603
13. Raj A, Umrani NR, Shilpashree GR, Audichya S, Kodipalli A, Martis RJ (2021) Forecast of covid-19 using deep learning. In: *2021 IEEE international conference on electronics, computing and communication technologies (CONECCT)*, pp 1–5. <https://doi.org/10.1109/CONECCT52877.2021.9622721>
14. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: *Advances in neural information processing systems*, pp 3104–3112
15. Venugopalan S, Rohrbach M, Donahue J, Mooney R, Darrell T, Saenko K (2015) Sequence to sequence-video to text. In: *Proceedings of the IEEE international conference on computer vision*, pp 4534–4542
16. Chiu C-C, Sainath TN, Wu Y, Prabhavalkar R, Nguyen P, Chen Z, Kannan A et al (2018) State-of-the-art speech recognition with sequence-to-sequence models. In: *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, pp 4774–4778
17. VAERS dataset page. <https://vaers.hhs.gov/data/datasets.html>. Accessed 13 Dec 2021

Comparison Between ResNet 16 and Inception V4 Network for COVID-19 Prediction



P. J. Rachana, Ashwini Kodipalli , and Trupthi Rao

1 Introduction

In the month of December 2019, the novel coronavirus which was later named COVID-19 appeared in Wuhan city of China [1]. It has been two years that this deadly virus has stepped in, yet there is no specific medication that deals in combating this virus permanently. We just have a temporary medicine for this virus. The deep learning algorithms have made it easy to diagnose COVID-19. They have proved to be a boon in improving common diagnostic methods. Coronavirus disease has become a serious concern for everyone around the globe. All activities came to hold as soon as this virus appeared. Students were deprived of offline schooling and were forced to confine themselves in a four-walled room and attend online schooling which did not benefit them [2]. This virus has directly affected the health of mankind. Also, it is one of the main reasons for economic crises all over the world. With different variants in it, coronavirus has shown an adverse impact on the patients internally over a long duration. Depending on different variants of this disease, the symptoms also keep changing [3]. Symptoms of the delta variant of this disease include lower pulse rate and drop in the oxygen level, whereas symptoms of the omicron variant are higher pulse rate and there is no significant change in oxygen levels. Artificial

P. J. Rachana

Department of Mechanical Engineering, National Institute of Technology Karnataka, Surathkal, Karnataka, India

e-mail: pjrachana.201me241@nitk.edu.in

A. Kodipalli (✉) · T. Rao

Department of Artificial Intelligence and Data Science, Global Academy of Technology, Bangalore, India

e-mail: dr.ashwini.k@gat.ac.in

T. Rao

e-mail: trupthirao@gat.ac.in

intelligence is useful in detecting infected patients and diagnosing them [4]. COVID-19 belongs to SARS (a family of coronaviruses). Coronaviruses cause dangerous diseases such as Middle East Respiratory Syndrome (MERS). They mainly cause heart and lung diseases. A deep convolutional neural network-based model is helpful in detecting infection in the lungs. Some measures like social distancing, washing hands frequently or using alcohol-based sanitizer, masking, and getting vaccinated help minimize the spread of this deadly virus on masses [5]. Scientists have discovered many vaccines like covishield, covaxin, sputnik, and many more but they give temporary relief only. With new variants popping up, these vaccines will lose their power. A good immune system can help us in fighting this disease. Coronavirus has a long-term impact on the health of the infected person. It mainly affects the growth of the young ones and leads to several unknown infections in adults. This virus brought mankind to a situation wherein there was not enough place to bury the dead people affected by a novel coronavirus. Indirectly, this virus ruined the life of people belonging to underdeveloped and developing countries around the globe. The World Health Organization is also striving hard along with the governments of different countries to make sure people are following a set of guidelines, usually called STANDARD OPERATING PROCEDURE (SOP).

The organization of the paper is as follows: Sect. 2 describes the literature survey, Sect. 3 describes the methodology, Sect. 4 describes the results, and Sect. 5 concludes the paper.

2 Literature Survey

Al Husaini et al. [6] used inception v4 and other modified models like inception mv4 to identify breast disorders. Using inception v4 and mv4 models, he was able to get accurate images. Also, inception v4 and mv4 models are not restricted to be used to detecting and treating early breast cancer but they can be used to detect lung cancer and probably COVID-19. With increasing the number of training epochs, the accuracy of inception mv4 decreases against the inception v4 model. These models helped in detecting cancer present in the human body. These models are efficient in terms of time-consuming and also energy consumption. Talo et al. [7] used AlexNet, ResNet-18, ResNet-34, ResNet-50, and Vgg-16 architectures to identify multi-class brain disease. He used these models on MRI images of the brain. These pre-trained models classify given MR images into different categories which include normal, cerebrovascular, degenerative, and inflammatory diseases. The main aim to introduce the models in detecting the disease is to minimize or prevent the error caused by humans in the manual reading of the MRI images. The manual reading of these MRI images might not give us information about the early detection of multi-class brain disease, whereas the use of these models helps in the early detection of multi-class brain disease and also helps in better treatment and early recovery. Among the above five models, ResNet-50 has the highest accuracy in classification. He et al. [8] used ImageNet to evaluate ResidualNet. She used it on COCO object dataset

detection and also on COCO segmentation. She was able to perform the tasks like ImageNet localization using these models. Also, these models helped in achieving a good amount of accuracy. Pravin [9] used ResNet models in their research for image recognition. The use of more layers does not increase the accuracy of the model because more number of layers causes the problem of vanishing gradients. This problem can be overcome by using architecture like ResNets in image recognition. Using this architecture helps in restoring the accuracy and also minimizes the effect of vanishing gradients to a good extent. Song et al. [10] used ResNet-18 model in semantic segmentation. The main aim was to improve the pixel-wise semantic segmentation. To achieve this task, deep learning neural models were used. ResNet-18 architecture helped in reducing the number of parameters used in the models. The datasets it worked upon were CamVid and Cityscapes and proved its efficiency.

John et al. prepared a textual question and answer format for the analysis of the efficacy and side effects of vaccines. Based on the responses obtained, information tests are conducted. Participants were randomized by quarterics randomization tool and showed that there is no effect of increasing the vaccinations but by individual perspective [11]. Snigdha et al. prepared a graphical visualization of the analysis of spread of COVID-19 pandemic situation with the help of current pandemic situation using machine learning regression algorithms and obtained satisfying results. However, the models are to be explored to obtain better results with the help of deep learning techniques [12]. Ashwini et al. [13] using LSTM predicted the number of possible cases for the next 10 days.

Majority of the literatures referred are using deep learning algorithms for sentiment analysis and survey-based approach to analyze the COVID vaccine symptoms. In the present work, the analysis of vaccine side effects is given importance.

3 Methodology

Two approaches were used to classify the given images into covid and non-covid images. The approaches used are as follows:

1. ResNet 16
2. Inception v4.

The input image size is $64 \times 64 \times 3$.

Architecture of the inception v4 is shown in Fig. 1.

The architecture of ResNet 16 is shown in Table 1.

In the end, after flattening two dense layers are added. In the last-second dense layer, the Relu Activation function is used. In the last layer, the sigmoid activation function is used. For training a model trained with Adam optimizer, binary cross-entropy loss and used early stopping by monitoring the validation accuracy.

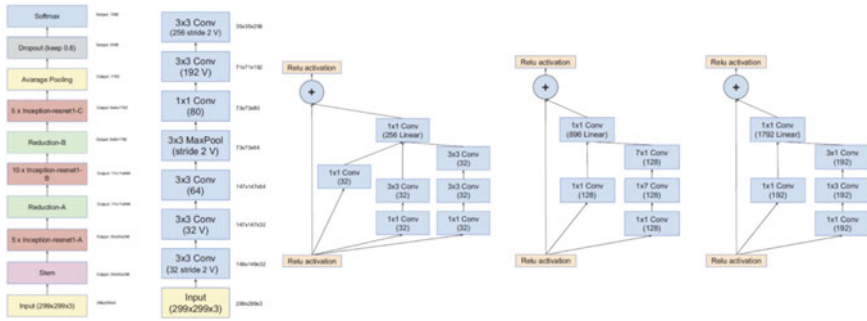


Fig. 1 Architecture of inception v4

Table 1 Architecture of ResNet 16

Layer name	ResNet 16
Conv1	7 × 7, 64, stride 2 3 × 3 max pool, stride 2
conv2_x	3 × 3, 64 3 × 3, 64 3 × 3, 64 3 × 3, 64
conv3_x	3 × 3, 128 3 × 3, 128 3 × 3, 128 3 × 3, 128
conv4_x	3 × 3, 256 3 × 3, 256 3 × 3, 256 3 × 3, 256
conv5_x	3 × 3, 512 3 × 3, 512

4 Result

ResNet 16: Model has trained with 238 images. No. of test images are 39.

Optimizer used is Adam. No. of epochs are 20; validation step = 5.

After training with 20 epochs, the results are shown in Fig. 2.

Training stopped after 20 epochs. From the graphs, we can observe that on increasing no. of epochs training accuracy is increased and loss is decreased. The training accuracy is 98.7%. The validation accuracy is 94.8% (Fig. 3).

Training loss is 0.0719, training accuracy is 0.9643, validation loss is 0.0318, and validation accuracy is 1.0000. The training process achieved reasonable accuracy with the model architecture, and both training and validations losses are very low (Fig. 4).

Fig. 2 Training and validation accuracy

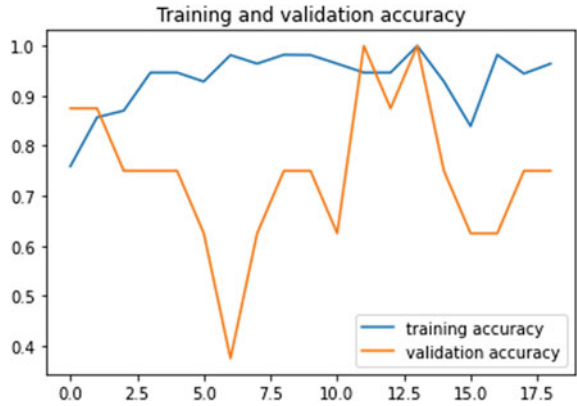
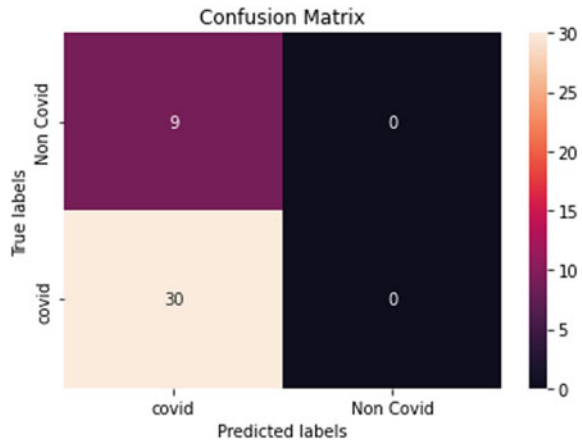


Fig. 3 Training and validation loss



Fig. 4 Confusion matrix for training process



From the confusion matrix, it is observed that Precision is 77%, Recall is 100%, and *F1* score is 87%. On the test data, 87% *F1* score is achieved.

The model predicted covid images correctly but non-covid images were also detected as covid. This can be because of overfitting or less no of sample images. Simplifying the architecture can further give better results.

Inception v4: Model has been trained with 238 images. No. of test images are 39.

Optimizer is Adam, and No. of epochs are 10 (Figs. 5 and 6).

From Fig. 7, it is observed that training loss is 2.7406, training accuracy is 0.9231, validation accuracy is 0.2308, Precision is 77%, Recall is 100%, and *F1* score is 87%.

This model also got 87% *F1* score. It could not be able to detect non-covid images correctly. The model seems to be overfitted.

Fig. 5 Training and validation accuracy

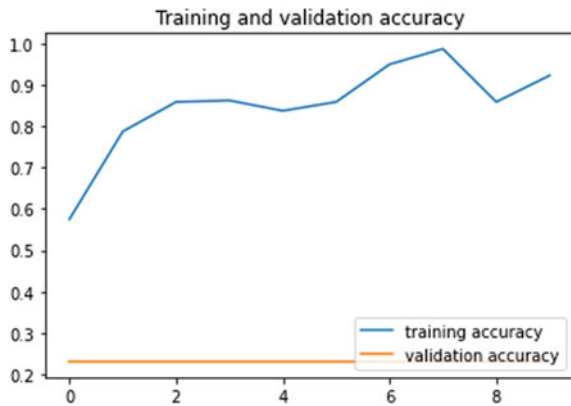


Fig. 6 Training and validation loss

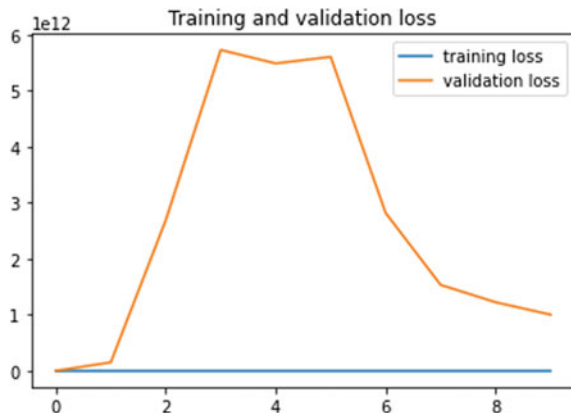
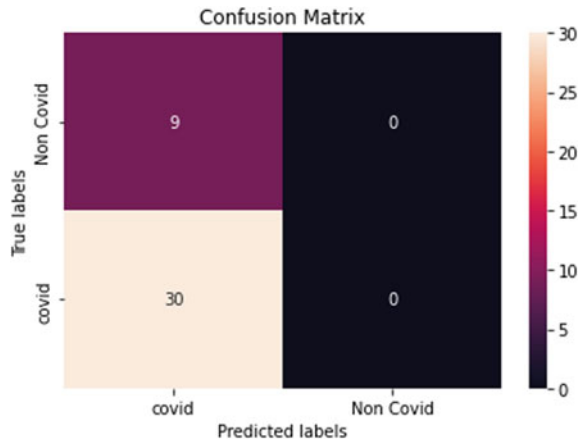


Fig. 7 Confusion matrix for testing data



5 Conclusion

World is facing a large challenge of mitigation of COVID-19 pandemic, by saving lives, by reducing the spread of infectious COVID-19 virus, by channelizing and optimizing the healthcare infrastructure, etc. Efficient methods to detect and predict COVID-19 are very much desired, and it is the need of the hour. In this direction, the proposed methodology not only provides the methodological and conceptional improvements of inception v4 network but also an improved performance. We authors strongly believe that this technology can bring a change in the healthcare management of COVID-19. It can potentially save many lives and can contribute to the mankind.

References

1. Kwekha-Rashid AS, Abduljabbar HN, Alhayani B (2021) Coronavirus disease (COVID-19) cases analysis using machine-learning applications
2. Asraf A, Islam MZ, Haque MR, Islam MM. Deep learning applications to combat novel coronavirus (COVID-19) pandemic
3. Al-Turjman F (2021) Artificial intelligence and machine learning for COVID-19
4. Agrawal T, Choudhary P (2021) FocusCovid: automated COVID-19 detection using deep learning with chest X-ray images
5. Swapnarekha H, Behera HS, Roy D, Das S, Nayak J (2021) Competitive deep learning methods for COVID-19 detection using X-ray images. *J Inst Eng (India) Ser B* 102:1177–1190
6. Al Husaini MAS, Habaebi MH, Gunawan TS, Islam MR, Elsheikh EAA, Suliman FM (2021) Thermal-based early breast cancer detection using inception V3, inception V4, and modified inception MV4. *Neural Comput Appl*
7. Talo M, Yildirim O, Acharya UR (2019) Convolutional neural networks for multi-class brain disease detection using MRI images. *Comput Med Imaging Graph* 78:101673
8. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition
9. Pravin (2021) Computer vision, deep learning

10. Song H, Zhou Y, Jiang Z, Guo X, Yang Z. ResNet with global and local image features, stacked pooling block, for semantic segmentation

Computational Deep Learning Models for Detection of COVID-19 Using Chest X-Ray Images



Srirupa Guha, Ashwini Kodipalli, and Trupthi Rao

1 Introduction

Coronavirus disease (COVID-19) is a highly infectious disease that transformed our day to day lives all over the globe in the twenty-first century. Developments in technology have a swift impact on each and every area of life. It may be the medical domain or any other domain for that matter. Approximately, 250 nations are still suffering from COVID in a short span of time. Indian Government is taking required action to manage the impact of the deadly virus. People across the globe are exposed to its reverberations in the future. People very often worry if they exhibit symptoms of COVID-19 or not [1].

The continuously increasing count of COVID-19 patients could not be handled efficiently because the coronavirus testing happened manually in the initial stage. The coronavirus disease is split into 3 stages, and it has distinct effects on lungs. For tackling this pandemic, artificial intelligence (AI) technologies have been used by researchers under which chest X-ray images and chest CT scan images are taken as data. AI helps in predicting the coronavirus disease, analyzing the structure of the virus whereas chest X-ray and CT scan images help in predicting the stages of corona virus [2].

S. Guha (✉)
Indian Institute of Science, Bangalore, India
e-mail: srirupaguha18@gmail.com

A. Kodipalli · T. Rao
Department of Artificial Intelligence and Data Science, Global Academy of Technology,
Bangalore, India
e-mail: dr.ashwini.k@gat.ac.in

T. Rao
e-mail: trupthirao@gat.ac.in

The substantial impact of this disease is that it is extremely contagious that leads life to a pause situation. But, as soon as people got some data on the virus, research on COVID-19 diagnosis boosted up. At present, the standard diagnosis method of COVID-19 is centered around the swab test, i.e., from the sample collected from nose and throat, which is very much time-consuming and is subjected to man made errors. However, the sensitivity of the swab tests is not good enough for timely detection of the disease [3].

Early detection of positive cases is important to avert further spread of the disease. In the diagnostic phase, radiological images of the chest are determinative as well as the reverse transcription-polymerase chain reaction (RT-PCR) test [4].

Using AI technology to spot the features of COVID-19 in CT images, swiftly screen COVID-19 patients, attain quick diversion and treatment of suspected patients, reduce the infection rate, and control the spread of the disease [5].

This disease has caused about 230,000 deaths all over the world by the end of April 2020. Within a span of six months, it has infected millions of people across the globe because of its high spreading rate. Thus, many countries have put lot of efforts in improving the diagnostic capability of their health care centers/hospitals such that disease could be recognized as early as possible. But, the results of the standard swab test come in a day or two that increases the chance of spreading of the disease due to late diagnosis. Hence, a fast-screening method employing already existing tools such as X-ray and computerized tomography (CT) scans can assist in alleviating the load of mass diagnosis tests. Pneumonia is the first symptom of COVID-19, and chest X-ray is the best technique for diagnosing it [6].

The organization of the paper is as follows: Sect. 2 describes the detailed literature survey. Section 3 shows the methodology and the dataset. Section 4 provides results and the comparative study. Section 5 explains the discussion and future work. Section 6 gives the conclusion of the work.

2 Literature Survey

Othman et al. [7] aimed to give a tool employed for forecasting which determines the COVID-19 cases for seven days. The author employed computational algorithms, namely artificial neural network (ANN), autoregressive integrated moving average (ARIMA), convolutional neural network (CNN), and long short-term memory (LSTM) for predicting COVID-19 cases. This paper mainly aimed to fine-tune each process, and comparisons were done using various performance measures, namely root mean squared logarithmic error (RMSLE), mean absolute percentage error (MAPE), and mean squared logarithmic error (MSLE).

Sevi et al. [8] used X-ray images of COVID-19 patients, and the method was aimed for the binary classification of X-ray images with COVID-19, viral pneumonia, and healthy patients. Data augmentation method was applied on the dataset and performed multi-class classification using deep learning models.

Ghada et al. [9] have detected and identified COVID-19 employing X-ray radiation. For the purpose of comparison, CNN deep learning models used are Inception v3 network, GoogLeNet, ResNet-101, and DAG3Net. Among the various deep learning models employed, DAG3Net outperformed with the accuracies of 96.15%, 94.34%, 96.75%, and 96.58% for validation, training, testing, and overall, respectively, whereas the GoogLeNet, Inception v3 network, and ResNet-101 have produced accuracies of 98.08%, 99.59%, and 100%, respectively.

Mohammad et al. [10] have used some of the AI-based DL models, namely long short-term memory (LSTM), generative adversarial networks (GANs) to provide the user-friendly platform for the detection of COVID-19 by both physicians and researchers.

Milon et al. [11] used X-ray and computer tomography (CT) for detecting COVID-19 by employing DL models. He carried out the detailed survey on the dataset and trained the model using these deep learning models which are developed by the researchers who carried out research in this field. The paper aimed at facilitating experts (medical or non-medical) and technicians in accepting the ways DL techniques are employed in this regard and how they could be employed to combat the COVID-19 outbreak.

Talha et al. [12] have used DL technology for diagnosing COVID-19 through chest CT scan. For early and accurate detection of coronavirus, EfficientNet deep learning architecture is employed, and the following performance measures were achieved: accuracy 0.897, $F1$ -score 0.896, and AUC 0.895. The three distinct learning rate methods employed are as follows: decreasing the learning rate as soon as model performance stops improving (reduce on plateau), cyclic learning rate, and constant learning rate. A $F1$ -score of 0.9, 0.86, and 0.82 was achieved on reduce on plateau, cyclic learning rate, and constant learning rate strategies, respectively.

Samira et al. [13] have proposed CoviNet a DL network which can automatically detect the presence of COVID-19 in chest X-ray images. This architecture is formed on histogram equalization, an adaptive median filter, as well as a CNN. The dataset used for the study is publicly available. This model attained 98.62% and 95.77% accuracies for binary and multi-class classification, respectively. This framework may be employed to aid radiologists in the early diagnosis of COVID-19, as the early diagnosis will limit the spreading rate of the virus.

Loveleen et al. [14] have presented a methodology to detect COVID-19 disease from chest X-rays while differentiating those from normal and affected by viral pneumonia through deep convolution neural networks (DCNNs). Here, three pre-trained CNN models (InceptionV3, VGG16, and EfficientNetB0) are assessed through transfer learning. The principle for the selection of these three specific models is their fairness toward accuracy and efficiency with less parameter appropriate for mobile applications. It is trained on a publicly accessible dataset which is collected from different sources. This study employs DL techniques and the following performance metrics: accuracy, $F1$ -score, precision, recall, and specificity. The results obtained: Accuracy of 92.93% and sensitivity of 94.79% indicate that the proposed technique is a high-quality model.

Table 1 Distribution of the dataset images

Train		Test	
COVID	Non-COVID	COVID	Non-COVID
110	128	30	9

The study portrays a definite chance to implement CV design to validate early and effective and detection as well as screening measures.

3 Methodology

3.1 Dataset Used

The dataset comprises 238 images for training and 39 images for testing, each consisting of two classes—COVID and non-COVID.

The distribution of the two classes is as in Table 1.

3.2 ResNet 16

ResNet16 architecture, as shown in Fig. 1, was used in this research with the following specifications of layers: The train and test images are rescaled before feeding into the model. The size of the input image is $224 \times 224 \times 3$. Zero padding is used along with 2D convolution. Batch normalization is used to normalize the data. For pooling layers, we used max pooling. Activation function used for hidden layers is ReLU. Since it is a binary classification task of classifying as COVID and non-COVID, sigmoid activation function is used at the output layer for binary classification with the batch size 32. There are 3 fully connected (FC) layers at the end of the network:

FC1: 256 units

FC2: 128 units

FC3 (output layer): 1 unit.

The detailed implementation of ResNet 16 is shown in Fig. 2. The same is available here.

The model was trained with 238 images comprising the two classes: COVID and non-COVID and tested on 39 images belonging to either of the two classes: COVID and non-COVID. To make the model robust, variation in training data was introduced by setting shear range = 0.2, zoom range = 0.2, and horizontal flip = True.

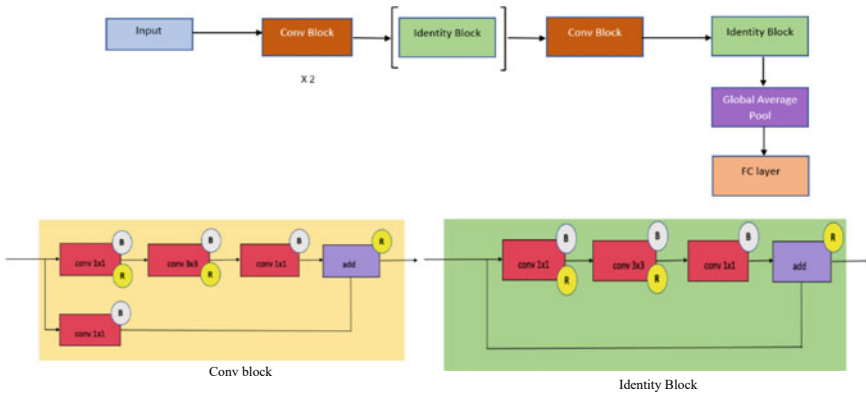


Fig. 1 Components of ResNet16

3.3 Inception V4

Inception v4 architecture is briefly explained as shown in Fig. 3. The train and test images are rescaled before feeding into the model. The dimension of input image is $299 \times 299 \times 3$. Zero padding is used along with 2D convolution. Batch normalization is used to normalize the data. Max pooling is used for pooling layer. ReLU activation function is used for all the hidden layers. Since it is binary classification, sigmoid activation function is used at the output layer, and batch size considered is 8. The model was trained with 238 images comprising the two classes: COVID and non-COVID and tested on 39 images belonging to either of the two classes: COVID and non-COVID. To make the model robust, variation in training data was introduced by setting sheer range = 0.2, zoom range = 0.2, and horizontal flip = True. The diagram showing the detailed implementation of Inception V4 is available here.

4 Results

ResNet 16

Using the following settings of model hyperparameters with the optimizer Adam, loss function binary cross-entropy and steps per epoch as 10, we ran for 100 epochs with validation steps of 5.

At the end of 100 epochs of training, the train and test results are as in Table 2.

From Fig. 4, it is observed that initially the test loss is higher than the train loss on an average for the first 40 epochs, and as the epochs progress, the train and test loss become closer. Initially, the train accuracy is higher than the test accuracy on an average for the first 40 epochs, and as the epochs progress, the train and test accuracies become close to 90%.

Fig. 2 Details of the implementation of layers of the ResNet16 architecture

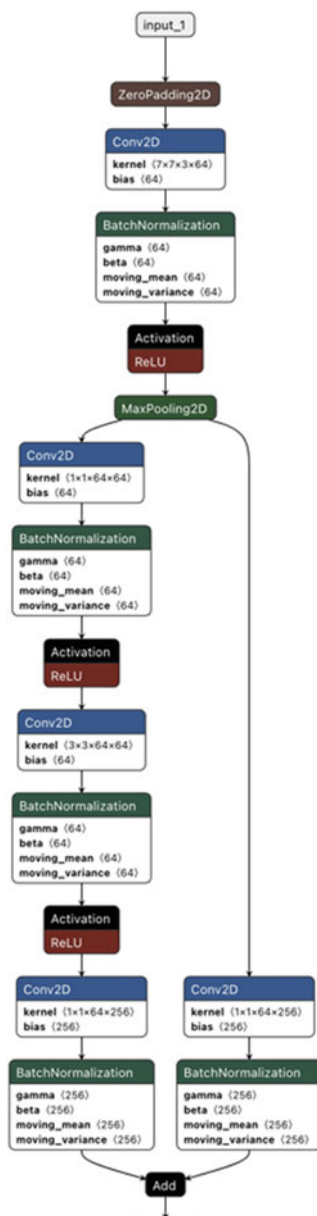


Fig. 2 (continued)

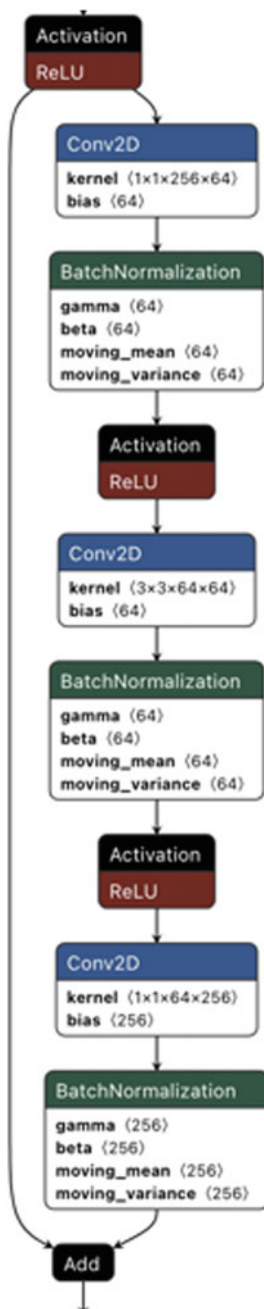


Fig. 2 (continued)

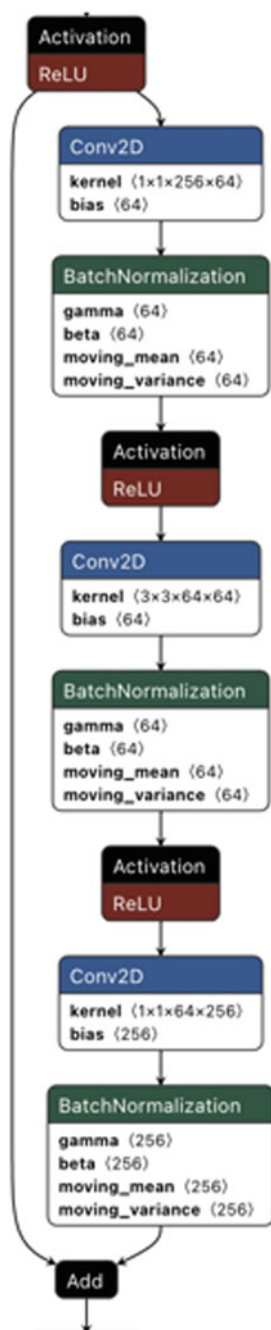


Fig. 2 (continued)

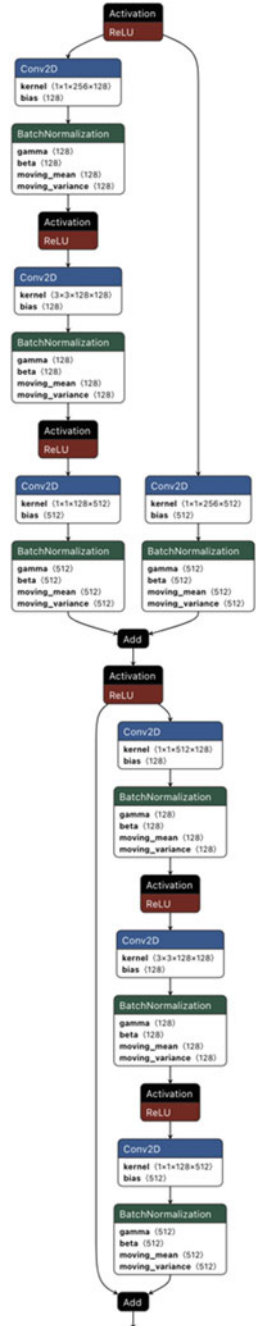
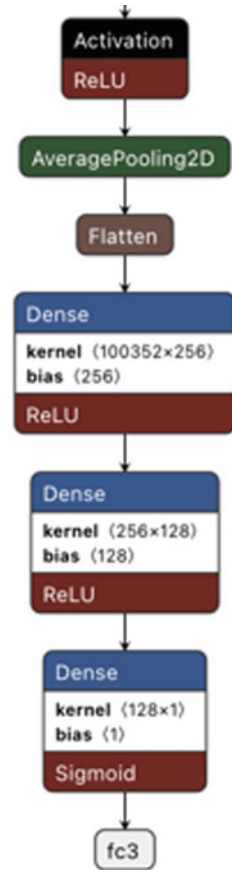


Fig. 2 (continued)



Post-completion of training, other model evaluation metrics were generated on the test set (considering COVID as the positive class and non-COVID as the negative class) as shown in Table 3 (Fig. 5).

From the results as shown in Table 3, it is observed that a precision of 0.967 indicates that 96.7% of the COVID predictions by the model were correct. Thus, in 96.7% of the predicted COVID cases, the model has accurately identified a COVID image. A recall of 1 indicates that 100% or all the actual COVID images were identified correctly by the model. An *F1*-score of 0.983 is indicative of good model performance since there is slight class imbalance in the dataset. The model exhibits specificity of 0.889 indicating that 88.9% of the actual non-COVID images were classified correctly by the model. A false-positive rate of 0.111 indicates that the model tends to raise a false alarm in 11.1% of the cases: that is, predict as COVID even when the actual image might be non-COVID. A training accuracy of 100% indicates that the model has low bias, that is, it has learnt the training data well. The

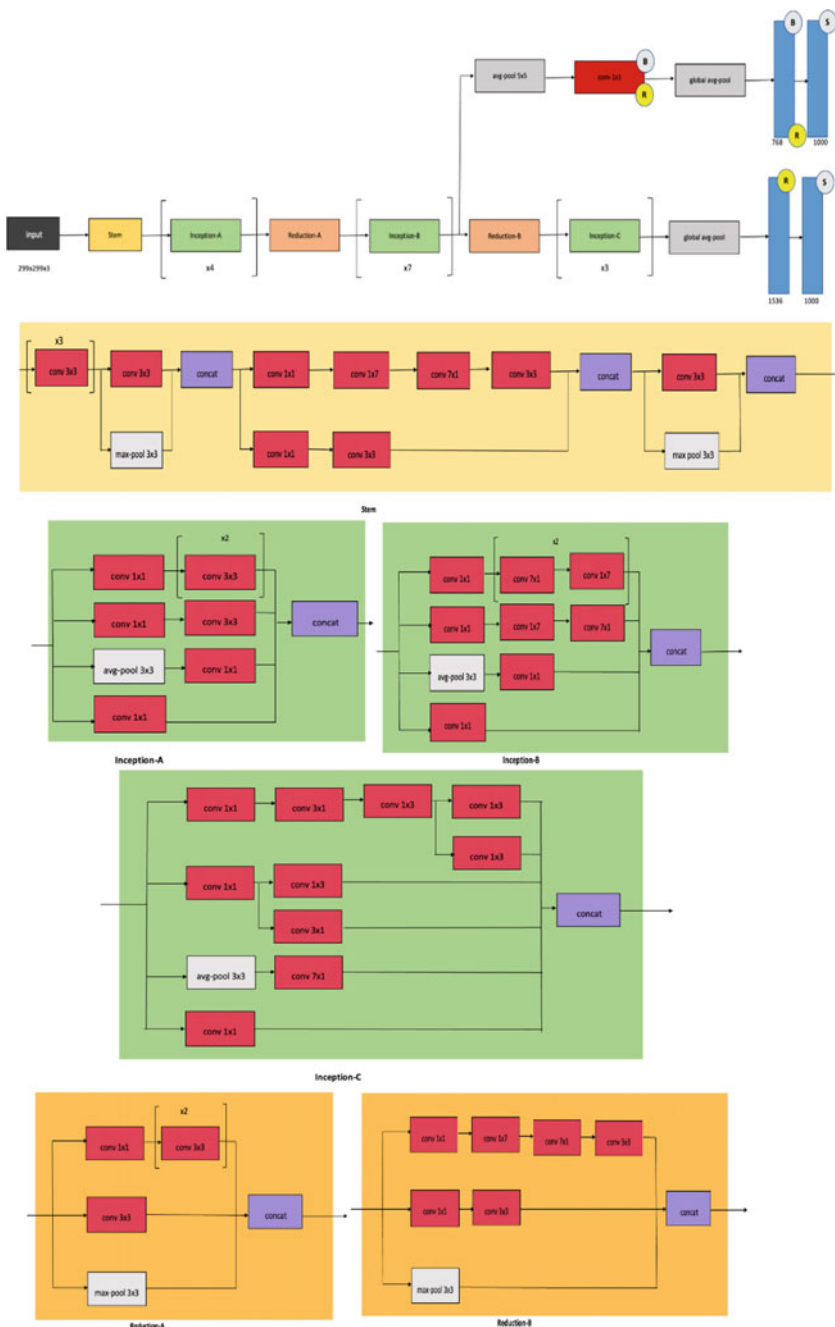


Fig. 3 Architecture diagram of Inception V4

Table 2 Train and test accuracy for ResNet 16

Training accuracy	Training loss	Test accuracy	Test loss
100%	7.8371e-04	97.44%	0.1162

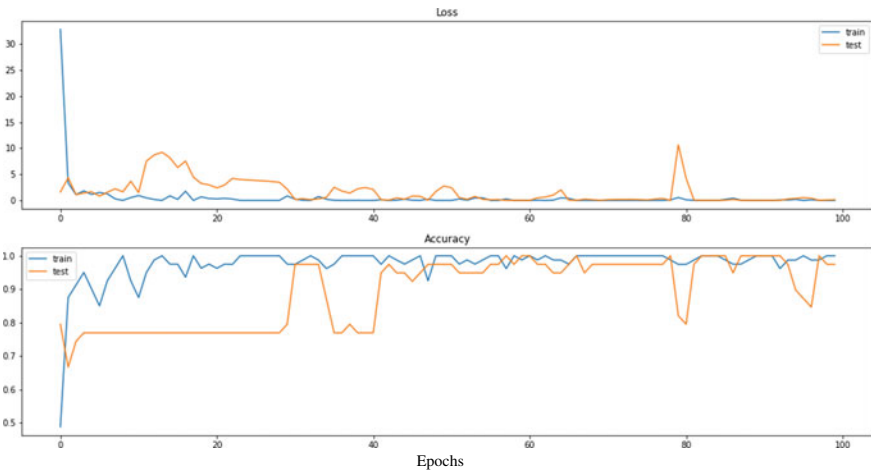


Fig. 4 Plots of train loss versus test loss and train accuracy versus test accuracy for various epochs of ResNet16

Table 3 Evaluation metric results on test set

Metric	Test set results
Precision	0.967
Recall/sensitivity	1
F1 score	0.983

Fig. 5 Confusion matrix for ResNet16 on test set

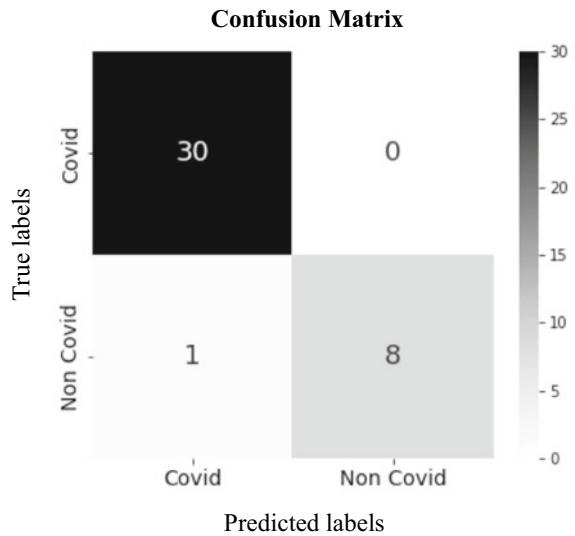


Table 4 Train and test accuracy for Inception V4

Training accuracy	Training loss	Test accuracy	Test loss
100%	0.0052	76.92%	0.6172

test accuracy of 97.44% indicates that the model is also able to generalize well on unseen data.

Inception V4

Using the following settings of model hyperparameters with the optimizer Adam, loss function binary cross-entropy and steps per epoch as 10, we ran for 100 epochs with the validation steps of 5. At the end of 100 epochs of training, the train and test results are as shown in Table 4.

From Fig. 6 graphs, we see that the training loss is almost constant throughout the epochs and is around 0, while the test loss increases till the 7th epoch, sharply declines thereafter, and becomes constant after the 20th epoch. The training accuracy increases sharply and becomes roughly constant after the 20th epoch, while the test accuracy fluctuates and reaches a constant value of 76.92% after 100 epochs. Post-completion of training, other model evaluation metrics were generated on the test set (considering COVID as the positive class and non-COVID as the negative class) as shown in Table 5 (Fig. 7).

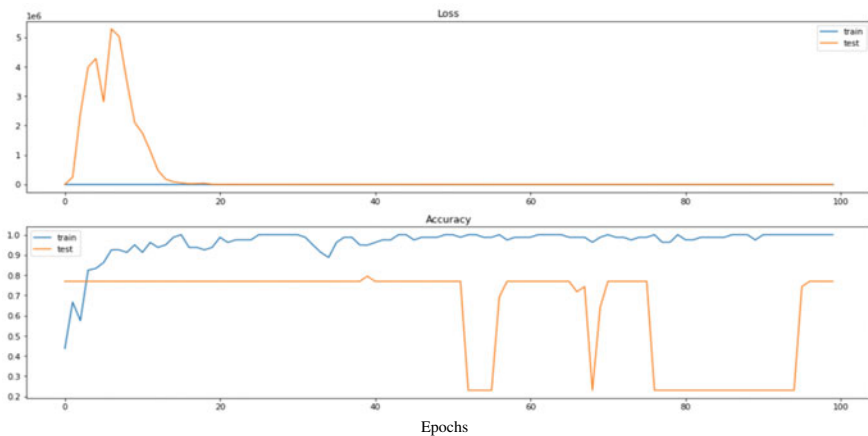
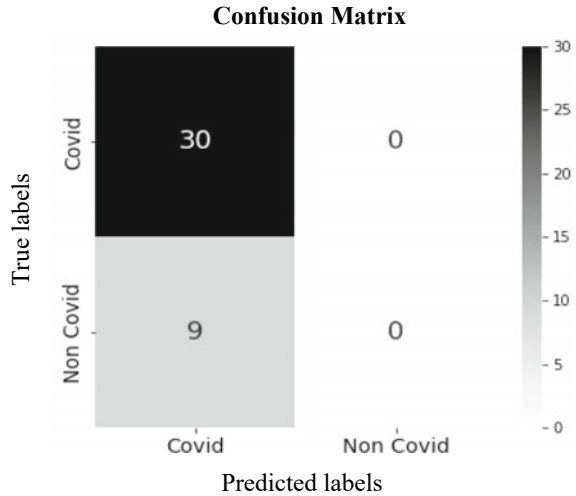


Fig. 6 Plots of train loss versus test loss and train accuracy versus test accuracy for various epochs of Inception V4

Table 5 Evaluation metric results

Metric	Test set results
Precision	0.77
Recall/sensitivity	1
F1 score	0.87

Fig. 7 Confusion matrix for Inception v4 on test set



From the results, it is observed that a precision of 0.77 indicates that 77% of the COVID predictions by the model was correct. Thus, in 77% of the predicted COVID cases, the model has accurately identified a COVID image. A recall of 1 indicates that 100% or all the actual COVID images were identified correctly by the model. An $F1$ -score of 0.87 is indicative of good model performance since there is slight class imbalance in the dataset. The model yields specificity of 0, indicating that none of the actual non-COVID images were classified correctly by the model. Thus, the model tends to predict positive (COVID) most of the times. A false-positive rate of 1 indicates that the model tends to raise a false alarm: that is, predict as COVID even when the actual image might be non-COVID. A training accuracy of 100% indicates that the model has low bias, that is, it has learnt the training data well. However, compared to this, the test accuracy of 76.92% indicates that the model is not able to generalize well. Hence, the model has slightly overfitted the data. This overfitting problem can be solved by training the model with more images of varying characteristics.

When the comparison is made, it is observed that ResNet16 performs better on the COVID dataset. It has fitted the data well and is also able to generalize well. However, due to class imbalance in the dataset, there is a tendency to classify an image as COVID in some cases by both models.

5 Discussion

The ResNet16 certainly is the more accurate classifier than Inception v4. Inception v4 suffers from overfitting problem due to the relatively more complex architecture and is unable to generalize well. However, some of the labeled COVID images in the

training dataset exhibit similar characteristics as non-COVID images, and hence, this affects the precision and recall performance of the classifiers. ResNet16 performs better than Inception V4 on the test set with the latter exhibiting a tendency to predict COVID positive in most of the cases. The ResNet16 shows improvement in performance after 40 epochs, while for Inception v4, the test accuracy continues oscillating with an upper bound of 76.92%. The computational complexity combined with overfitting problem makes the ResNet16 architecture better suited for this use case.

6 Conclusion

We presented two architectures for detection of COVID-19 from chest X-ray images in this paper:

- i. ResNet16 with 16 convolutional layers and 3 fully connected layers, which is a custom-made architecture for this use case.
- ii. Inception v4 that builds on previous iterations of the Inception family by simplifying the architecture and using more inception modules than Inception v3.

Our results indicate that ResNet16 performed better on the test set after few epochs, while the Inception v4 owing to its complexity overfitted the data and was unable to generalize well on the test set. We saw that Inception V4 is inclined to predict COVID positive in most of the cases. On investigating the cause of false positives, it was found that some images which exhibited the characteristics of non-COVID were labeled as COVID. This has likely impacted the true positive and true negative predictive performance of Inception V4 and owing to the overfitting problem, caused the model to learn those image characteristics as COVID. This problem can be overcome by applying regularization to reduce the complexity of the Inception V4 model and training on more samples.

References

1. Rohini M, Naveena KR, Jothipriya G, Kameshwaran S, Jagadeeswari M (2021) Proceedings of the international conference on artificial intelligence and smart systems (ICAIS-2021). IEEE Xplore part number: CFP21OAB-ART. ISBN: 978-1-7281-9537-7
2. Sharma S, Tiwari S (2021) COVID-19 diagnosis using X-ray images and deep learning. In: Proceedings of the international conference on artificial intelligence and smart systems (ICAIS-2021). IEEE Xplore part number: CFP21OAB-ART. ISBN: 978-1-7281-9537-7
3. Liu J, Zu L, Zhong Y, Zhang Z, Wang H (2020) Intelligent detection for CT image of COVID-19 using deep learning. In: 2020 13th international congress on image and signal processing, biomedical engineering and informatics (CISP-BMEI)

4. Istaiteh O, Owais T, Al-Madi N, Abu-Soud S (2020) Machine learning approaches for COVID-19 forecasting. In: International conference on intelligent data science technologies and applications (IDSTA)
5. Sevi M, Aydin İ (2020) COVID-19 detection using deep learning methods. In: 2020 international conference on data analytics for business and industry: way towards a sustainable economy (ICDABI)
6. Ghada A, Abdullah A, Nahedh H (2021) COVID-19 detection using deep learning models. In: 1st Babylon international conference on information technology and science 2021 (BICITS 2021), Babil, Iraq
7. Anwar T, Zakir S (2020) Deep learning based diagnosis of COVID-19 using chest CT-scan images. In: 2020 IEEE 23rd international multitopic conference (INMIC). 978-1-7281-9893-4/20/\$31.00 ©2020 IEEE. <https://doi.org/10.1109/INMIC50486.2020.9318212>
8. Lafraxo S, El Ansari M (2020) CoviNet: automated COVID-19 detection from X-rays using deep learning techniques. In: 2020 6th IEEE congress on information science and technology (CiSt). 978-1-7281-6646-9/21/\$31.00 ©2021 IEEE. <https://doi.org/10.1109/CIST49399.2021.9357250>
9. Zheng C, Deng X, Fu Q, Zhou Q, Feng J, Ma H et al (2020) Deep learning based detection for COVID-19 from chest CT using weak label. medRxiv
10. Zhang J, Xie Y, Li Y, Shen C, Xia Y (2020) Covid-19 screening on chest X-ray images using deep learning based anomaly detection. arXiv preprint [arXiv:2003.12338](https://arxiv.org/abs/2003.12338)
11. Wang Y, Hu M, Li Q, Zhang XP, Zhai G, Yao N (2020) Abnormal respiratory patterns classifier may contribute to a largescale screening of people infected with COVID-19 in an accurate and unobtrusive manner. arXiv preprint [arXiv:2002.05534](https://arxiv.org/abs/2002.05534)
12. Wang X, Deng X, Fu Q, Zhou Q, Feng J, Ma H et al (2020) A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT. IEEE Trans Med Imaging
13. Hu S, Hoffman EA, Reinhardt JM (2001) Automatic lung segmentation for accurate quantitation of volumetric X-ray CT images. IEEE Trans Med Imaging 20(6):490–498
14. Dong L, Hu S, Gao J (2019) Discovering drugs to treat coronavirus disease 2019 (COVID-19). Drug Discov Ther 14(1):58–60
15. Springenberg JT, Dosovitskiy A, Brox T, Riedmiller M (2014) Striving for simplicity: the all convolutional net. arXiv preprint [arXiv:1412.6806](https://arxiv.org/abs/1412.6806)

An Ensemble Approach for Detecting Malaria Using Classification Algorithms



S. Ruban , A. Naresh, and Sanjeev Rai

1 Introduction

Malaria is estimated to cause one million deaths annually according to the WHO report [1]. Though there are various reasons that may be contributing toward this life threatening disease such as climatic conditions, underdeveloped sanitation, and deforestation [2], it is mainly due to the infection caused by the bite of a female mosquito. Getting fever, headache, and other symptoms reveal the presence of the parasite in the patient's body. In our country, malaria continues to cause a major threat in the public health. Though this is a treatable disease and medications are available, it is important to understand the type of parasite that is found in the patient's body. Thick blood smear test is used to detect whether the malaria parasite is present, and the other blood test done with thin blood smear is helpful to find out the species of malaria parasite that is the reason for the infection [3]. However, the precision of these tests depends upon the quality of the smear that is extracted and also on the classification of the infected cells and others. Over the period of time, there were few more alternative methods that were suggested for diagnosing malaria such as polymerase chain reaction which is referred to as PCR test and rapid diagnostic tests for malaria commonly referred to as RDT. However, these alternate tests have been found to have some issues related to its performance and are less cost-effective [4, 5].

Technology can assist in the detection of the malaria parasite, thereby helping to manage the treatment more accurately and ensuring a timely service. With significant improvement possible over the current diagnosis, this research intends to study the detection of malarial parasites by using machine learning algorithms over the health

S. Ruban (✉) · A. Naresh
St Aloysius College (Autonomous), Mangalore, India
e-mail: ruban@staloyusius.ac.in

S. Rai
Father Muller Medical College, Mangalore, India

data, depending on the ability to extract feature from the dataset. Machine learning is used in medical diagnosis and intelligent systems. It is about developing machines with the ability to be intelligent. This intelligence is imputed into the application by the data it is trained with. With many advances that is going on with respect to the data collection, data processing, and data computation, intelligent AI systems are used in multiple tasks that were once accomplished by manual intervention. Autonomous vehicles to health care [6], the situations are changing so fast, in a way that could not be comprehended. However, the challenge in developing these AI-based applications is the availability of data. There are plenty of data that are available in different silos of the healthcare organizations. However, most of these available data are written by Hand [7]. Poor handwritten clinical notes too, causes a severe challenge for the data scientists who are involved in analyzing the data.

Machine learning algorithms have been used by researchers to predict malaria like other infectious diseases. One of the sub-domains of machine learning called deep learning has become very popular these days. The authors Lee et al. [8] in their work on malaria detection used back-propagation neural networks, one of the commonly used method in deep learning to model the climatic factors and other attributes related to malaria.

Another familiar one called as long short-term memory network (LSTM) was used for prediction by the researchers [9]. Chae et al. [10] in their work found that the performance of the prediction could be increased using the LSTM. All these works that have been quoted above tell about using deep learning methods to develop models for predicting infectious diseases. However, by just using one algorithm, the accuracy may be limited [11]. Hence, an innovative method called stacked generalization has got more attention recent days, where stacking frameworks with different machine learning algorithms can be applied together to increase the prediction performance. In another work, Bhatt et al. [12] came out with a model that displayed a better prediction performance than the other models in predicting malaria prevalence. In this work, we propose a multiclassifier using different classifiers, namely XGBoost classifier, random forest (RF), and gradient boosting.

2 Materials and Methods

This part of the paper discusses about, the data sources from where the real-time data were collected, format of malaria data, data gathering process, data preprocessing, data processing, and ensemble classifiers.

2.1 Data Sources

As one of the coastal cities, Mangalore is named as one of the malaria-prone city [13] in south India. Most of the works on malaria that were done in Indian scenario

were epidemiological assessment based on the demographic data. So far, no research has been conducted using case records from malaria patients who were admitted and treated. The case sheets were accessed in this study with the approval of the scientific and ethical committees.

2.2 Malaria Information

Person treated from malaria exhibit various symptoms like fever and flu-like sicknesses like chills, headache, muscle ache, and tiredness. Mostly, people suffer from fever and chills. If not identified at the right time, malaria may lead to anemia and jaundice, which may lead to kidney failures, seizures, mental confusion, coma, and death [14]. These symptoms normally last from 2 days to a week. The outpatient registration and medical records departments had all the necessary information for building a model and the data were arranged according to ICD code [15].

2.3 Medical Records Department Data Format

After obtaining the necessary approval from the Father Muller Medical College's scientific and ethical council, the clinical notes linked to malaria from Father Muller Medical College were retrieved. Malaria-related information was kept in the form of electronic medical records (EMRs). The case sheets and clinical notes were scanned and archived in the Medical Records Department's repository. As a result, the scanned photos would not be used for data analysis. They must be converted into a format that the machine learning algorithms can understand. The patient's clinical notes were retrieved using the inpatient number, which serves as a unique identifier for the data in the Medical Records Department and the data in the Registration Department.

2.4 Preprocessing

The original data gathered from the clinical notes were unprocessed, and most of the portion were handwritten. Most of these handwritten portion of the clinical notes were written by various doctors and nurses who were attending to the patients who were undergoing treatment. Only, the discharge summary which was part of the clinical notes is the typed one. The quality of data plays an important role in the machine learning project. The missing entries, inconsistencies, typographical, and semantic errors that were there in the raw data were clarified and rectified based on the discussion with the healthcare professionals who were assigned for that. This step does not give you any meaningful insight. However, it helps to find out the right assumption to be made for the analysis and the features that have to be extracted. We

did use the Tesseract optical character recognition engine (OCR) [16] for extracting the raw data from the clinical notes. However, the accuracy of data that were extracted was moderate depending upon the clarity of the images, blur and noise. So, the data that were extracted had to be manually checked by the healthcare professionals. It was followed by pattern identification. Preprocessing of data was done in four stages.

Cleansing the Data: This step was done with the help of the healthcare professionals from the medical college hospital, who helped in identifying the errors that have crept into the data.

Integrating the Data: The data that were available in two places were captured and integrated. The medical records section provided the clinical data. The IP No, which serves as a main key, was then used to combine both slices of data pertaining to a patient.

Transforming the Data: This crucial step takes care of transforming the data from its raw format to a format which is computable. This transformation also makes sure the original intended meaning of the data is not lost.

Dimensionality Reduction: This step is important with reference to the processing of the application. This step ensures that, no repeated data are available, and the data that are not relevant to the analysis are also pruned.

2.5 *Data Processing*

Based on the discussions with the physicians, a data dictionary was created, which acts as a metadata for the efficient and smooth processing of the data as shown in Table 1.

2.6 *Ensemble-Based Classification Algorithms*

Machine learning techniques are basically algorithms that try to find out the relationship between different features that are found in the dataset. Most of the machine learning techniques are classified into supervised and unsupervised learning. Supervised learning refers to a set of methods that are trained on a set of factors mapped on a target label. A machine learning model that produces discrete categories [17] can be called as classification. Few case studies to quote include, those which can predict whether a person has malaria or not. Predicting whether a tumor is malignant or benign. In medicine, such kind of classification problems do exist, and classification algorithms are used in those areas. There are many classification algorithms that are available. In this research work, we have used few algorithms such as random forest algorithm, gradient boost algorithm, and XGBoost algorithm. XGBoost is a classification algorithm that is quite popular these days since it is used in many machine

Table 1 Features in the data dictionary related to malaria

<i>Plasmodium falciparum</i>	Mixed malaria	<i>Plasmodium vivax</i>
<i>Plasmodium ovale</i>	Fever	Headache
Body ache	Vomiting	Chills
Cough	Joint pain	Abdominal discomfort
Breathlessness	Loose stool	Puffiness in the face
Running nose	Throat pain	Bleeding
Hypertension	Nausea	Urinal variation
TB	Asthma	Pallor (anemia)
Epilepsy	Drowsy	Appetite
Sleep	Skin rashes	Age
Gender	BP	Pulse
Respiratory rate	Temperature	Platelet

learning and Kaggle competitions [18] that deal with structured data. This algorithm is mainly designed for faster response and good performance. Gradient boosting is one of the widely used classification algorithm [19]. Random forest classification algorithm is a supervised machine learning algorithm [20] that is based on decision trees. It is also used in many practical applications in finance and health care. All the above algorithms, bagging, boosting, and random forest [21] are normally referred to as ensemble learning algorithms. Our earlier work in this area has been done using the normal machine learning algorithms [22, 23]. Ensemble methods are machine learning methods that incorporate many base models [24] to provide the best optimal result.

3 Results and Discussion

The analysis that was done on the data collected revealed few important insights that are displayed in the following Figs. 1, 2, 3 and 4.

The raw data which were captured from the patient’s case sheets are now transformed into a format that is suitable for creating a simple model. The model can perform any functions based on its necessity. Since our task is related to classification, few algorithms that are known to perform better are used in this research work (Figs. 5 and 6).

The ensemble methods give better results than other traditional algorithms, since they use different base methods within them. The model was trained using the XGBooster classifier, gradient booster classification algorithm, and random forest classification algorithm. The different performance measures were compared with one another, and the metrics are displayed (Table 2).

Fig. 1 Patients tested with malaria

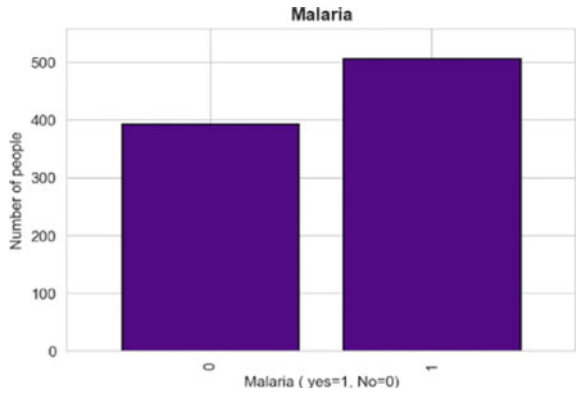


Fig. 2 Patients who had fever and its severity

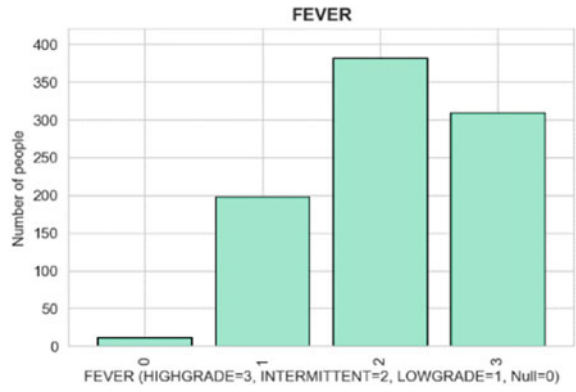


Fig. 3 Patients who were suffering with chills

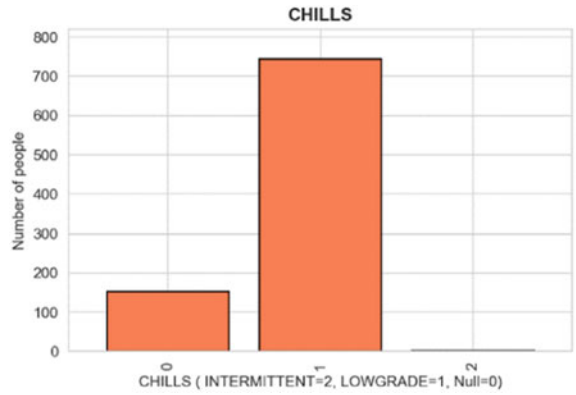


Fig. 4 Patients who were suffering with body ache

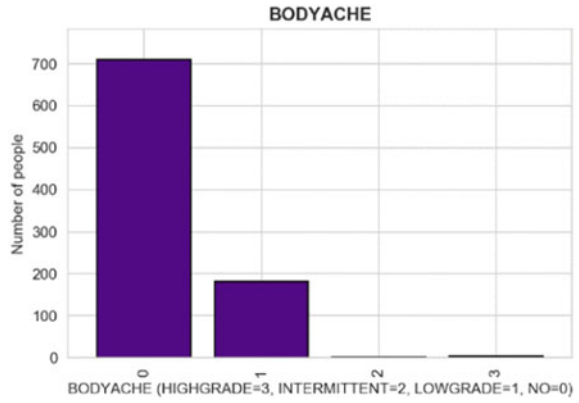


Fig. 5 Patients related to their age

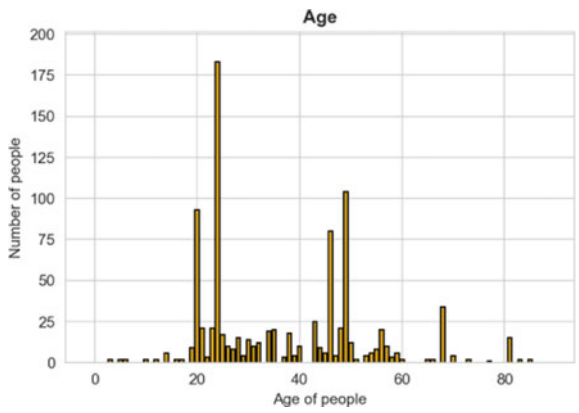


Fig. 6 Patients who were suffering having discomfort in their abdomen

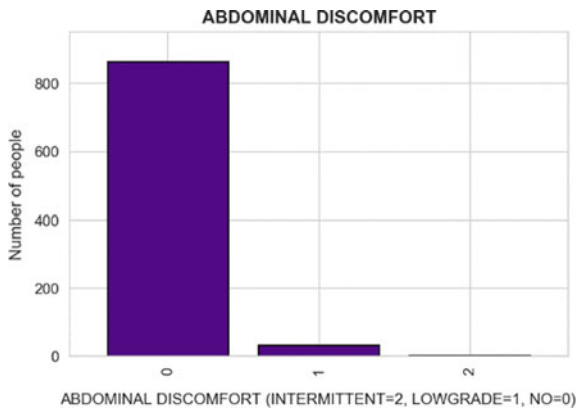


Table 2 Comparison of the ensemble algorithms with respect to the performance metrics

Algorithm	Precision	Recall	F1-score	Accuracy
Random forest classifier	0.77	0.71	0.74	90.97
XGBooster classifier	0.73	0.76	0.75	82.08
Gradient booster classifier	0.70	0.56	0.63	68.47

4 Conclusion

This study, which was based on a malaria patient's clinical records, offers insight on the types of symptoms that patients encounter before being brought to the hospital. It also investigates the efficacy of malaria diagnostic therapy. The faster a doctor checks a patient based on their symptoms, the more likely the treatment will be beneficial. This study was based on information from a single source. More data from different hospital settings and locations could help the system function better. Other traditional methods have been demonstrated to be less effective than ensemble approaches. It is crucial to determine which ensemble method is the most effective. In this investigation, the random forest classifier has an accuracy of 90.97 when compared to other techniques. The same approaches employed in the preparation phases can be utilized to gather data and transform raw clinical data into valuable data in any medical setting.

Acknowledgments Authors acknowledge, that this work was carried out in the Big Data Analytics Lab funded by VGST, Govt. of Karnataka, under K-FIST(L2)-545, and the data were collected from Father Muller Medical College, protocol no: 126/19 (FMMMCIEC/CCM/149/2019).

References

1. World Health Organization (WHO) Malaria report. Available at <https://www.who.int/malaria/publications/world-malaria-report-2019/en/>. Accessed 2 Sept 2020
2. Babagana et al (2017) Towards a predictive analytics-based intelligent malaria outbreak warning system. *Appl Sci* 7: 836. <https://doi.org/10.3390/app7080836>
3. Centers for Disease Control and Prevention. CDC_Malaria. Available at <http://www.cdc.gov/malaria/about/biology>. Accessed 16 Aug 2020
4. Hommelsheim CM, Frantzeskakis L, Huang M, Ülker B (2014) PCR amplification of repetitive DNA: a limitation to genome editing technologies and many other applications. *Sci Rep* 4:5052. <https://doi.org/10.1038/srep05052>
5. Hawkes M, Katsuva J, Masumbuko C (2009) Use and limitations of malaria rapid diagnostic testing by community health workers in war-torn Democratic Republic of Congo. *Malar J* 8(1):308. <https://doi.org/10.1186/1475-2875-8-308>
6. Weng J, McClelland J, Pentland A, Sporns O, Stockman I, Sur M et al (2020) Autonomous mental development by robots and animals. *Science* 291(5504):599–600
7. Rodriguez-Vera FJ, Marin Y, Sanchez A, Borrachero C, Pujol E (2002) Illegible handwriting in medical records. *J R Soc Med* 95:545–546

8. Lee KY, Chung N, Hwang S (2016) Application of an artificial neural network (ANN) model for predicting mosquito abundances in urban areas. *Ecol Inform* 172–180
9. Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: continual prediction with LSTM. *Neural Comput* 12(10):2451–2471. <https://doi.org/10.1162/089976600300015015>
10. Chae S, Kwon S, Lee D (2018) Predicting infectious disease using deep learning and big data. *Int J Environ Res Public Health* 5(8):1596
11. Wang M, Wang H, Wang J, Liu H, Lu R, Duan T et al (2019) A novel model for malaria prediction based on ensemble algorithms. *PLoS ONE* 14(12). <https://doi.org/10.1371/journal.pone.0226910>
12. Bhatt S, Cameron E, Flaxman SR, Weiss DJ, Smith DL, Gething PW (2017) Improved prediction accuracy for disease risk mapping using Gaussian process stacked generalization. *J R Soc Interface* 14(134)
13. Shivakumar, Rajesh BV, Kumar A, Achari M, Deepa S, Vyas N (2015) Malarial trend in Dakshina Kannada, Karnataka: an epidemiological assessment from 2004 to 2013. *Indian J Health Sci* 8:91–94
14. Symptoms of malaria. <https://www.cdc.gov/malaria/about/faqs.html>. Accessed 26 June 2021
15. ICD code of malaria. <https://www.icd10data.com/ICD10CM/Codes/A00-B99/B50-B64/B54-B54>. Accessed 1 July 2021
16. Smith R (2007) An overview of the Tesseract OCR engine. In: Proceedings of ninth international conference on document analysis and recognition (ICDAR). IEEE Computer Society, pp 629–633
17. Sidey-Gibbons JAM, Sidey-Gibbons CJ (2019) Machine learning in medicine: a practical introduction. *BMC Med Res Methodol* 19:64. <https://doi.org/10.1186/s12874-019-0681-4>
18. XGBOOST machine learning algorithm. <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>. Accessed 2 July 2021
19. Gradient boost machine learning algorithm. <https://towardsdatascience.com/machine-learning-part-18-boosting-algorithms-gradient-boosting-in-python-ef5ae6965be4>. Accessed 2 July 2021
20. Random forest machine learning algorithm. <https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/>. Accessed 2 July 2021
21. Lohumi P, Garg S, Singh TP, Gopal M (2020) Ensemble learning classification for medical diagnosis. In: 5th international conference on computing, communication and security (ICCCS), pp 1–5. <https://doi.org/10.1109/ICCCS49678.2020.9277277>
22. Ruban S, Rai S (2021) Enabling data to develop an AI-based application for detecting malaria and dengue. In: Tanwar P, Kumar P, Rawat S, Mohammadian M, Ahmad S (eds) Computational intelligence and predictive analysis for medical science: a pragmatic approach. De Gruyter, Berlin, Boston, pp 115–138. <https://doi.org/10.1515/9783110715279-006>
23. Ruban S, Naresh A, Rai S (2021) A noninvasive model to detect malaria based on symptoms using machine learning. In: Advances in parallel computing technologies and applications. IOS Press, pp 23–30
24. Ensemble methods. <https://towardsdatascience.com/ensemble-methods-in-machine-learning-what-are-they-and-why-use-them-68ec3f9fef5f>. Accessed 10 July 2021

IoT-Enabled Intelligent Home Using Google Assistant



Balarama Krishna Veeramalla, S. Aruna, and K. Srinivasa Naik

1 Introduction

Home is the place where one wishes to feel comfort after working whole day. Some have exhaustive working hours, and in such time, if a device or a technology, that helps switch lights on/off or play favorite music, controlling geyser and adjusting the room temperature were already done before reaching home just by giving simple voice commands on smart phone making life pleasant [1–8]. Housekeepers, a way for rich people to keep up their homes in tact with ease, however, even after technology advancements only the rich people's houses are equipped with latest smart home devices, as they cost high. Hence, realizing low-cost smart device for home automation for the normal families is the need of the hour [10–13].

This paper proposes an affordable system which uses ESP 8266 Node MCU IC and relay board of 4 relays as major hardware elements and Google Assistant, IFTTT and BLYNK applications as major software components. All the elements are interconnected over Internet using Wi-Fi which puts this system under Internet of things (IoT) [14–17] (Fig. 1).

B. K. Veeramalla (✉) · S. Aruna
Department of ECE, Andhra University, Visakhapatnam, India
e-mail: balavenkata3@gmail.com

S. Aruna
e-mail: dr.saruna@andhrauniversity.edu.in

K. Srinivasa Naik
Vignan's Institute of Information Technology, Duvvada, Visakhapatnam, India

Fig. 1 Intelligent home model

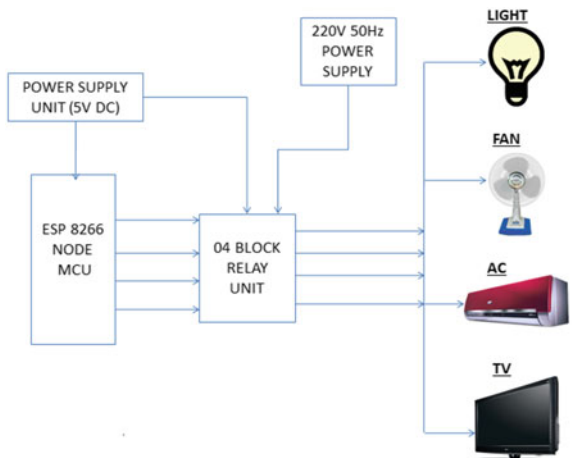


2 System Architecture

The architecture explains the home automation system. The voice commands given by the user are sent through the Google Assistant application which is the front facia of our project. In the back drop primarily, BLYNK application takes the command of the hardware circuit designed as shown in Fig. 2, and the BLYNK is an IoT application which gets connect with ESP 8266 Node MCU on Wi-Fi through hotspot. When the voice command is given on smart phone, the particular node of MCU at which concerned relay gets actuated, and home appliance starts function as per the voice command. This is how the system function using Internet of things (IoT). In our project, we are controlling 04 home appliances by using 04 relays [18–20].

The system hardware mainly comprises of ESP 8266 Node MCU, 04 no. relays, 5 V DC source, 220 V 50 Hz domestic supply, 04 no. home appliances, a smart phone, and Wi-Fi connectivity.

Fig. 2 Block diagram of proposed system



3 Design Implementation

Initially, we design code for functioning of the proposed project by using Arduino software. The code is written in the Arduino application and dumped into ESP 8266 Node MCU board. Below mentioned is the code uploaded into the board (Fig. 3).

The Arduino application software helps to write the software to give command to the BLYNK application, thereby the BLYNK application in the mobile smart phone takes full control over the circuit and acts as the heart of the project in communicating the commands from Wi-Fi module to the home electrical equipments.

We cannot directly communicate Google Assistant with BLYNK, therefore we use IF This Then That (IFTTT) application software as an intermediate interfacing tool between Google Assistant and BLYNK.

In order to activate Google Assistant, we start with “OK GOOGLE” command. Once the Google Assistance is activated, the commands which are used to control the Light, TV, and Fan are given one after the other. For each command given as input, the respective output system will be controlled (Fig. 4).

Flowchart: A simple flowchart defines the operation of home automation model which is given in Fig. 5.

Operation: In **Case 1**, we consider performing ON and OFF operation of television. Now, we provide a command “Switch ON the TV” to Google Assistant. With the help of Wi-Fi module, TV will get switched ON. “Switch OFF the TV”: This command

Fig. 3
ESP8266_Standalone |
Arduino 1.8.15 code

```

ESP8266_Standalone | Arduino 1.8.15
File Edit Sketch Tools Help

ESP8266_Standalone

/* Fill-in your Template ID (only if using Blynk.Cloud) */
#define BLYNK_TEMPLATE_ID "YourTemplateID"

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "Rwg8_I Ea2dBDIXgU1RkKyMA6BpmkVWV";

// Your Wi-Fi credentials.
// Set password to "" for open networks.
char ssid[] = "jishnu";
char pass[] = "vtp@2012";

void setup()
{
  // Debug console
  Serial.begin(9600);

  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Blynk.run();
}

```

Fig. 4 Design work

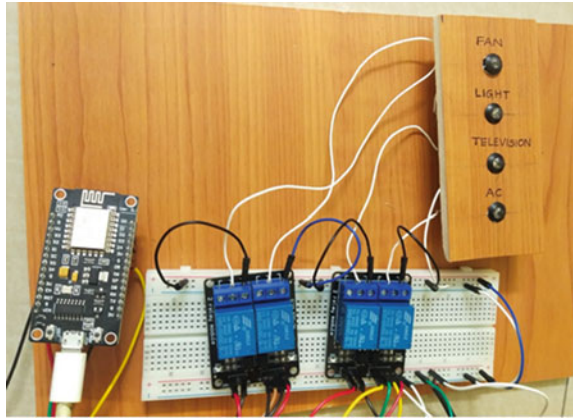
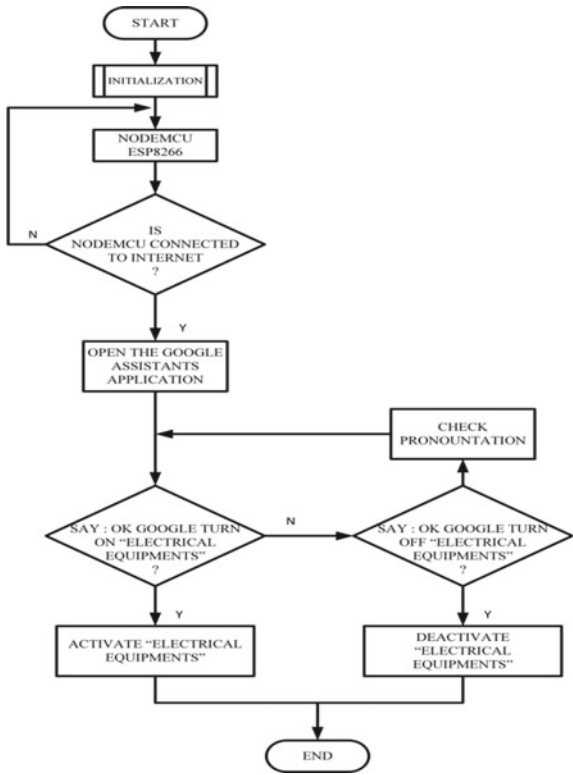


Fig. 5 Flowchart of model



is used to switch off the TV. The operation can be performed from anywhere with the only condition that there should be Internet connectivity. In the same way, we consider performing ON and OFF operation for a fan in **Case 2**, in **Case 3**, we considered light, and in **Case 4**, an air conditioner.

4 Software Applications

4.1 ARDUINO 1.8.15 Software

ARDUINO is an open-source electronics platform to interface hardware and software. The software helps in controlling the hardware based on the application that the hardware intend to perform for example reading a message, controlling a valve or a motor, logging the data, etc. In our project work, we are using the ARDUINO for ESP 8266 Node MCU which is not of ARDUINO but a commercially available microcontroller chip and a low cost. In order to use ESP 8266 chip, ARDUINO has the option to use the ESP 8266 in its preferences and then add BLYNK library.

4.2 BLYNK

The BLYNK is the software application which allows us to interface between the smart phone and the ESP 8266 microcontroller unit. The application takes charge post uploading the requisite library and ARDUINO code into the microcontroller. After loading the code, BLYNK software takes over the control of the microcontroller's input and output nodes. Therefore, all the electrical appliances are controlled using this application (Fig. 6).

4.3 IFTTT Application

IFTTT stands for "IF This Then That". IFTTT is a Website launched in 2010 with a motto of "put the Internet to work for you". The main vision of IFTTT is to automate everything from our favorite apps and Websites to application-enabled accessories and smart devices. Here, IFTTT application is used as an intermediate platform between the Google Assistant and the BLYNK application.

After logging into IFTTT Web page, we need to create an applet and then "This", i.e., the trigger, here, we select Google Assistant and then type the command to which the Google Assistant should respond and to this command the concerned home appliance actuate. The response from the Goggle Assistant can also be written as desired (Fig. 7).

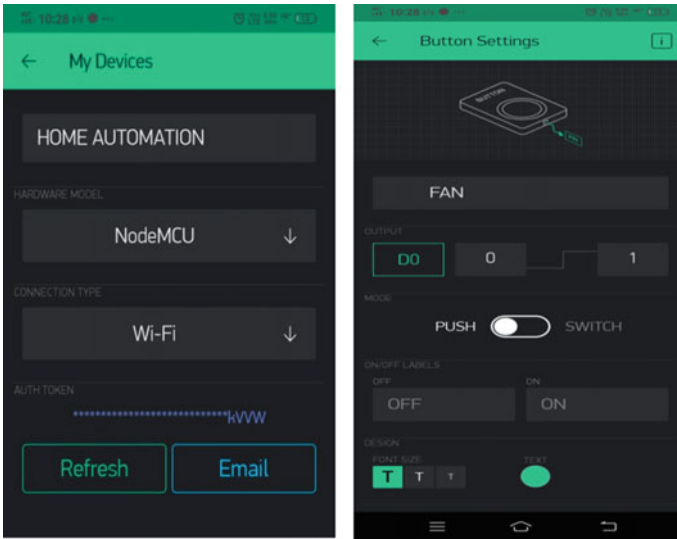


Fig. 6 BLYNK-IoT app

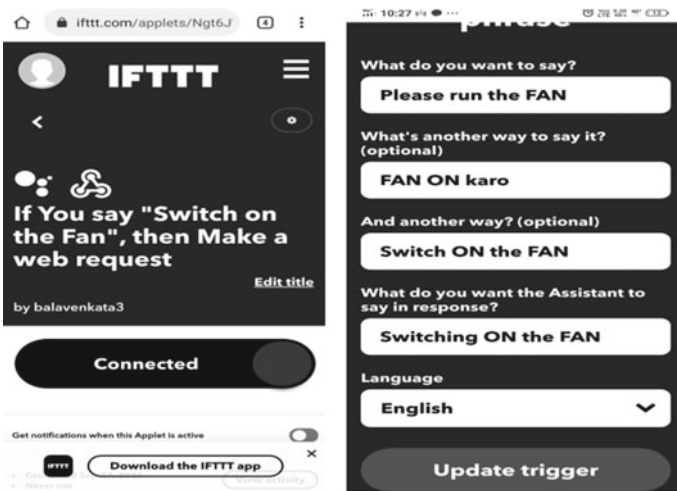
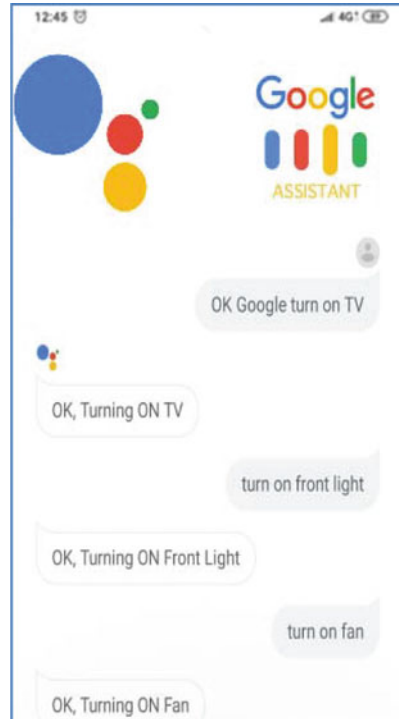


Fig. 7 IFFT application and command view

4.4 Google Assistant

It is a software application that permits users to have direct control over all the applications in the device using voice commands. It eases users and more specifically to the disabled people like blind as they only have to give voice commands to the

Fig. 8 Google Assistant illustration when voice commands given



Google Assistant. A easy to use application and for interfacing, the Google Assistant with BLYNK is done through IFTTT (Fig. 8).

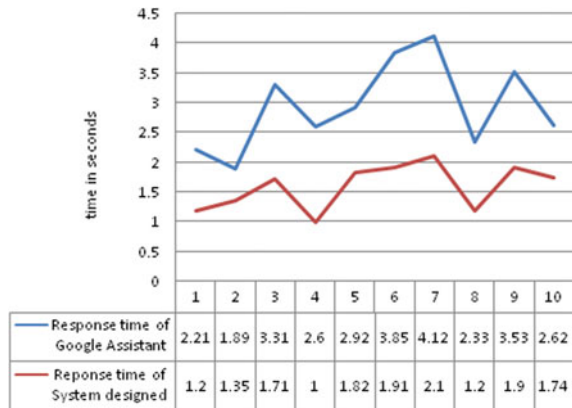
5 Results

- A. Checking the voice commands delivered with Google Assistant application. It aims to check the accent, correctness, and clarity of voice command when communicating with an Android smart phone using Google Assistant. If the voice command is correctly delivered, then only the Google Assistant will accept it; otherwise it will reject it. Table 1 is shown describes the response of Google Assistant for 15 instances.
- B. The time taken by Google Assistant to respond and the system designed during switching ON and OFF conditions are represented in Figs. 9 and 10.

Table 1 Response of Google Assistance against voice command delivered for 15 instances

No. of trials undertaken	Voice command delivered (correct/wrong)	Response from Google Assistant (accepted/rejected)
1	Correct	Accepted
2	Correct	Accepted
3	Wrong	Rejected
4	Correct	Accepted
5	Correct	Accepted
6	Correct	Accepted
7	Wrong	Rejected
8	Correct	Accepted
9	Correct	Accepted
10	Correct	Accepted
11	Wrong	Rejected
12	Correct	Accepted
13	Wrong	Rejected
14	Correct	Accepted
15	Correct	Accepted

Fig. 9 Switching ON condition

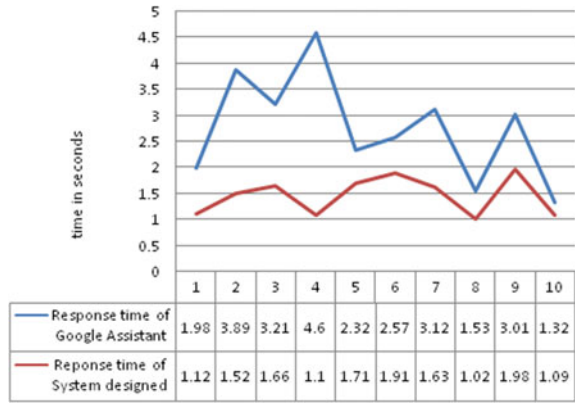


6 Conclusion

The goal of this article is to design and realize an IoT-based intelligent home using Google Assistant for managing common household appliances for the men in need. As the Google Assistant controlled home automation concept was effectively executed, the technique outlined in the paper was successful.

This article proposes and realizes a low-cost IoT-based intelligent home based on the ESP 8266 Node MCU microcontroller. Overall, Arduino is simple to comprehend

Fig. 10 Switching OFF condition



and program. We can also assure energy conservation and efficiency of our appliances with the aid of this technology. Specifically, the disabled people like blind can have total control over our household appliances from afar. It also improves human comfort and reduces human effort.

References

- Li RYM, Li HCY, Mak CK, Tang TB (2016) Sustainable smart home and home automation: big data analytics approach. *Int J Smart Home* 10(8):177–198
- Sharma R, Chirag P, Shankar V (2014) Advanced low-cost security system using sensors, Arduino and GSM communication module. In: *Proceedings of IEEE TechSym 2014 satellite conference*, VIT University
- Javale D, Mohsen M, Nandewar S, Shingate M (2013) Home automation and security using Android ADK
- Yavuz E, Hasan B, Serkan I, Duygu K (2007) Safe and secure PIC based remote control application for intelligent home. *Int J Comput Sci Netw Secur* 7(5)
- Sriskanthan N, Karand T (2002) Bluetooth based home automation system. *J Microprocess Microsyst* 26:281–289
- Kusuma SM (1999) Home automation using internet of things
- Shrotriya N, Kulkarni A, Gadhawe P (1996) Smart home using Wi-Fi. *Int J Sci Eng Technol Res (IJSETR)*
- Celtek SA, Durgun M, Soy H (2017) Internet of things based smart home system design through wireless sensor/actuator networks. In: *2017 2nd international conference on advanced information and communication technologies (AICT)*, pp 15–18
- Guravaiah K, Velusamy RL (2019) Prototype of home monitoring device using internet of things and river formation dynamics-based multi-hop routing protocol (RFDHM). *IEEE Trans Consum Electron* 65(3):329–338
- Li X, Lu R, Liang X, Shen X, Chen J, Lin X (2011) Smart community: an internet of things application. *IEEE Commun Mag* 49(11):68–75
- Wang D (2016) The internet of things the design and implementation of smart home control system. In: *2016 international conference on robots & intelligent system (ICRIS)*, pp 449–452
- Tang S, Kalavally V, Ng KY, Parkkinen J (2017) Development of a prototype smart home intelligent lighting control architecture using sensors onboard a mobile computing system. *Energy Build* 138:368–376

13. Kelly SDT, Suryadevara NK, Mukhopadhyay SC (2013) Towards the implementation of IoT for environmental condition monitoring in homes. *IEEE Sens J* 13(10):3846–3853
14. Suryanegara M, Arifin AS, Asvial M, Wibisono G (2017) A system engineering approach to the implementation of the internet of things (IoT) in a country. In: 2017 4th international conference on information technology, computer, and electrical engineering (ICITACEE), pp 20–23
15. Arifin AS, Suryanegara M, Firdaus TS, Asvial M (2017) IoT based maritime application: an experiment of ship radius detection. In: *Proceedings of the international conference on big data and internet of thing*, London
16. Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. *Comput Netw* 2787–2805
17. Chebudie AB, Minerva R, Rotondi D (2014) Towards a definition of the internet of things (IoT)
18. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M (2015) Internet of things: a survey on enabling technologies, protocols, and applications. *IEEE Commun Surv Tutor* 17(4):2347–2376
19. Pavithra D, Balakrishnan R (2015) IoT based monitoring and control system for home automation. In: 2015 global conference on communication technologies (GCCT), pp 169–173
20. Schneider GP (2015) *Electronic commerce*. Cengage Learning, Stamford, CT

Analysis of a Microstrip Log-Periodic Dipole Antenna with Different Substrates



Suresh Prasad, S. Aruna, and K. Srinivasa Naik

1 Introduction

In the field of wireless communication, highly directional as well as wide bandwidth antenna is preferred. Log-periodic dipole array antenna is used to fulfill wide bandwidth and high gain requirement. LPDA is capable of working on high frequency, very high frequency, and ultra-high frequency. However, when the frequency increases, the number of elements grows, making the structure unwieldy. To address this limitation, a log-periodic antenna is used in conjunction with a microstrip antenna. The microstrip antenna has the benefit of shrinking in size as the frequency increases. Because of their lightweight and small size, microstrip antennas are becoming increasingly common. However, the basic element's small bandwidth is a drawback of microstrip antennas. As a result, a combination of a log-periodic antenna with a microstrip antenna, known as a microstrip log-periodic dipole antenna, provides an efficient and broader bandwidth antenna (MLPDA). In the fields of tracking antennas, microwave radar communication, satellite communication, mobile phone antennas, and GPS antennas, MLPDA has a wide range of applications.

It has been always a point of research to increase the efficiency of antenna. The bandwidth of the microstrip antenna depends on patch shape, resonant frequency, dielectric constant, and the thickness of the substrate [1]. Microstrip patch antenna substrates range from 2.2 to 12. The lower the permittivity of dielectric material,

S. Prasad (✉)

Department of Electronics and Communication Engineering, Andhra University College of Engineering, Visakhapatnam, A.P., India

e-mail: sureshpdd@gmail.com

S. Aruna

Andhra University College of Engineering, Visakhapatnam, India

K. Srinivasa Naik

Vignan's Institute of Information Technology, Visakhapatnam, India

the larger the size of the antenna, but it achieves better efficiency and larger bandwidth. The dielectric constant (ϵ_r) is limited by radio frequency or microwave circuit connected to antennas. When substrates of higher dielectric constants were used, then the performance result degrades [2]. To achieve larger bandwidth, the antenna is designed without ground plane [3].

2 Design Specifications for Antenna

In this project design, first microstrip log-periodic antenna is designed, then radiation parameters are compared with different substrate material. A log-periodic array is a collection of dipole antennas of various sizes that are linked together and alternately supplied through a common transmission line. Design variables are apex angle α (alpha), spacing factor (τ), and scale factor (σ) [4].

Spacing factor (σ), scaling factor (τ), and substrate permittivity (ϵ_r) are important factor of antenna design. Bandwidth of microstrip antenna is inversely proportional to the substrate permittivity (ϵ_r) or dielectric constant. So here five different substrate materials are used to realize the antenna radiation characteristics return loss, VSWR, directivity, and gain.

Figure 1 shows the geometrical dimensions of the antenna the lengths l_n , spacing R_n , gap spacing at dipole centers S_n , diameters d_n of the LPA. Relationship [5] is defined by

$$\frac{1}{\tau} = \frac{l_2}{l_1} = \frac{l_{n+1}}{l_n} = \frac{R_2}{R_1} = \frac{R_{n+1}}{R_n} = \frac{d_2}{d_1} = \frac{d_{n+1}}{d_n} = \frac{s_2}{s_1} = \frac{s_{n+1}}{s_n} \tag{1}$$

Microstrip Log-Periodic Antenna Design Parameters

In the first step, log-periodic antenna is designed as with following method [5].

- a. Number of elements in the antenna array

It is calculated based on upper and lower frequencies of antenna. Formulated as

$$\log(\text{Upper frequency}) - \log(\text{Lower frequency}) = (n - 1) \log(1/\tau) \tag{2}$$

Here, n = no of elements, τ = spacing factor = 0.802,

Upper frequency = 3 GHz,

Lower frequency = 600 MHz,

Achieved, $n = 8$,

Number of elements, $n = 8$.

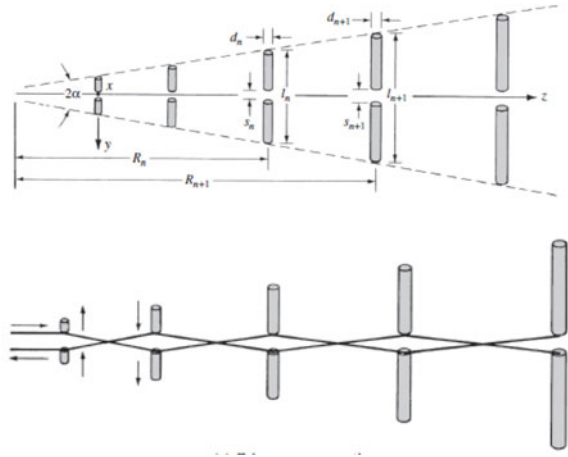
- b. Length of last dipole is calculated as

$$L_n = \lambda/2 = c/(2 * \text{upper frequency})$$

Fig. 1 Log-periodic dipole antenna structure

$$\frac{l}{\tau} = \frac{l_2}{l_1} = \frac{l_{n+1}}{l_n} = \frac{R_2}{R_1} = \frac{R_{n+1}}{R_n} = \frac{d_2}{d_1} = \frac{d_{n+1}}{d_n} = \frac{s_2}{s_1} = \frac{s_{n+1}}{s_n}$$

.....1



where speed of light, $c = 3 \times 10^8$ m/s.
Then,

$$L_n/L_{n+1} = \tau = 0.802$$

c. Distance of dipole elements is calculated by,

$$D_n/D_{n+1} = \tau = 0.802$$

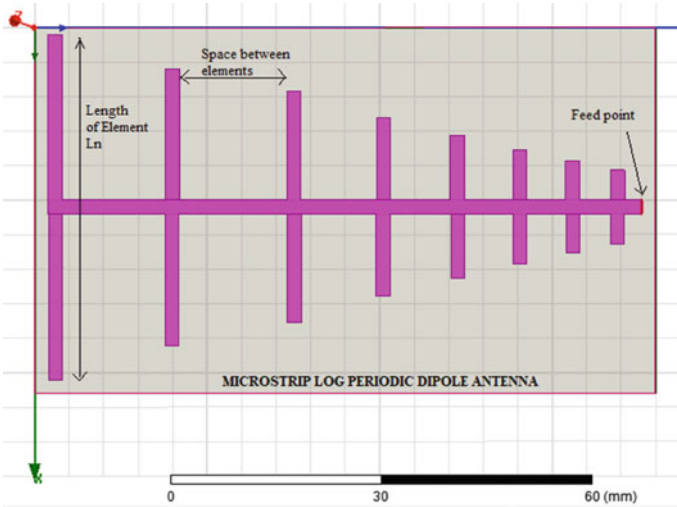
d. Width of the elements is fixed, 2 mm.
See Table 1.

Table 1 Length and distance of antenna element, width = 2 mm

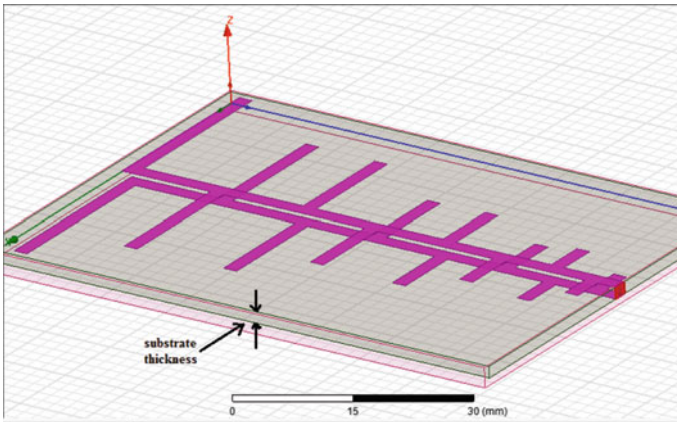
Elements	Length (mm)	Spacing (mm)
1	50	17
2	40.1	13.63
3	32.16	10.93
4	25.79	8.76
5	20.68	7.033
6	16.58	5.64
7	13.30	4.52
8	10.67	3.62

Antenna Model on HFSS

See Fig. 2.



(a)



(b)

Fig. 2 a Microstrip log-periodic antenna design. b Microstrip log-periodic antenna, substrate view

3 Antenna Simulation and Analysis with Different Substrate

The antenna properties were designed and analyzed using ANSOFT HFSS 15.0 software. To examine the output properties, I employed five different substrate materials (Table 2).

- A. **Substrate Material: Duroid, Relative Permittivity = 2.2**
See Figs. 3, 4, 5 and 6.
- B. **Substrate Material: Taconic TLC, Relative Permittivity = 3.2**
See Figs. 7, 8, 9 and 10.
- C. **Substrate Material: RO4003, Relative Permittivity = 3.55**
See Figs. 11, 12, 13 and 14.
- D. **Substrate Material: FR4 Epoxy, Relative Permittivity = 4.4**
See Figs. 15, 16, 17 and 18.

Table 2 List of substrate material

S. No.	Material	Relative permittivity, ϵ_r	Thickness (mm)
1	Duroid	2.2	1.6
2	Taconic TLC	3.3	1.6
3	RO4003	3.55	1.6
4	FR4 Epoxy	4.4	1.6
5	RO3005	6.15	1.6

Fig. 3 S11 (return loss)

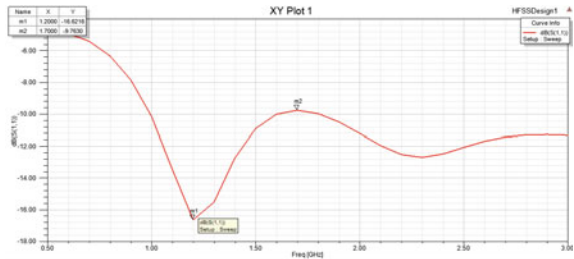


Fig. 4 VSWR

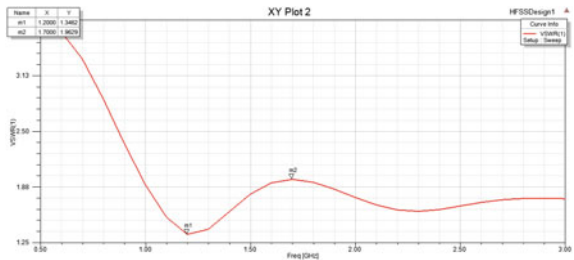


Fig. 5 3D plot, max gain

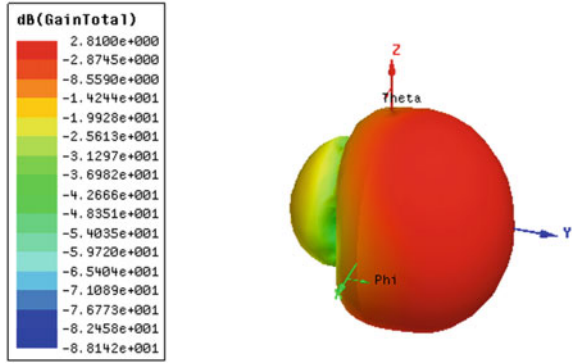


Fig. 6 Radiation pattern (gain)

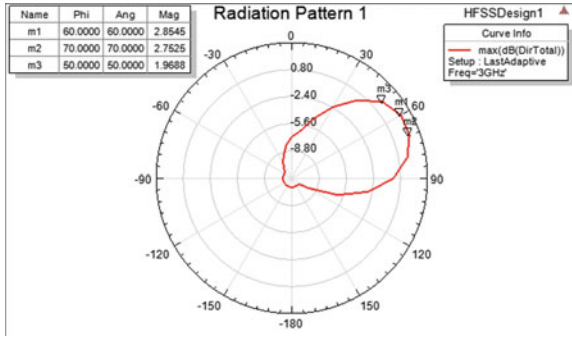


Fig. 7 S11 (return loss)

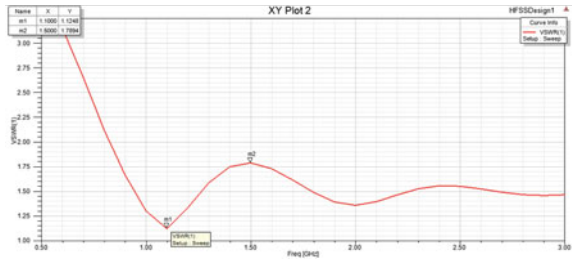


Fig. 8 VSWR

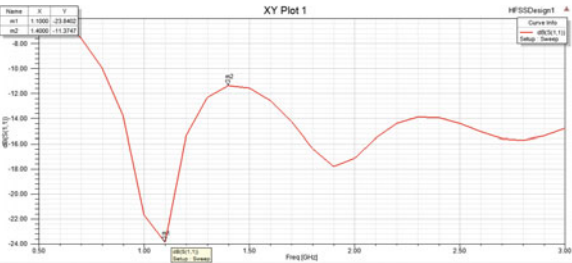


Fig. 9 3D plot, max gain

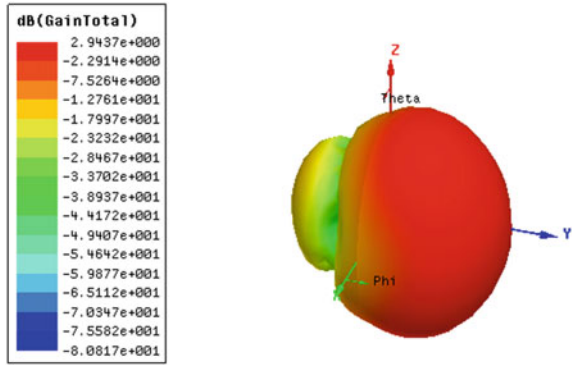


Fig. 10 Radiation pattern (gain)

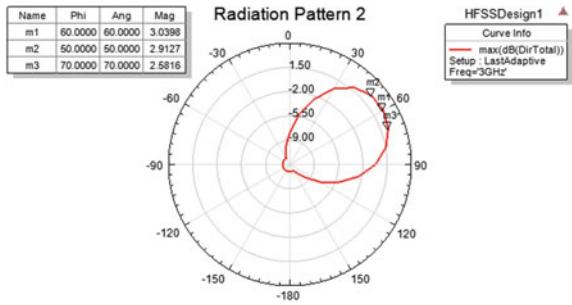


Fig. 11 S11 (return loss)

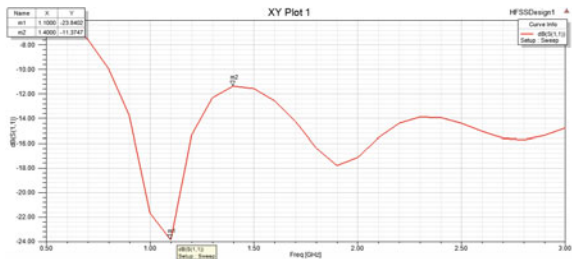


Fig. 12 VSWR

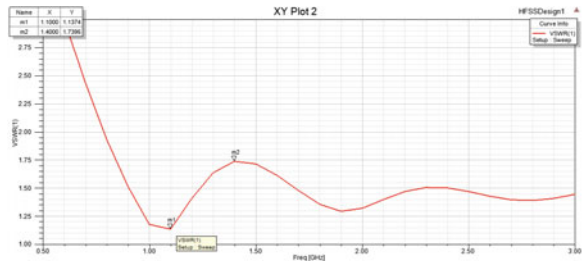


Fig. 13 3D plot, max gain

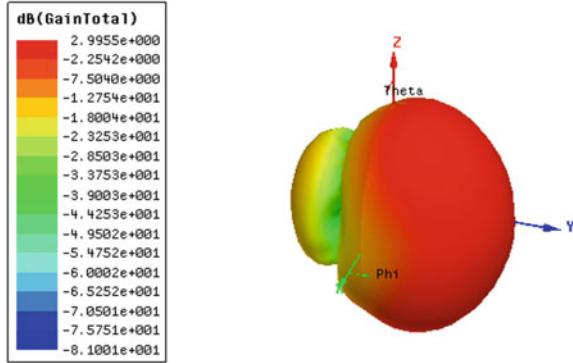


Fig. 14 Radiation pattern (gain)

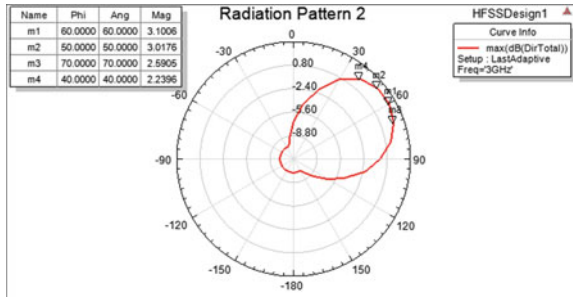


Fig. 15 S11 (return loss)

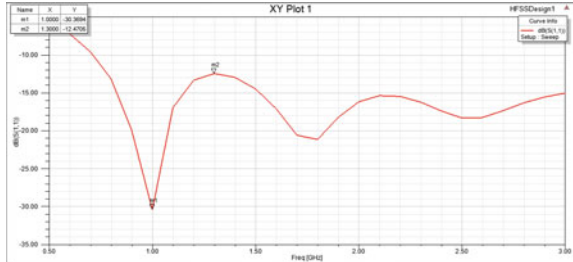


Fig. 16 VSWR

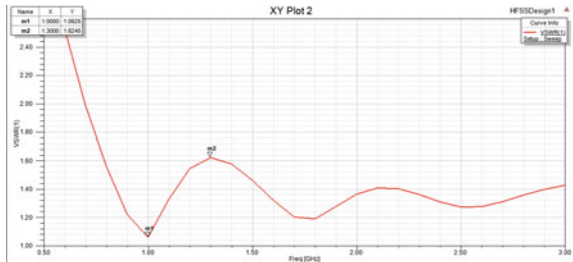


Fig. 17 3D plot, max gain

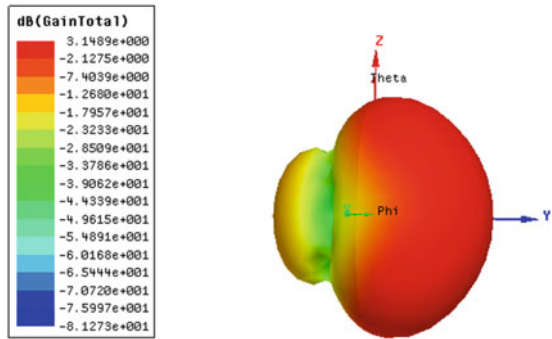


Fig. 18 Radiation pattern (gain)

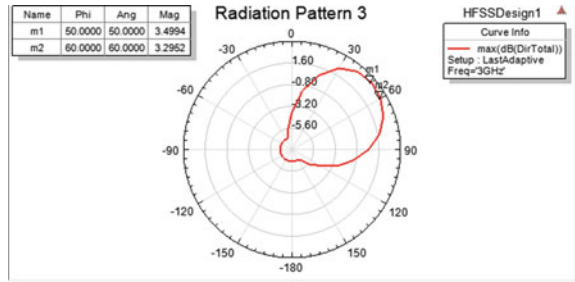
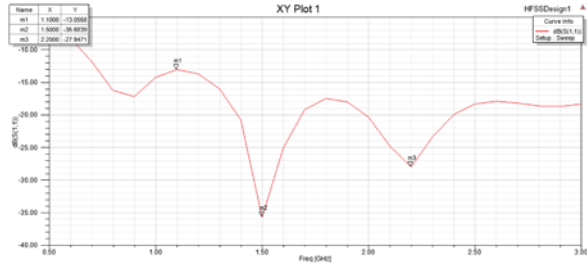


Fig. 19 S11 (return loss)



E. Substrate Material: Roger RO3005, Relative Permittivity = 6.15

See Figs. 19, 20, 21 and 22.

4 Result and Discussion

See Table 3.

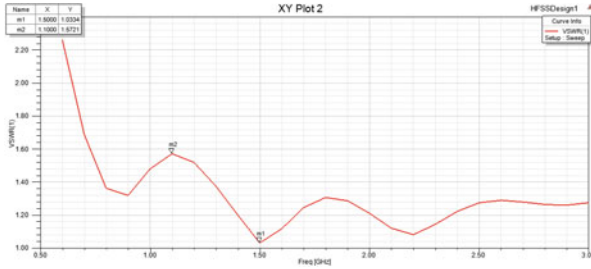


Fig. 20 VSWR

Fig. 21 3D plot, max gain

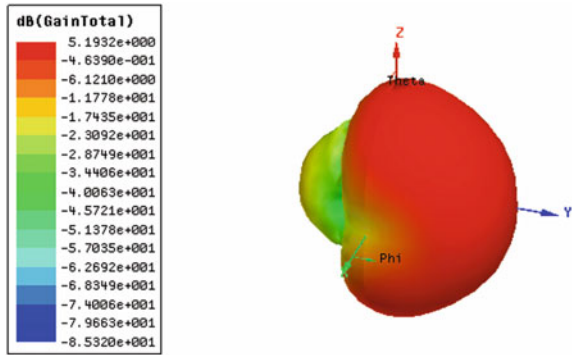


Fig. 22 Radiation pattern (gain)

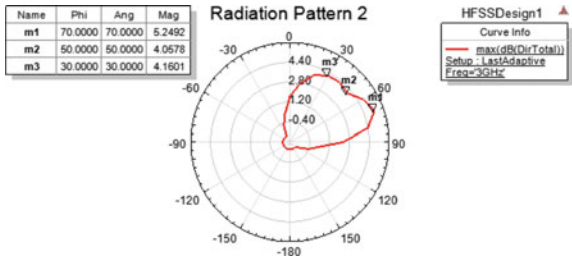


Table 3 Result

Parameters	Dielectric constant	S11 (return loss)	VSWR (max)	Gain
Duroid	2.2	-16.2	1.92	2.85
Taconic TLC	3.3	-24.62	1.78	3.03
Roger RO4003	3.55	-23.84	1.73	2.99
FR4 Epoxy	4.4	-30.36	1.62	3.14
Roger RO3005	6.15	-36.68	1.57	5.2

5 Conclusion

The properties of the microstrip log-periodic antenna were displayed in a table in five different circumstances. When the results are compared, it is discovered that when the dielectric constant of the substrate increases, the return loss, VSWR, and gain improve. As a result, the antenna's efficiency improves. The best material for this antenna is Roger RO3005. When choosing a substrate material, other factors such as size, price, availability, and loss tangent are taken into account.

6 Future Perspective

Different substrates and substrate thicknesses can be used to investigate the microstrip log-periodic dipole antenna. The antenna's shape can be altered to improve performance.

References

1. Rahim MKA, Gardner P (2004) The design of nine element quasi microstrip log periodic antenna. In: RF and microwave conference, RFM 2004, proceedings, 5–6 Oct 2004, pp 132–135
2. Jain K, Gupta K. Different substrates use in microstrip patch antenna—a survey. *Int J Sci Res (IJSR)*. ISSN (Online): 2319-7064
3. Casula G, Montisci G, Mazzarella G (2013) A wideband PET inkjet-printed antenna for UHF RFID. *IEEE Antennas Wirel Propag Lett* 12:1400–1403
4. Malusare S, Patil V, Wakode S, Bhilegaonkar SM. Microstrip log periodic antenna array. *Int J Adv Manag Technol Eng Sci*. ISSN no: 2249-7455
5. Balanis CA (2016) *Antenna theory: analysis and design*, 4th edn. Wiley, Hoboken, NJ

Detection of Diabetic Retinopathy Using Fundus Images



S. V. Viraktamath, Deepak Hiremath, and Kshama Tallur

1 Introduction

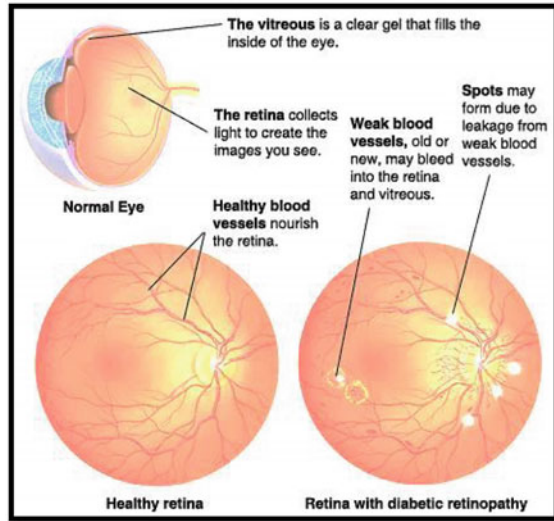
One of the key concerns of modern health care is the rapidly growing rate of diabetes as the number of people suffering from the condition is increasing at an alarming rate. Over the next 10 years, the World Health Organization predicts that the number of diabetes would rise from 135 to 400 million [1]. The fact that only half of the patients are aware of the ailment aggravates the issue. Diabetes, from a medical standpoint, causes serious late consequences. Heart disease, kidney difficulties, and retinopathy are among the consequences that might occur as a result of macro and microvascular alterations. Diabetic Retinopathy is a condition that develops as a result of long-term diabetes [2]. Arteriosclerosis, or the hardening and thickening of artery walls, plays a role in the development of cardiovascular illnesses, which are the leading cause of mortality in persons over the age of 45.

Figure 1 depicts the difference between a healthy retina and a non-healthy retina. The retina is the only place in the body where blood vessels may be seen noninvasively and in real-time. Digital ophthalmoscopes can now obtain very clear images of the retina, save them in a digital format, and do automated image processing and analysis. Despite the fact that this concept has piqued the interest of several research organizations, the problem remains unsolved. The retinal images impose important technical challenges, both while capturing and while processing them.

In the context of this study, the retina is the light-sensitive layer of the eye that is the most essential anatomical portion of the eye. The retina is a multi-layered structure made up of several cells that convert light into energy, pre-process visual information, and send neurological signals. Next to the choroid and pigment epithelium, the photoreceptive layer is the furthest away from the pupil. The retina receives

S. V. Viraktamath · D. Hiremath (✉) · K. Tallur
SDM College of Engineering and Technology, Dharwad, Karnataka, India
e-mail: hiremathdeepak62@gmail.com

Fig. 1 Comparison of healthy and non-healthy retina



a twofold blood supply from the top and bottom of the layer; the component that comes via the choroid provides 65% of the blood supply, while the portion that comes from the top of the retina provides 35%. The photoreceptive cells are separated into rods and cones, which provide achromatic and color vision, respectively.

The fovea, which has the largest density of cones, is responsible for pin-focus high-resolution coloring. Rods outnumber cones on the remainder of the retinal surface. The Optic Disc (OD) is the portion of the retina where neural fibers and blood arteries enter the retina; it contains no photoreceptive cells, thus the name “blind spot”. One artery and one vein enter the retina inside the OD and then branch out to fill the retinal tissue. From a technological standpoint, each vessel creates a tree-like structure in actual three-dimensional space, with one root at the OD. Two-dimensional projections of the trees overlap in the retinal pictures, generating vessel crossings and cycles. However, even in two-dimensional projections, the arteries do not cross arteries and veins do not cross veins [3].

2 Literature Survey

- Feng et al. [4], the authors of this paper investigate the relationship between DenseNet performance and connection density. In the beginning authors give the brief introduction to CNN and the evolution of algorithms. Then it explains about Connection trimming of DenseNet, where in the reduction of the connections in a dense block is elaborated. The implementation is for tiny picture inputs such as CIFAR and SVHN. They used the 264-layer DenseNet design, which is made up of four dense blocks of six, twelve, sixty-four, and forty-eight layers,

respectively. They came to the conclusion that connection-trimming architecture provides consumers a significant trade-off choice for tiny pictures like CIFAR and SVHN.

- Huang et al. [5] proposed through observation, the introduction to the DenseNet, a feed-forward network that connects each layer to every other layer. Author explained the other architecture such as Highway Networks, RESNET, ImageNet, and GoogleNet. In the architecture, ResNets Dense connectivity, Composite function, Pooling layers, Growth rate, Bottleneck layers, Compression are being elaborated. Colored natural pictures with 32×32 pixels and SVHN are included in the two CIFAR datasets. For training and testing, the Street View House Numbers (SVHN) dataset is employed, which comprises 32×32 colored digit pictures. On the majority of them, it was possible to make considerable gains above the state-of-the-art while using less computing to attain excellent performance.
- Mishra et al. [6], the authors of the research titled ‘Diabetic Retinopathy Detection Using Deep Learning’ focus on analyzing distinct phases of DR using Deep Learning (DL), which is a subset of Artificial Intelligence (AI). They created an architecture for automated detection of DR in which two architectures are evaluated to see which one is best in whatever situation. To automatically detect the DR stage, the DenseNet model is trained using a vast dataset of roughly 3662 train photos, which are then categorized into high resolution fundus images. VGG16 and DenseNet121 are the two designs, with accuracies of 73.26% and 96.11%, respectively.
- Albahli et al. [7], the objective of this paper is to develop an automatic and cost-effective method for classifying DR samples. DenseNet-65 model is used for computing the deep features from the given sample on which Faster-RCNN is trained for the goal of this work is to provide a system for identifying DR samples that is both automated and cost-effective. Faster-RCNN is trained for DR recognition using the DenseNet-65 model to compute deep features from a given sample. The precision attained is around 97.2%.

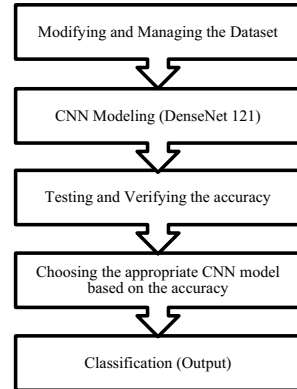
3 Methodology

The process of detection of Diabetic Retinopathy includes various steps. Initially, the images are captured and the dataset is created. Different CNN models are trained using the dataset. The test cases are being tested and verified. The algorithm which gives the best accuracy is considered. The same is depicted in Fig. 2.

3.1 Software Components

- Libraries used are:
 - i. OpenCV (CV2)—The Open-Source Computer Vision Library (OSCVL) is a programming library geared primarily at real-time computer vision.

Fig. 2 Flowchart of methodology



- ii. NumPy—A Python library that adds support for big, multi-dimensional arrays and matrices, as well as a vast variety of high-level mathematical methods to manipulate these arrays.
- iii. Matplotlib—A Python package that allows you to create static, animated, and interactive visualizations.
- iv. Keras—A Python interface for artificial neural networks is provided by this open-source software package. Keras serves as a user interface for TensorFlow.
- v. Pandas—A data manipulation and analysis software package created in the Python programming language. It includes data structures and methods for manipulating numerical tables and time series, in particular. It is open-source software with a three-clause BSD license.
- vi. Sklearn—Python’s most useful and stable machine learning library. It uses a Python consistency interface to give a set of fast tools for machine learning and statistical modeling, such as classification, regression, clustering, and dimensionality reduction.
- vii. SciPy—An open-source library for addressing issues in mathematics, science, engineering, and technology. It gives users the ability to alter and view data using a variety of high-level Python commands. SciPy is based on the NumPy Python extension.
- viii. TensorFlow—All developers can use this open-source machine learning framework. It’s used to build deep learning and machine learning applications.

- Database used: APTOS 2019.

APTOS was founded by a group of outstanding tele-ophthalmology specialists in the Asia–Pacific region in May 2016, the Asia Pacific Tele-Ophthalmology Society (APTOS) aims to bring together clinicians, researchers, technicians, institutes and organizations to form an alliance that promotes communication, exchange and collaboration in tele-ophthalmology. It provides a platform on which eye care or

Table 1 Train dataset with 5 classes of diabetic retinopathy from modified dataset

Diagnosis	No. of images
Normal (0)	1805
Mild (1)	999
Moderate (2)	370
Severe (3)	295
Proliferative (4)	193
Total	3662

tele-medical professionals can share knowledge and collaborate to deliver efficient, accessible and quality universal eye care throughout the region.

APTOS 2019 Dataset Summary

The dataset consists of 5590 images. The dataset is used as a train dataset and a test dataset. The total training dataset images are 3662 and testing dataset images are 1928. The training dataset images are classified into 5 types based on the severity of the disease which is depicted in Table 1.

3.2 Modifying and Managing the Database

The steps followed to manage and utilize the dataset are:

- Creating two sections of the database:
 - i. Train dataset—Labeling the train dataset by creating a related .csv file, this has two columns, name of the image, and diagnosis type.
 - ii. Test dataset—Creating a .csv file for the test data set which has similar columns but the second column (diagnosis type) is left empty, which is filled by the program after execution. New test images (if any), are updated in both the test data set and its related .csv file.

APTOS 2019 Modified Dataset Summary

The dataset consists of 5590 images. The dataset is used as a train dataset and a test dataset. The total training dataset images are 3662 and testing dataset images are 1928. The training dataset images are classified into 5 types based on the severity of the disease which is depicted in Table 2.

APTOS 2019 Partially Modified Dataset Summary

The dataset consists of 1395 images. The dataset is used as a train dataset and a test dataset. The total training dataset images are 950 and testing dataset images are 1928. The training dataset images are classified into 5 types based on the severity of the disease which is depicted in Table 3.

Table 2 Train dataset with 5 classes of diabetic retinopathy from modified APTOS dataset

Diagnosis	No. of images
Normal (0)	190
Mild (1)	190
Moderate (2)	190
Severe (3)	190
Proliferative (4)	190
Total	950

Table 3 Train dataset with 5 classes of diabetic retinopathy from partially modified APTOS dataset

Diagnosis	No. of images
Normal (0)	190
Mild (1)	190
Moderate (2)	190
Severe (3)	190
Proliferative (4)	190
Total	950

3.3 CNN Modeling (DENSENET 121)

DenseNet was proposed by Cornell Uni, Tsinghua Uni, and Facebook Research in the paper: “Densely Connected Convolutional Networks”. It is an architecture that focuses on making the deep learning networks go even deeper, but at the same time making them more efficient to train, by using shorter connections between the layers [8]. DenseNet achieves similar accuracy as ResNet on the large scale ILSVRC 2012 (ImageNet) dataset by employing less than half the number of parameters and nearly half the number of floating-point operations per second. DenseNet has been used on a variety of datasets. Different sorts of dense blocks are utilized depending on the dimensionality of the input. The following is a basic summary of these layers:

- i. Basic DenseNet Composition Layer: Each layer is followed by a pre-activated batch normalization layer, ReLU activation function, and 3×3 convolution in this form of dense block (Fig. 3).
- ii. Bottleneck DenseNet (DenseNet-B): Because each layer generates k output feature maps, computation becomes more difficult at each level. As a result, a bottleneck structure is adopted, with 1×1 convolutions used before a 3×3 convolution layer [10] (Fig. 4).
- iii. DenseNet Compression: The feature maps at the transition layers are lowered to increase model compactness. So, if a dense block has m feature maps and the transition layer produces i output feature maps, where $0 < i \leq 1$, i also signifies the compression factor. The number of feature mappings across transition layers remains unaltered if the value of i is one ($i = 1$). If i is less than 1, the architecture is known as DenseNet-C, and the value of i is set to 0.5. The model is known as

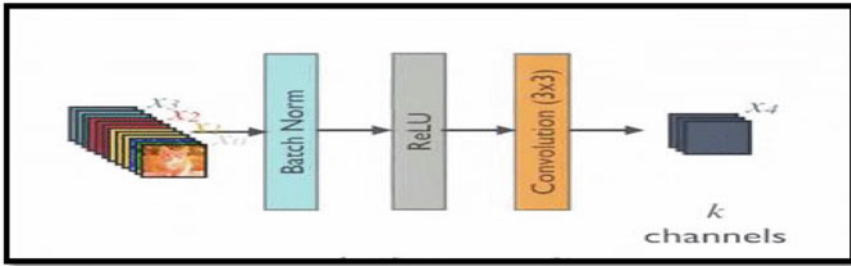


Fig. 3 Composition layer [10]

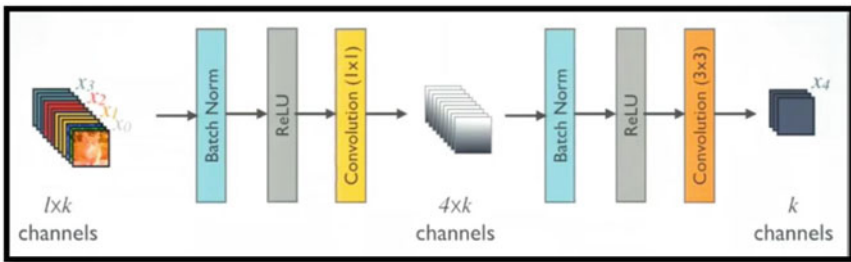


Fig. 4 Bottleneck structure [10]

DenseNet-BC when both bottleneck and transition layers with i less than 1 are employed.

- iv. Multiple Dense Blocks with Transition Layers: A 1×1 Convolution layer and a 2×2 average pooling layer follow the dense blocks in the design. It is simple to concatenate the transition layers when the feature map sizes are the same. A global average pooling is conducted at the conclusion of the dense block and is linked to a softmax classifier (Fig. 5).

4 Results

The results are obtained from training the CNN model is stated in the form of tables. Parameters are values that are given to the network when it is created; the network cannot learn these values during training. Image size [9], kernel size, number of layers in the neural network, batch size, number of epochs to train, and so on, these are some of the factors. Various terminologies used in this section are:

- Batch size: In one forward/backward pass, the amount of training instances is called as batch size. More memory space is required as the batch size grows.

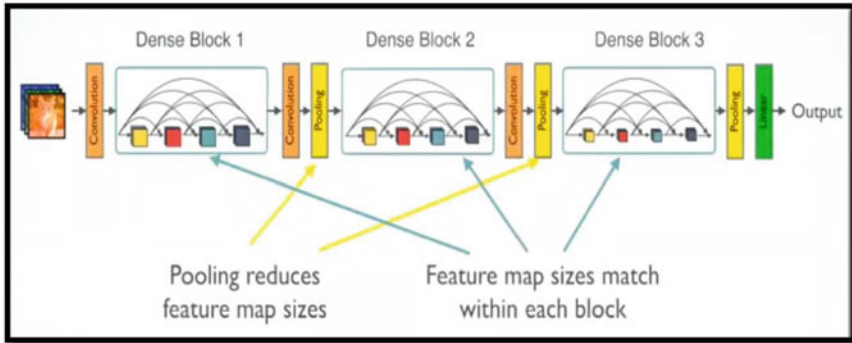


Fig. 5 Multiple dense blocks with transition layer [10]

- Epoch: When a whole dataset is only transported forward and backward through the neural network once, it is called an Epoch. Because a single epoch is too large to transmit to the computer all at once, it is split into smaller chunks [11].
- Accuracy: Number of correct predictions in total number of predictions.
- Train accuracy: The accuracy of a model on examples it was constructed on.
- Validation accuracy: The accuracy of a model on examples it has not seen.
- Kappa: The Kappa statistic is a measure of how well the cases categorized by the machine learning classifier matched the data labeled as ground truth, while accounting for the predicted accuracy of a random classifier [12].

After considering DenseNet model, the results of various trials which are obtained by changing the parameters such as image size, epoch, and batch size are shown in Tables 4, 5, and 6.

Table 4 Execution summary of APTOS test dataset for DenseNet

Trial No.	Image size	Batch size	Epoch	Max train accuracy (%)	Max validation accuracy (%)	Max validation kappa (%)
1	224 × 224	100	15	–	–	–
2	224 × 224	75	15	–	–	–
3	224 × 224	72	15	96.44	96.88	92.14
4	224 × 224	50	15	96.29	97.06	91.53
5	224 × 224	32	15	97.56	96.15	91.88
6	224 × 224	25	15	96.22	97.45	92.47
7	224 × 224	35	15	96.25	97.48	92.12

When the model was executed with different batch sizes, the bold values in the tables are the best batch sizes for which the accuracy is the highest when compared to other batch sizes

Table 5 Execution summary of modified APTOS test dataset

Trial No.	Image size	Batch size	Epoch	Max train accuracy (%)	Max validation accuracy (%)	Max validation kappa (%)
1	224 × 224	100	15	–	–	–
2	224 × 224	75	15	–	–	–
3	224 × 224	72	15	91.05	90.66	84.35
4	224 × 224	55	15	93.55	92.31	85.45
5	224 × 224	50	15	92.17	90.62	83.67
6	224 × 224	35	15	91.47	92.29	84.21
7	224 × 224	25	15	92.03	94.07	85.36

When the model was executed with different batch sizes, the bold values in the tables are the best batch sizes for which the accuracy is the highest when compared to other batch sizes

Table 6 Execution summary of partially modified APTOS test dataset

Trial No.	Image size	Batch size	Epoch	Max train accuracy (%)	Max validation accuracy (%)	Max validation kappa (%)
1	224 × 224	100	15	–	–	–
2	224 × 224	75	15	–	–	–
3	224 × 224	72	15	92.73	91.19	82.24
4	224 × 224	50	15	93.06	91.53	83.42
5	224 × 224	35	15	93.35	91.09	82.23
6	224 × 224	40	15	93.73	91.61	84.18
7	224 × 224	45	15	93.33	91.19	83.69

When the model was executed with different batch sizes, the bold values in the tables are the best batch sizes for which the accuracy is the highest when compared to other batch sizes

4.1 Summary of Epoch for Different Test Datasets

- i. Execution summary of APTOS test dataset.
- ii. Execution summary of modified APTOS test dataset.
- iii. Execution summary of partially modified APTOS test dataset.

4.2 Summary of Results Obtained from Different Test Datasets

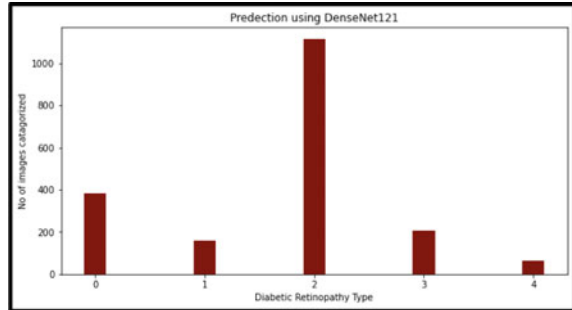
- i. Output summary of APTOS test dataset.

Table 7 summarizes the no of output images diagnosed by the DenseNet model in each category of the DR which use the APTOS dataset, followed by Fig. 6 which shows the graphical representation of the same.

Table 7 Output of test dataset for DenseNet

Diagnosis	No. of images
Normal (0)	381
Mild (1)	160
Moderate (2)	1116
Severe (3)	207
Proliferative (4)	64
Total	1928

Fig. 6 Graph of APTOS 19 dataset for DenseNet



- ii. Output summary of modified (balanced) APTOS test dataset.
Table 8 summarizes the no of output images diagnosed by the DenseNet model in each category of the DR which use the modified APTOS dataset, followed by Fig. 7 which shows the graphical representation of the same.
- iii. Output summary of partially modified APTOS test dataset.
Table 9 summarizes the no of output images diagnosed by the DenseNet model in each category of the DR which use the partially modified APTOS dataset, followed by Fig. 8 which shows the graphical representation of the same.

Table 8 Output of modified test dataset for DenseNet

Diagnosis	No. of images
Normal (0)	94
Mild (1)	81
Moderate (2)	84
Severe (3)	134
Proliferative (4)	07
Total	400

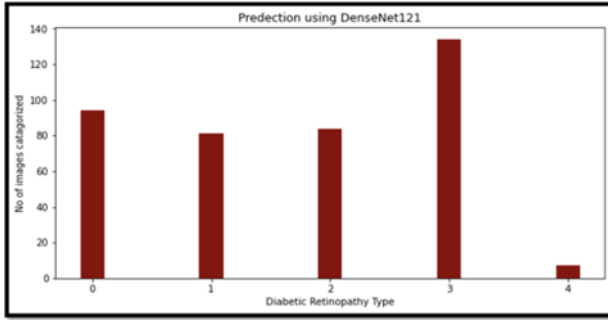


Fig. 7 Graph of modified APTOS 19 dataset for DenseNet

Table 9 Output of partially modified test dataset for DenseNet

Diagnosis	No. of images
Normal (0)	422
Mild (1)	402
Moderate (2)	327
Severe (3)	698
Proliferative (4)	79
Total	1928

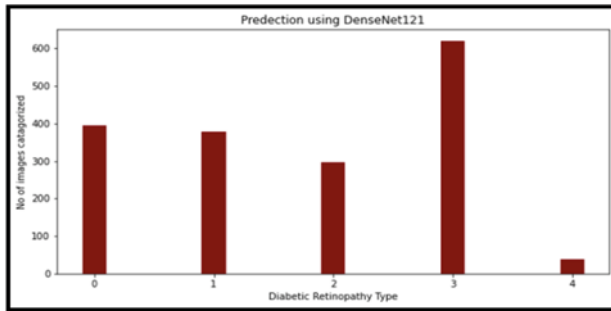


Fig. 8 Graph of partially modified APTOS 19 dataset for DenseNet

5 Conclusion

The CNN model DenseNet121 is tested with respect to the data. By comparison, it is evident that DenseNet121 offers good accuracy with respect to Detection of Diabetic Retinopathy using the fundus images. The error rate of prediction is found to be better than other CNN models.

Although, Diabetic Retinopathy Detection offers a unique opportunity to prevent a significant proportion of vision loss, the data obtained from the results of the

project stand alone cannot be considered as final diagnosis (without the consultation and human analysis of the physician), as the project does not deliver 100% accurate results, but it can be used by the physicians as it saves significant amount of time and efforts, discards human errors and helps the physicians to better judge the situation of the disease and the patient.

References

1. Author F (1998) World Diabetes, A newsletter from the World Health Organization, 4
2. Solanki MS (1998) CS365: artificial intelligence: diabetic retinopathy detection using eye. World Diabetes, A newsletter from the World Health Organization, 4
3. Umarfarooq AS. Blood vessel identification and segmentation from retinal images for diabetic retinopathy
4. Feng X, Yao H, Zhang S (2019) An efficient way to refine DenseNet. Springer-Verlag London Ltd., part of Springer Nature
5. Huang G, Liu Z, van der Maaten L, Weinberger KQ (2018) Densely connected convolutional networks. Facebook AI Research, 28 Jan 2018
6. Mishra S, Hanchate S, Saquib Z (2020) Diabetic retinopathy detection using deep learning. In: International conference on smart technologies in computing, electrical and electronics (ICSTCEE 2020)
7. Albahli S, Nazir T, Irtaza A, Javed A. Recognition and detection of diabetic retinopathy using DenseNet-65 based faster-RCNN. *Comput Mater Contin.* <https://doi.org/10.32604/cmc.2021.014691>
8. Xia M, Song W, Sun X, Liu J, Ye T, Xu Y (2019) Weighted densely connected convolutional networks for reinforcement learning. *Int J Pattern Recognit Artif Intell.* <https://doi.org/10.1142/S0218001420520011>
9. Luke JJ, Joseph R, Balaji M (2019) Impact of image size on accuracy and generalization of convolutional neural networks. *Int J Res Anal Rev (IJRAR)*
10. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
11. Kale S, Sekhar A, Sridharan K (2021) SGD: the role of implicit regularization, batch-size and multiple epochs. In: 35th conference on neural information processing systems (NeurIPS 2021)
12. Ben-Davi A (2008) Comparison of classification accuracy using Cohen's weighted kappa. *Expert Syst Appl* 34(2):825–832. <https://doi.org/10.1016/j.eswa.2006.10.022>

Artificial Intelligence in the Tribology: Review



Manoj Rajankunte Mahadeshwara , Santosh Kumar ,
and Anushree Ghosh Dastidar 

1 Introduction

There is considerable time spent in performing experiments and analyzing the results obtained. It is also expensive and requires modern technology to minimize the time and cost. As a result, the advancement in the application of computers in the field of mechanical systems has considerably increased over time [1]. Artificial neural network (ANN) is a method that plays a significant role in this application. It has been proved that ANN can effectively minimize the cost and time involved in conducting experiments [2].

This approach was formulated to study various tribological properties such as coefficient of friction, wear, lubricant properties, film thickness formation, other surface mechanical properties of composites, polymers, and so on. The models designed by ANN are enabled to predict the performance of a mechanism in the conceptual phase by using the critical performance parameters of the experiment. ANN is a mathematical tool that resembles the nervous system in the human brain. It accepts the required input and output data and solves complex engineering and scientific problems [3]. This mathematical technique is used for simulating and understanding mechanisms that are otherwise difficult to describe by experimental procedures. Furthermore, ANN has the capability of predicting the output with limited

M. R. Mahadeshwara (✉)
University of Leeds, Leeds LS29JT, UK
e-mail: mn20mrm@leeds.ac.uk

S. Kumar
BMS Institute of Technology and Management, Bengaluru, India
e-mail: santosh.kumar@bmsit.in

A. G. Dastidar
Queens University Belfast, Belfast BT71NN, UK
e-mail: aghoshdastidar01@qub.ac.uk

input data after the learning process which is not possible in the case of conventional analytical techniques [4]. A major advantage of ANN is its superior learning potential and its capability to build models of multi-dimensional, nonlinear, and complex functions. The ANN ‘learns’ by organizing the experimental data provided and by assuming the nature of the relationships in the given problem [5]. The insensitivity of the neural network to minute changes such as noise helps in avoiding errors. In recent times, experiments such as pin-on-disk (POD), fretting wear test, tensile, and compressive strength experiments have been investigated using ANN [6].

1.1 ANN Technique

An ANN is composed of basic units called artificial nodes, which resemble the neurons in the human brain. These neurons connect to form synapses and send signals. The neurons receive the signal, process it, and transmit it to the next receiving neuron. Each input neuron acts as the output of the previous layer of neurons [7]. The input values are scaled in distinct functions such as linear, logistic, and hyperbolic tangential functions. These inputs are multiplied with a weight factor that imitates the synaptic strength in the nervous system of the brain. They also determine the activation level of the neuron where it is manipulated by transfer functions to obtain the output signal. In most cases, the transfer function can be a sigmoid or logistic function of the form in Eq. (1).

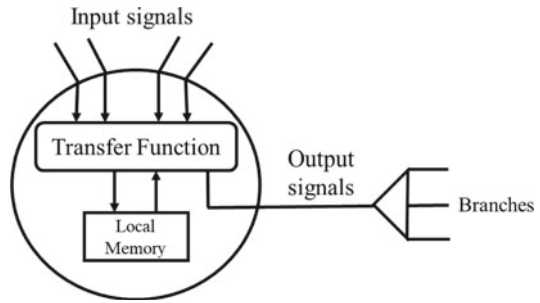
$$x = \frac{1}{1 + e^{(-x)}} \quad (1)$$

However, the transfer function can also be any function that represents the nonlinear characteristics of the system [8]. Complex relations can be modeled by using multiple neurons in single or in multiple layers. There are several types of neural network models with three things in common, namely the neurons, the connections, and the learning rules. Figure 1 illustrates the artificial neuron. A neural network model has an output that is dependent only on the input variables and the weight function. However, this can be modified by recurrent models where the output is re-circulated back to the neurons in the same or previous layers so the output thus generated will be changed at every stage [9].

1.2 Classifications of Neural Network Models

Based on the primary components in the ANN structure such as data flow, neurons, and the number of layers in the networks of ANN, the models of the neural network are classified into the following types.

Fig. 1 Schematic diagram of artificial neuron

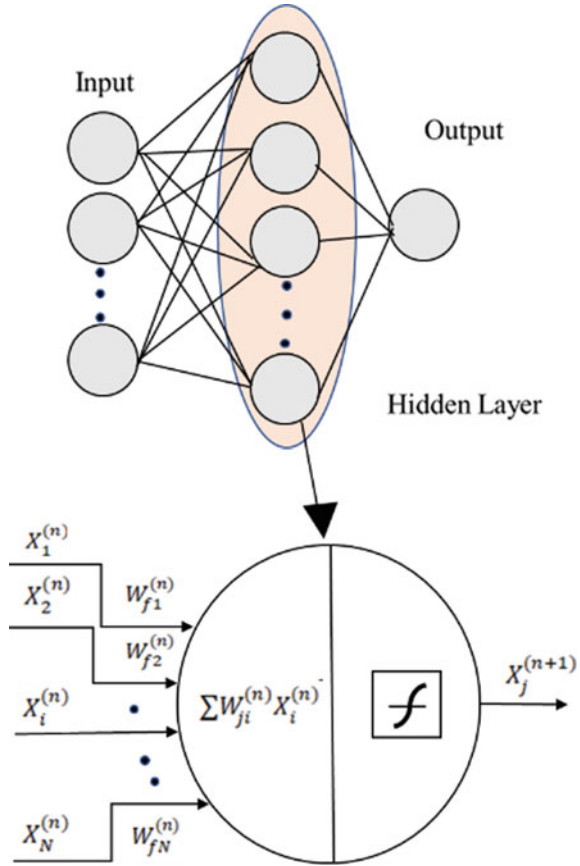


1.2.1 The perceptron is the basic model of neurons. It accepts the input data, processes it, and supplies the output data [10].

1.2.2 The feed-forward neural network is a widely used model in engineering applications [11]. It consists of single or multi-layered perceptrons in which the neurons are in the input, output, or hidden layers. The neurons present in the input and output layers communicate with the outside environment, while the neurons in the hidden layer communicate with interconnecting neurons. An ANN with an enlarged neuron is shown in Fig. 2. In the feed-forward neural network, the activation is fed from the input to the output layers via the weighted interconnections which are only in the forward direction. The lack of backpropagation in the single-layered feed-forward neural network is the reason for its loss of capability in deep learning. In the case of a multi-layer feed-forward neural network, the layers contain one or more hidden layers. These are placed between the input and the output layers, in which each layer has several nodes which are interconnected with each other. It has a bi-directional propagation approach, i.e., both forward and backward. The input data is multiplied with a weight factor and is fed in the form of an activation function. These neural networks use backpropagation to help modify the output, thereby reducing the error loss and improving its self-dependency [12]. Also, the difference in the predicted outputs and the trained input can be identified using this approach. Hence, these types of networks are used in deep learning and are popular among tribological applications [13].

Other types of neural networks include radial basis function neural networks, recurrent neural networks, convolutional neural networks, sequence-based modular neural networks, etc. These neural networks are used based on their applications. The details of these neural networks are beyond the scope of this paper. However, more information about these types of neural networks can be found in the following citations [14–16].

Fig. 2 Artificial neuron with activation function [13]



1.3 Training the ANN

The developed architecture should be fed with a training algorithm to perform its function of learning the input and providing the desired outputs. In this process, the applied inputs having the corresponding weights are adjusted in a way to obtain the desired output values. The network training will be considered complete when no further modifications in the input weights are necessary and when closer approximations to the output values are obtained. This minimizes the inaccuracy between the approximate and the actual output values [17]. Hence, weights are analyzed to determine the actual impact variable to produce the correct output. Greater the weights on a specific input variable, the larger the impact on the output parameter. This is decided as the contribution strength of the input variable [18].

Training an ANN is conducted by three methods, namely supervised learning, unsupervised learning, and reinforcement learning.

Supervised Learning: In the supervised learning method, both the input and desired output are provided. The network analyses the input data to produce an output which is then compared to the desired outputs. Any errors obtained are then circulated back into the network as weights. The final output is then provided after the weights are adjusted suitably. This process of adjusting weights will occur until the desired output is refined. The dataset that enables this training is called a training set [19].

Unsupervised Learning: In the unsupervised method of training, only the input datasets are provided along with a relative function without the desired output. The network segregates the distinct groups of datasets provided in a process called ‘clustering’ where the training examples are automatically grouped into categories based on similarities. This is followed by principal component analysis which discards the unwanted datasets and compresses the training dataset for identifying the most useful dataset. However, in the case of supervised learning, there are already pre-assigned category labels for desired outputs. The main advantage of unsupervised learning is the minimal workload compared to supervised learning [20].

Reinforcement Learning: It is applicable in areas such as operational research, game theory, information theory, and simulation-based theory, which is beyond the scope of this paper. However, for more details about reinforcement learning the following citations can be referred [21–23].

1.4 Backpropagation Process

Backpropagation is a method to compensate the weights in the neural network, in which algorithms are used to tune the weights based on the feedback of the errors obtained from the previous iteration [24]. A representation of the backpropagation neural network is displayed in Fig. 3. Tuning the weights in the network ensures lower error rates resulting in a more reliable model by increasing its generalization. Training the network using the backpropagation process involves three steps, namely the input or feed-forward of a training pattern, followed by analysis, and finally backpropagation or feedback of the accompanying errors, with final adjustments of the weights. This can be processed by different optimization strategies such as the gradient descent algorithm, momentum, stochastic gradient descent algorithm, Nesterov accelerated gradient, mini-batch gradient descent, and Adagrad [25].

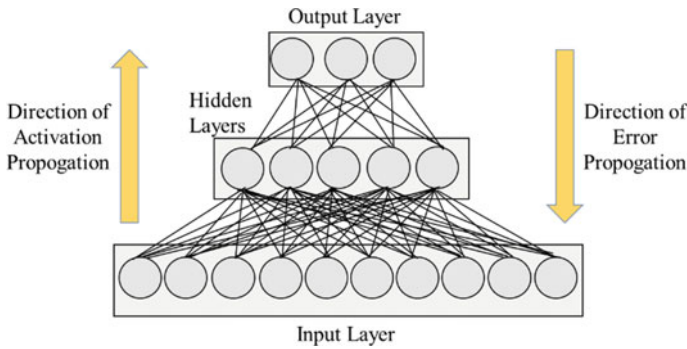


Fig. 3 Schematic representation of the backpropagation process

2 Tribological Properties Studied Using ANN Technique

2.1 Condition and Tool Wear Monitoring

The ANN technique has been adapted in various condition monitoring applications for different machining tools to reduce tool wear. These approaches are briefly explained below.

Lin and Lin [26] monitored the tool wear in a face milling operation using two methods. In the first method, i.e., the backpropagation neural network technique, the condition of the tool wear was obtained. The inputs provided were cutting parameters, and the output obtained was the average flank wear on cutter inserts. In the second method, a regression model was used to estimate the tool wear using the experimental data. In a regression model, the output is predicted as a function of the given input, and the input features can be either categorical (nominal or ordinal data) or numeric (continuous or discrete data). It was seen from both the models that an ANN can be utilized in predicting the tool wear for aluminum using a multi-tooth cutter and with varying geometries of the workpiece. Subrahmanyam and Sujatha [27] evaluated localized defects in ball bearings using two ANN methods, namely a multi-layered feed-forward neural network which was further trained with supervised error backpropagation (EBP) technique and an unsupervised adaptive resonance theory-2 (ART2). The ART2 is a method of segregating data in an unsupervised learning training, in which the data is evaluated using cohesion and separation. These networks were trained with vibrational accelerating signals from a rolling bearing test rig which helped in differentiating between a defective and a normal ball bearing. It was proved that these techniques were 95% capable of detecting the defects in the ball bearings. Assessing the wear in turning carbide inserts with the help of neural networks was investigated by Das et al. [28]. A simple neural network system with a 5–3–1 (5 input layers—3 hidden layers—1 output layer) structure was utilized for monitoring the cutting tool wear utilizing the components of the cutting force. The measured value was comparable to the model output but was unstable

due to external factors such as chipping and mild vibration that occurred during machining. Monitoring and detecting drilling wear during a cutting process were evaluated using multi-layer feed-forward neural network using a backpropagation algorithm. The neural network gave 90% of the accurate classification, and hence, it was concluded that a well-trained ANN would be an exceptionally dependable tool for solving pattern recognition problems in the applications that monitor the drilling process [29]. Following drilling wear, tool wear in a milling operation was predicted using a backpropagation neural network by Chen and Chen[30]. The input parameters used were the depth of cut, feed rate, and average peak force for 100 units acquired from experimental data. Tool wear could be predicted with an error of ± 0.037 mm on average using this neural network. Palanisamy et al. [31] compared the prediction of tool wear using a regression model and feed-forward neural network technique at an end milling operation. In this study, the neural networks were trained with experimental values for predicting flank wear in a tool. The predicted values by both the methods, i.e., the mathematical regression model with an error less than 5% and the feed-forward neural network technique with an error less than 2%, were found to be comparable to the obtained experimental results. Gouarir et al. [32] studied the tool flank wear in a milling operation incorporating a convolution neural network model. This neural network model was incorporated with the adaptive control which adjusted the feed rate and spindle speed to correct the flank wear, and this combination displayed accuracy of 90% similarities with experimental data. Ozel and Karpat [33] used a feed-forward neural network model for predicting tool wear and surface roughness in a hard turning process along with a regression technique. Thus, the prediction model developed was found to be capable of accurately predicting surface roughness and tool wear for the range in which it was trained. The neural network models were further compared to regression models. The neural network models provided better prediction capabilities than the regression models. Rao et al. [34] studied the development of a hybrid model using a multi-perceptron neural network in a neuro solution packaging which is a neural network solution software package for ANN simulation. The surface roughness in an electric discharge machine was optimized utilizing this model which resulted in a reduction of the error from 5 to 2%. Further analysis concluded that the type of materials used in the EDM influences the performance measures of the surface roughness.

2.2 ANN for Wear and Friction Properties

In addition to the condition monitoring, the ANN approach has also been utilized to analyze the wear and frictional properties of the materials. Jones et al. [35] first introduced Artificial Intelligence in tribology and showcased modeling neural networks for complex mechanical systems. Models for POD rig, rub shoe rig, and four-ball rig have emerged to predict wear regardless of the lubricants used in the system. The wear rates were predicted by extrapolating or interpolating the existing data between the known inputs which resulted in an approximate wear rate. Myshkin

et al. [36] utilized two techniques to classify wear debris based on its morphological features. Fourier descriptors were utilized to create a set of points in a cluster that depends on the location, morphology of the wear particles, and the current conditions of the contact system. These descriptors were used to coordinate with the cluster that was followed by training the backpropagation neural network. It was proved that the neural network was capable of classifying the wear debris. However, a large volume of the wear particles was required to classify the information hence, using different features would decrease the wear particle volume. Velten et al. [37] studied the wear behavior of glass, carbon fiber, poly tetra fluoro ethylene (PTFE), and graphite modified polyamide 4.6 (PA4.6) composites using three-layered feed-forward neural network technique. A database of 60 wear volume measurements was investigated and concluded with comparable predictions of the wear volume that were obtained in comparison to the neural network architecture investigated by Jones et al. [35]. Zhang et al. [38] studied the coefficient of friction and specific wear rate of short fiber reinforced PA4.6 composites using a multiple-layer feed-forward neural network based on experimental data. The predicted values obtained were comparable to the real test values which could be improved by expanding the training datasets and optimizing the neural network. Zhang et al. [39] studied erosive wear in three polymers, namely polyurethane, epoxy, and polyethylene which were modified by hydrothermally decomposed polyurethane and assessed using a multi-layer feed-forward neural network. The random datasets were selected, in which 35–80% of the tests predicted that the coefficient of determination, which is the value proportional to the variation in the dependent variable that is predictable from the independent variable, is greater than or equal to 0.9 in all the cases. Ranking this coefficient of determination property could provide information about the dominant properties among the polymers to cause erosive wear. Genel et al. [40] utilized a multi-layer feed-forward neural network to study the tribological properties of zinc-aluminum composites reinforced with alumina fiber. It was found that with increasing fiber volume fraction, a decrease in the specific wear rate, and by increasing the load, an increase in the wear rate was observed because the fibers enhance the mechanical strength in the composite. However, it was established that the composites had a better friction coefficient and wear resistance compared to that of the unreinforced materials due to the addition of short Saffil fiber (δ -Al₂O₃). This was further verified with the ANN model along with experimental data, and the degree of accuracy of the prediction was 99.4% and 94.2 for friction coefficient and specific wear rate, respectively, thus proving ANN as an excellent analytical tool. The structure of the ANN is shown in Fig. 4 along with the input and output variables.

ANN technique was also utilized in predicting the flank wear during drilling operation using backpropagation neural network algorithm and proved that various parameters for instance the spindle speed, feed rate, and drill diameter affect the flank wear and cutting conditions of a high-speed steel drill bit that is utilized for drilling holes in copper. These served as the input parameter in predicting the wear rate, and it was proved that the predicted values by the neural network coincided with the experimental data which was within $\pm 7.5\%$ of the experimental value [41]. Durmus et al. [42] predicted surface roughness and abrasive wear of the aluminum alloy AA 6351

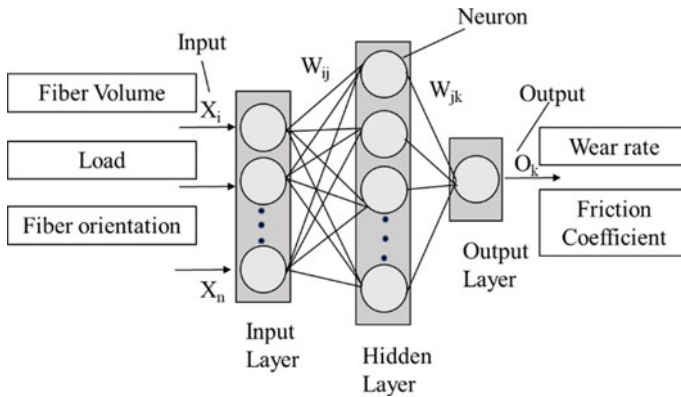


Fig. 4 The structure of the three-layered neural network to study the wear rate and COF [40]

using a multi-layered feed-forward neural network. The effects of aging in AA 6351 alloy under various conditions such as sliding speed, temperature, abrasive grit diameter, and load were assessed to find the test results obtained by the ANN technique to be comparable to the experimental data. Jiang et al. [43] reviewed the ANN applied in polyphenylene sulfide polymer composites to study their tribological properties and concluded that certain experimental datasets such as abrasive wear, COF experiments, or any tribological characteristic experimental datasets are needed to predict the properties of polymers for a well-designed neural network. Once the data is fed to the neural network, it can predict the output without performing further experiments. The wear and frictional properties of short carbon fiber (SCF) reinforced polyphenylene sulfide (PPS) composites and particles of sub-micro-titanium di-oxide (TiO_2) were studied using Powell–Beale conjugate gradient algorithm which is an algorithm used for largely unconstrained optimization. This is the method of finding the local minimum in a function irrespective of whether the function is differentiable or not [44]. In this study, it was found that PPS with a composition of 15 vol.% SCF as well as 5 vol.% TiO_2 experienced the least specific wear rate due to the existence of SCF fillers in the composite matrix. However, the more stable composition of PPS with 6 vol.% TiO_2 and 15 vol.% SCF was predicted using backpropagation neural network which was further examined by scanning electron microscope and concluded that the hybrid reinforcement provided a positive rolling effect between the sliding surfaces by protecting the SCFs being pulled out of the matrix. Rashed and Mahmoud [45] studied the sliding wear behavior of aluminum A356 alloy and Silicon Carbide (SiC) metal matrix composite using the multi-layer perceptron ANN approach. It was noted that the wear behavior was influenced by factors such as SiC particle size, SiC weight percent, test temperature, and applied pressure. This was analytically verified using the ANN approach proving that the ANN approach saves time and cost of the experiment. Younesi et al. [46] predicted the abrasive wear behavior of a hydroxyapatite bio-composite of nickel-free stainless steel with a backpropagation neural network

model considering various input parameters such as the hydroxyapatite volume fraction, wear load, and wear distance to predict the output parameter, i.e., wear volume. The model was further used to predict the loss of volume of different composites for wear distances in the range of 0–1000 m at different applied wear loads. The specific wear rate and friction coefficient of polyphenylene sulfide composites were predicted using the ANN model using input variables of mechanical and thermomechanical properties (e.g., compressive and tensile properties tested at room temperature along with dynamic mechanical thermal analysis properties analyzed in the range of 23–230 °C). Optimizing and improving the ANN was done through the implementation of the optimal brain surgeon (OBS) method [47]. The OBS algorithm is a powerful technique for network optimization, which is used to improve the efficiency and performance of the ANN. This algorithm identified and removed the irrelevant nodes in the ANN model. Further, the optimized ANN predicted the tribological properties of PPS composite material which was comparable to the real test values [48]. The abrasive wear rates of Cu-Al₂O₃ nanocomposites were predicted using the backpropagation neural network and multi-variable regression analysis. This is a model used to establish the relationship between multiple independent variables and a dependent variable. The data acquired from a range of wear tests was used as the input and compared with the two models. In addition to this, a comparison was made by implementing the genetic algorithm (GA) which is a technique of solving unconstrained and constrained problems using the process of natural selection. It was shown that ANN with GA is a better tool to predict the abrasive wear rate accurately on Cu-Al₂O₃ nanocomposite materials when compared with the results shown of ANN without GA [49]. Kumar et al. [50] predicted the dry sliding wear in a metal matrix composite of aluminum 6061 alloy reinforced with Al₂O₃ using a backpropagated neural network to show a nonlinear relation between wear and other influential factors, i.e., density, the weight percentage of reinforcement, and the load applied sliding distance. The nonlinear relationship between applied load, density, sliding distance, the weight percentage of reinforcement, and height with wear has been predicted using ANN. The results obtained by the ANN were comparable with that of experimental results. A study was conducted to predict the wear between a rail and the train wheel with different contact conditions using a nonlinear autoregressive model with exogenous input neural network. This is a nonlinear autoregressive model which has the exogenous inputs that can relate both the current and the past values of the externally determined series which influences the series of interest. The results were obtained as the mean absolute percentage error from the ANN which was compared with the experimental data from the profilometer demonstrating equivalent results [51]. Borjali et al. [52] quantified the results of various POD experiments using a series of machine learning techniques. To begin with, an interpretable model-based method such as linear regression was used, wherein the relationship between input and target attributes is defined. In addition, data-driven models have been used that utilize a dataset, without explicitly defining the relation between the input and target attributes. Here, the neural network is trained by the dataset from polyethylene wear rate and relates the operating parameters to the wear rate of polyethylene by employing neurons that communicate with each other in a nonlinear manner.

Instance-based methods like the *K*-Nearest Neighbor (KNN) technique were then implemented, which predicted the wear rate of polyethylene, based on clustering the data into subgroups, thus reducing the prediction error. This study proves that the data-driven model can successfully predict the polyethylene wear rate for new POD experiments provided that the operating parameters fall within the dataset ranges that were used for training the model. This could help to reduce the need for more experimental studies or designing a new experiment.

2.3 Lubrication and Lubricant Formulation

In addition to the friction, wear, and other tribological properties, the ANN technique can also be applied in the lubrication and lubricant formulation of various oils. Bhaumik et al. [53] studied bio-degradable lubricants based on various vegetable oils (e.g., palm oil, coconut oil, castor oil, etc.) with additives of nano-frictional modifiers such as carbon nanotubes and graphene. The database from the previous literature was used to train the ANN model along with the genetic algorithm. This study was performed to study the simulation understandings of the four-ball test and POD experimental results. The predicted results proved that the ANN technique can be implemented to study and design lubricants with various tribological properties. Furthermore, a feed-forward neural network ANN technique was utilized to analyze the experimental database obtained from a four-ball tester and POD technique to predict the anti-wear properties in the castor oil with dispersing non-carbonaceous and carbonaceous friction modifiers such as graphite, zinc oxide nanoparticles, multi-walled carbon nanotubes, and graphene. The speed, load, and concentration of the friction modifiers were the input variables to obtain the COF in this experiment [54]. It was concluded that the COF of the lubricant with the multi-frictional modifiers is 40–50% lower and the diameter of the wear scar is 87.5% less in comparison with other mineral oils. Furthermore, the method of implementing ANN for analyzing the compound relationship between the percentages of vegetable oil in the fuel mixtures to reduce the coefficient of friction was assessed [55]. The data obtained from experiments such as the POD for sunflower seed oil in biodiesel mixtures was used as an input. A backpropagation neural network algorithm was used to predict the data which showed a perfect correlation between the experimental. Two types of biodiesel were compared with 0 and 6.5% sunflower oil. It was found that the coefficient of friction in the biodiesel having 6.5% of sunflower oil was more than double that of the 0% sunflower oil in biodiesel which thus identified the optimal percentage of sunflower oil in the biofuel mixture.

2.4 *Surface Modification and Technologies*

Rutherford et al. [56] studied the abrasive wear resistance of a multi-layered coating of TiN/NbN deposited by physical vapor deposition using a multi-layer perceptron model. It was found that the most influential parameters on abrasive wear included the hardness of the interlayer, deposition pressure, interlayer mixing, and relative proportion of two layers of the material with the multi-layer coatings. Moder et al. [57] identified the lubrication regimes in hydrodynamic journal bearings. The neural network was trained using logistic regression models with high-speed data signals from torque sensors. The results obtained displayed the 99.25% accuracy of fast Fourier transforms of the high-speed torque signals to predict lubrication regimes with distinctive frequencies. Gorasso and Wang [58] optimized the journal bearing for its power loss and mass flow using a genetic algorithm and a multi-perceptron neural network. The ANN was trained with Reynold's equation and computational fluid dynamic simulations from Ansys Gambit and Ansys Fluent. In conclusion, it was shown that ANN can accurately predict the performance like power loss and mass flow of a hydrodynamic journal bearing. In a different study, the feed-forward neural network was utilized to predict the friction coefficient in lubricated conditions. The neural networks were trained through the data obtained from tribological tests on a mini-traction machine which were then compared with conventional simulation tools such as linear regression models. It was concluded that the ANN can be used as an excellent simulation tool to predict the COF in the thermal elastohydrodynamic contacts [59].

2.5 *Materialistic Properties*

Along with the tribological properties of materials, the materialistic properties have also been studied using the ANN approach. Zhang et al. [60] investigated the damping and storage modulus of SCF-reinforced PTFE-based composites using the dynamic mechanical thermal analysis method. The results obtained in this method were further verified using the Bayesian regularization of a backpropagated algorithm. Bayesian regularization decreases the linear combination of weights and squared errors, which proved that the complexity of the nonlinear relation between the input and output data increased, as the number of the training dataset increased. This proved that ANN is the potential analytical tool for structural property analysis of polymer composites. Altinkok and Koker [61] studied the tensile properties and the density of $\text{Al}_2\text{O}_3/\text{SiC}$ dual ceramic reinforced aluminum matrix composites produced by a stir casting process using a backpropagated neural network with a gradient descent learning algorithm. The sizes of SiC particles were provided as an input to the neural network. The density and the tensile strength values were predicted using the neural network with an error of 0.000472 compared to the experimental results. Koker et al. [62] assessed the mechanical properties such as bending strength and the hardness behavior of

the Al–Si–Mg metal matrix composites using various neural network training algorithms. The four different neural networks were investigated in studying the bending strength and the hardness behavior by feeding the SiC size particle as the input. The neural networks studied were quasi-Newton, Levenberg–Marquardt, resilient backpropagation, and variable learning rate backpropagation. In this comparative study, it was found that the Levenberg–Marquardt algorithm supplied the high accuracy and fastest prediction of the output in the composites due to its speed in prediction. In a different study, the fretting wear and mechanical properties (Izod impact energy, tensile strength, modulus, flexural strength, and modulus) of the reinforced PA composites with two experimental databases were studied by Jiang [63]. The ANN equipped with a backpropagated algorithm was trained with the input of 101 independent wear tests from PA 4.6 composite and the 93 pairs of independent tension test, Izod impact test, and bending tests of PA 6.6 composites. The property profiles of the composite as a function of the short fiber content were suitably predicted by the neural network, thus proving the capability of a well-optimized model. Partheepan [64] evaluated the fracture toughness of different steels such as chromium steel (H11), die steel (D3), medium carbon steel (MC), and low carbon steel (LC) using a miniature specimen test and feed-forward neural network. The load elongation obtained from finite element methods was fed as the input to the feed-forward neural network model. The fracture toughness was predicted as the output which was then compared to the ASTM standard test results. The obtained results were varying from 1 to 6.63% accuracy for various materials. Hassan et al. [65] predicted the porosity, density, and hardness of aluminum-copper-based composite materials using the volume fraction of the reinforced particle and the weight percentage of copper as the input using the feed-forward backpropagated neural network model. The maximum absolute relative error using the ANN technique did not exceed 5.99% proving the ANN to be a time- and cost-saving analytical tool. Hafizpour et al. [66] analyzed the influence of reinforcing particles on the compressibility of Al–SiC composite powders by utilizing a backpropagated neural network model. An accuracy of 97% was predicted by the model and the outcome of the reinforced particle size and the volume fraction on the densification of the Al–SiC composite powder using iso-density curves. These curves refer to an ellipse on a two-dimensional scatter diagram or a scatter plot that encircles a specified proportion of the cases constituting groups which in this case is the Si group. Suresh et al. [67] evaluated the solid particle erosion on PPS composites using Bayesian regularization trained neural network. The steady-state erosion rates were calculated at different velocities and impact angles using silica sand particles as an erodent to obtain experimental values which were verified with the predicted values from the neural network, thus obtaining the results which matched the experimental values.

3 Conclusion

The purpose of this study was to summarize the potential of ANN in the field of tribology. ANN is a promising mathematical technique used in modeling and is one of the most efficient techniques in reducing the time for analysis compared to the conventional modeling techniques. Complex nonlinear fundamental problems can be solved using ANN which can help tackle various tribological problems. Tribological properties such as condition and tool wear monitoring, wear and friction, lubrication and lubricant formulation, and studying surface modification and technologies and other materialistic properties have been studied using the application of ANN which contributes to a reduction in the duration of an experiment. Using the first experimental datasets, subsequent results can be predicted without performing experiments which is one of the most promising applications of ANN. ANN is thus a powerful simulation tool and can direct future tribologists to use this technique effectively to save time and resources.

References

1. Kartalopoulos SV (1998) An associative RAM-based CAM and its application to broadband communications systems. *IEEE Trans Neural Networks* 9(5):1036–1041. <https://doi.org/10.1109/72.712186>
2. Hammer B (2001) Neural smithing—supervised learning in feedforward artificial neural networks. *Pattern Anal Appl* 4(1):73–74. <https://doi.org/10.1007/s100440170029>
3. Zeng P (1998) Neural computing in mechanics. *Appl Mech Rev* 51(2):173–197. <https://doi.org/10.11105/1.3098995>
4. Wen B, Ravishankar S, Pfister L, Bresler Y (2020) Transform learning for magnetic resonance image reconstruction: from model-based learning to building neural networks. *IEEE Signal Proc Mag* 37(1):41–53. <https://doi.org/10.1109/MSP.2019.2951469>
5. Murata N, Yoshizawa S, Amari S (1994) Network information criterion-determining the number of hidden units for an artificial neural network model. *IEEE transactions on neural networks* 5(6):865–872. <https://doi.org/10.1109/72.329683>
6. El Kadi H (2006) Modeling the mechanical behavior of fiber-reinforced polymeric composite materials using artificial neural networks—a review. *Compos Struct* 73(1):1–23. <https://doi.org/10.1016/j.compstruct.2005.01.020>
7. Mishra M, Srivastava M (2014) A view of artificial neural network. In: *International conference on advances in engineering & technology research*, pp 1–3. <https://doi.org/10.1109/ICAETR.2014.7012785>
8. Jang JSR, Sun CT, Mizutani E (1997) Neuro-fuzzy and soft computing—a computational approach to learning and machine intelligence [book review]. *IEEE Trans Autom Control* 42(10):1482–1484. <https://doi.org/10.1109/tac.1997.633847>
9. Sandercock PP (1999) Dictionary for clinical trails. *Brain* 122(12):2413. <https://doi.org/10.1093/brain/122.12.2413>
10. Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev* 65(6):386–408. <https://doi.org/10.1037/h0042519>
11. Sampaio TP, Ferreira Filho VJM, Neto ADS (2009) An application of feed forward neural network as nonlinear proxies for use during the history matching phase. In: *Latin American and Caribbean petroleum engineering conference*, Cartagena de Indias, Colombia. <https://doi.org/10.2118/122148-MS>

12. Laguna M, Martí R (2002) Neural network prediction in a system for optimizing simulations. *IIE Trans* 34:273–282. <https://doi.org/10.1023/A:1012485416856>
13. Argatov I (2019) Artificial neural networks (ANNs) as a novel modeling technique in tribology. *Frontiers Mech Eng* 5(30). <https://doi.org/10.3389/fmech.2019.00030>
14. Bishop CM (1995) *Neural networks for pattern recognition*. Oxford University Press
15. Dash CSK, Behera AK, Dehuri S, Cho SB (2016) Radial basis function neural networks: a topical state-of-the-art survey. *Open Comput Sci* 6(1):33–63. <https://doi.org/10.1515/comp-2016-0005>
16. Parfitt S (1991) *An introduction to neural computing* by Igor Aleksander and Helen Morton, Chapman and Hall, London, 1990, pp 255, £15.95. *The Knowl Eng Rev* 6(4):351–352. <https://doi.org/10.1017/s0269888900005968>
17. Burger C, Traver R (1996) Applying neural networks system auditing. *EDPACS EDP Audit Control Secur Newsl* 24(6):1–10. <https://doi.org/10.1080/07366989609452285>
18. Marshall JA (1995) Neural networks for pattern recognition. *Neural Netw* 8:493–494. [https://doi.org/10.1016/0893-6080\(95\)90002-0](https://doi.org/10.1016/0893-6080(95)90002-0)
19. Ojha VK, Abraham A, Snášel V (2017) Metaheuristic design of feedforward neural networks: a review of two decades of research. *Eng Appl Artif Intell* 60:97–116. <https://doi.org/10.1016/j.engappai.2017.01.013>
20. Buhmann J, Kuhnel H (1992) Unsupervised and supervised data clustering with competitive neural networks. In: *IJCNN international joint conference on neural networks*, vol 4, pp 796–801
21. Jaksch T, Ortner R, Auer P (2010) Near-optimal regret bounds for reinforcement learning. *J Mach Learn Res* 11(4)
22. Busoniu L, Babuska R, De Schutter B, Ernst D (2017) *Reinforcement learning and dynamic programming using function approximators*. CRC press
23. Sutton RS, Barto AG (2018) *Reinforcement learning: an introduction*. MIT press
24. Leung H, Haykin S (1991) The complex backpropagation algorithm. *IEEE Trans Sign Process* 39(9):2101–2104. <https://doi.org/10.1109/78.134446>
25. Chauvin Y, Rumelhart DE (2013) *Backpropagation: theory, architectures, and application*. Psychology press
26. Lin SC, Lin RJ (1996) Tool wear monitoring in face milling using force signals. *Wear* 198(1–2):136–142. [https://doi.org/10.1016/0043-1648\(96\)06944-x](https://doi.org/10.1016/0043-1648(96)06944-x)
27. Subrahmanyam M, Sujatha C (1997) Using neural networks for the diagnosis of localized defects in ball bearings. *Tribol Int* 30(10):739–752. [https://doi.org/10.1016/s0301-679x\(97\)00056-x](https://doi.org/10.1016/s0301-679x(97)00056-x)
28. Das S, Roy R, Chattopadhyay AB (1996) Evaluation of wear of turning carbide inserts using neural networks. *Int J Mach Manuf* 36(7):789–797. [https://doi.org/10.1016/0890-6955\(95\)00089-5](https://doi.org/10.1016/0890-6955(95)00089-5)
29. Abu-Mahfouz I (2003) Drilling wear detection and classification using vibration signals and artificial neural network. *Int J Mach Tools Manuf* 43(7):707–720. [https://doi.org/10.1016/s0890-6955\(03\)00023-3](https://doi.org/10.1016/s0890-6955(03)00023-3)
30. Chen JC, Chen JC (2004) An artificial-neural-networks-based in-process tool wear prediction system in milling operations. *Int J Adv Manuf Technol* 25(5–6):427–434. <https://doi.org/10.1007/s00170-003-1848-y>
31. Palanisamy P, Rajendran I, Shanmugasundaram S (2007) Prediction of tool wear using regression and ANN models in end-milling operation. *Int J Adv Manuf Technol* 37(1–2):29–41. <https://doi.org/10.1007/s00170-007-0948-5>
32. Gouarir A, Martínez-Arellano G, Terrazas G, Benardos P, Ratchev SJPC (2018) Process tool wear prediction system based on machine learning techniques and force analysis. *Procedia CIRP* 77:501–504. <https://doi.org/10.1016/j.procir.2018.08.253>
33. Özel T, Karpaz Y (2005) Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks. *Int J Mach Tools Manuf* 45(4–5):467–479. <https://doi.org/10.1016/j.ijmachtools.2004.09.007>

34. Rao GKM, Ranganjardhaa G, Hanumantha Rao D, Sreenivasa Rao M (2009) Development of hybrid model and optimization of surface roughness in electric discharge machining using artificial neural networks and genetic algorithm. *J Mater Process Technol* 209(3):1512–1520. <https://doi.org/10.1016/j.jmatprotec.2008.04.003>
35. Jones SP, Jansen R, Fusaro RL (1997) Preliminary investigation of neural network techniques to predict tribological properties. *Tribol Trans* 40(2):312–320. <https://doi.org/10.1080/10402009708983660>
36. Myshkin NK, Kwon OK, Grigoriev AY, Ahn HS, Kong H (1997) Classification of wear debris using a neural network. *Wear* 203:658–662. [https://doi.org/10.1016/s0043-1648\(96\)07432-7](https://doi.org/10.1016/s0043-1648(96)07432-7)
37. Velten K, Reinicke R, Friedrich K (2000) Wear volume prediction with artificial neural networks. *Tribol Int* 33(10):731–736. [https://doi.org/10.1016/s0301-679x\(00\)00115-8](https://doi.org/10.1016/s0301-679x(00)00115-8)
38. Zhang Z, Friedrich K, Velten K (2002) Prediction on tribological properties of short fibre composites using artificial neural networks. *Wear* 252(7–8):668–675. [https://doi.org/10.1016/s0043-1648\(02\)00023-6](https://doi.org/10.1016/s0043-1648(02)00023-6)
39. Zhang Z, Barkoula NM, Karger-Kocsis J, Friedrich K (2003) Artificial neural network predictions on erosive wear of polymers. *Wear* 255(1–6):708–713. [https://doi.org/10.1016/s0043-1648\(03\)00149-2](https://doi.org/10.1016/s0043-1648(03)00149-2)
40. Genel K, Kurnaz SC, Durman M (2003) Modeling of tribological properties of alumina fiber reinforced zinc-aluminum composites using artificial neural network. *Mater Sci Eng A* 363(1–2):203–210. [https://doi.org/10.1016/s0921-5093\(03\)00623-3](https://doi.org/10.1016/s0921-5093(03)00623-3)
41. Singh AK, Panda SS, Chakraborty D, Pal SK (2005) Predicting drill wear using an artificial neural network. *Int J Adv Manuf Technol* 28(5–6):456–462. <https://doi.org/10.1007/s00170-004-2376-0>
42. Durmuş HK, Özkaya E, Meri C (2006) The use of neural networks for the prediction of wear loss and surface roughness of AA 6351 aluminium alloy. *Mater Des* 27(2):156–159. <https://doi.org/10.1016/j.matdes.2004.09.011>
43. Jiang Z, Zhang Z, Friedrich K (2007) Prediction on wear properties of polymer composites with artificial neural networks. *Compos Sci Technol* 67(2):168–176. <https://doi.org/10.1016/j.compscitech.2006.07.026>
44. Zhenyu J, Gyurova LA, Schlarb AK, Friedrich K, Zhang Z (2008) Study on friction and wear behavior of polyphenylene sulfide composites reinforced by short carbon fibers and sub-micro TiO₂ particles. *Compos Sci Technol* 68(3–4):734–742. <https://doi.org/10.1016/j.compscitech.2007.09.022>
45. Rashed FS, Mahmoud TS (2009) Prediction of wear behaviour of A356/Sicp MMCs using neural networks. *Tribol Int* 42(5):642–648. <https://doi.org/10.1016/j.triboint.2008.08.010>
46. Younesi M, Bahrololoom ME, Ahmadzadeh M (2010) Prediction of wear behaviors of nickel free stainless steel-hydroxyapatite bio-composites using artificial neural network. *Comput Mater Sci* 47(3):645–654. <https://doi.org/10.1016/j.commatsci.2009.09.019>
47. Hassibi B, Stork DG, Wolff GJ (1993) Optimal brain surgeon and general network pruning. In: *IEEE international conference on neural networks*, vol 1, pp 293–299. <https://doi.org/10.1109/ICNN.1993.298572>
48. Gyurova LA, Miniño-Justel P, Schlarb AK (2010) Modeling the sliding wear and friction properties of polyphenylene sulfide composites using artificial neural networks. *Wear* 268(5–6):708–714. <https://doi.org/10.1016/j.wear.2009.11.008>
49. Fathy A, Megahed AA (2011) Prediction of abrasive wear rate of in situ Cu–Al₂O₃ nanocomposite using artificial neural networks. *Int J Adv Manuf Technol* 62(9–12):953–963. <https://doi.org/10.1007/s00170-011-3861-x>
50. Kumar GBV, Pramod R, Rao CSP, Shivakumar Gouda, PS (2018) Artificial neural network prediction on wear of Al6061 alloy metal matrix composites reinforced with-Al₂O₃. *Mater Today: Proc* 5(5):11268–11276. <https://doi.org/10.1016/j.matpr.2018.02.093>
51. Shebani A, Iwnicki S (2018) Prediction of wheel and rail wear under different contact conditions using artificial neural networks. *Wear* 406:173–184. <https://doi.org/10.1016/j.wear.2018.01.007>

52. Borjali A, Monson K, Raeymaekers B (2019) Predicting the polyethylene wear rate in pin-on-disc experiments in the context of prosthetic hip implants: deriving a data-driven model using machine learning methods. *Tribol Int* 133:101–110. <https://doi.org/10.1016/j.triboint.2019.01.014>
53. Bhaumik S, Mathew BR, Datta S (2019) Computational intelligence-based design of lubricant with vegetable oil blend and various nano friction modifiers. *Fuel* 241:733–743. <https://doi.org/10.1016/j.fuel.2018.12.094>
54. Bhaumik S, Pathak SD, Dey S, Datta S (2019) Artificial intelligence based design of multiple friction modifiers dispersed castor oil and evaluating its tribological properties. *Tribol Int* 140:105813. <https://doi.org/10.1016/j.triboint.2019.06.006>
55. Humelnicu C, Ciortan S, Amortila V (2019) Artificial neural network-based analysis of the tribological behavior of vegetable oil-diesel fuel mixtures. *Lubricants* 7(4):32. <https://doi.org/10.3390/lubricants7040032>
56. Rutherford KL, Hatto PW, Davies C, Hutchings IM (1996) Abrasive wear resistance of TiN/NbN multi-layers: measurement and neural network modelling. *Surf Coat Technol* 86:472–479. [https://doi.org/10.1016/s0257-8972\(96\)02956-8](https://doi.org/10.1016/s0257-8972(96)02956-8)
57. Moder J, Bergmann P, Grün F (2018) Lubrication regime classification of hydrodynamic journal bearings by machine learning using torque data. *Lubricants* 6(4):108. <https://doi.org/10.3390/lubricants6040108>
58. Gorasso L, Wang L (2014) Journal bearing optimization using nonsorted genetic algorithm and artificial bee colony algorithm. *Adv Mech Eng* 6:213548. <https://doi.org/10.1155/2014/213548>
59. Echávarri Otero J, De La Guerra Ochoa E, ChacónTanarro E, LafontMorgado P, DíazLantada A, Muñoz-Guijosa JM, Muñoz Sanz JL (2013) Artificial neural network approach to predict the lubricated friction coefficient. *Lubr Sci* 26(3):141–162. <https://doi.org/10.1002/lvs.1238>
60. Zhang Z, Klein P, Friedrich K (2002) Dynamic mechanical properties of PTFE based short carbon fibre reinforced composites: experiment and artificial neural network prediction. *Compos Sci Technol* 62(7–8):1001–1009. [https://doi.org/10.1016/s0266-3538\(02\)00036-2](https://doi.org/10.1016/s0266-3538(02)00036-2)
61. Altinkok N, Koker R (2006) Modelling of the prediction of tensile and density properties in particle reinforced metal matrix composites by using neural networks. *Mater Des* 27(8):625–631. <https://doi.org/10.1016/j.matdes.2005.01.005>
62. Koker R, Altinkok N, Demir A (2007) Neural network based prediction of mechanical properties of particulate reinforced metal matrix composites using various training algorithms. *Mater des* 28(2):616–627. <https://doi.org/10.1016/j.matdes.2005.07.021>
63. Jiang Z, Gyurova L, Zhang Z, Friedrich Z, Schlarb AK (2008) Neural network based prediction on mechanical and wear properties of short fibers reinforced polyamide composites. *Mater Des* 29(3):628–637. <https://doi.org/10.1016/j.matdes.2007.02.008>
64. Partheepan G, Sehgal DK, Pandey RK (2008) Fracture toughness evaluation using miniature specimen test and neural network. *Comput Mater Sci* 44(2):523–530. <https://doi.org/10.1016/j.commatsci.2008.04.013>
65. Hassan AM, Alrashdan A, Hayajneh MT, Mayyas AT (2019) Prediction of density, porosity and hardness in aluminum–copper-based composite materials using artificial neural network. *J Mater Process Technol* 209(2):894–899. <https://doi.org/10.1016/j.jmatprotec.2008.02.066>
66. Hafizpour HR, Sanjari M, Simchi A (2009) Analysis of the effect of reinforcement particles on the compressibility of Al–SiC composite powders using a neural network model. *Mater Des* 30(5):1518–1523. <https://doi.org/10.1016/j.matdes.2008.07.052>
67. Suresh A, Harsha AP, Ghosh MK (2009) Solid particle erosion studies on polyphenylene sulfide composites and prediction on erosion data using artificial neural networks. *Wear* 266(1–2):184–193. <https://doi.org/10.1016/j.wear.2008.06.008>

House Price Prediction Using Advanced Regression Techniques



Hemin Vasani, Harshil Gandhi, Shrey Panchal, and Shakti Mishra

1 Introduction

Machine learning predictions have been proven very useful and for stock price predictions, market trends, etc. Real estate is one of the prime fields to apply the ideas of machine learning on how to enhance and foresee the costs with high accuracy [1]. We have applied machine learning for the problem statement and house price prediction. The task consists of predicting the price of a house depending upon various factors. These factors include variables such as square feet, number of rooms, and location. There are countless numbers of features that can influence the price. We have approached the problem through feature engineering and applied methods such as imputation, handling outliers, log transforming skewed variables, OneHotEncoding categorical features, and feature selection.

The organization of this paper is as follows: Sect. 2 provides brief review on contemporary work done by the researchers. Section 3 presents brief description about different methods applied throughout the process along with respective mathematical formulation. The description of the dataset and experimental setup is present in Sect. 4. Section 5 contains the experimental results and analysis on the popular house pricing dataset, and the final section presents the conclusions and future work.

H. Vasani · H. Gandhi · S. Panchal · S. Mishra (✉)
School of Technology, Pandit Deendaya Energy University, Gandhinagar, India
e-mail: shakti.mishra@sot.pdpu.ac.in

H. Vasani
e-mail: hemin.vce17@sot.pdpu.ac.in

H. Gandhi
e-mail: harshil.gce17@sot.pdpu.ac.in

S. Panchal
e-mail: shrey.pce17@sot.pdpu.ac.in

2 Literature Survey

In [2], the authors propose a hybrid model to predict house prices. They have employed the feature engineering techniques that improved the data normality and linearity of data and defined RMSE for a different number of features. Truong et al. [3] implement random forest, XgBoost, and LightGBM machine learning models. If there is much irrelevant and redundant information present or if the data is noisy and missing, then knowledge discovery during the training phase becomes difficult [4]. Data pre-processing gave us an insight into the data and extraction of features through different feature selection methods allowed us to model advanced regression techniques on the most important features. Apart from this, we have tried to work toward the interpretability of the machine learning models. Models like random forest, gradient boosting, and XGBoost are considered to be the black-box models.

3 Methodology

We started by applying feature engineering and its different techniques. We performed imputation, handled outliers, log transformation of target variable, OneHotEncoding of categorical features, and feature selection. Imputation mainly deals with the blank/missing values in a dataset.

For this work, we set the threshold value to 0.7. Now there are two types of imputations, these are as follows:

- In numerical imputation, the missing value is generally replaced with the default value or the median value. For example: If the column consists of values that are only YES or NO.
- In categorical imputation, the missing value is replaced with the value that has occurred the most number of times. But if there is not a formidable value which can be taken due to uniform distribution, then a category like ‘none’.

Outliers can be defined as the points which deviate considerably from other points. Here, we deal with them using inter-quartile range. Using IQR, we defined a range of acceptable values between the 1st quartile and the 3rd quartile. The values that do not fall within the range are considered the outliers, and the corresponding entries are removed from the dataset. We defined the lower quartile to be 0.15 and upper quartile to be 0.85. This process is not so robust, but it will help the model remove extreme outliers which would supposedly affect the model performance if not taken care of. As we will see that the target variable [‘Sale Price’] is right skewed, and there is a possibility that at the very end of the tail, the data points might be considered to be outliers. The tail region may act as an outlier for the statistical model, and we know that outliers adversely affect the model’s performance especially regression-based models. Hence here, log transformation comes into play. It helps in handling the skewed data, and it transforms it into somewhat of a normal distribution.

After this, we applied three feature selection techniques: (1) Feature-based method: Here, we used the Pearson correlation to find the correlation of variables with the target variable ‘Sale Price’. (2) Wrapper-based method: Here, we used the recursive feature elimination method with decision tree regressor. Recursive feature elimination (RFE) selects features by recursively considering smaller and smaller sets of features. For our project, we used decision tree regressor as the estimator for RFE. (3) Embedded/Intrinsic method: It uses tree-based models like decision tree or random forest to identify the feature importance. For instance, random forest regressor has an in-build method to select the features based on its importance in predicting the target variable.

Ridge regression analyzes multiple regression data that are burdened with multicollinearity. Wherever multicollinearity is present, the least squares estimates are unbiased, but their variances are large. This leads to huge variation from the original/real value. By adding a degree of bias to the regression estimates, the standard errors are reduced considerably. There is a shrinkage parameter ‘lambda’. This is said to be applying L2 penalty to the original RSS function. Equation (1) shows the mathematical formation of the ridge regularization, where λ stands for shrinkage parameter, and RSS means residual sum of squares.

$$L_{\text{ridge}}(\beta) = \sum_{i=1}^n (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^m \beta_j^2 = \text{RSS} + \lambda \|\beta\|^2 \quad (1)$$

Lasso regression uses shrinkage. Shrinkage is basically where data points are shrunk toward a central point. Lasso is usually applied on models that contain high levels of multicollinearity. The additional term after RSS is called the shrinkage penalty or L1 norm. Equation (2) shows the mathematical formulation of lasso regularization, where λ stands for shrinkage parameter.

$$L_{\text{lasso}}(\beta) = \sum_{i=1}^n (y_i - x_i'\beta)^2 + \lambda \sum_{j=1}^m |\beta_j| = \text{RSS} + \lambda \|\beta\| \quad (2)$$

ElasticNet linear regression uses the penalties from both the lasso and ridge techniques to regularize regression models. ElasticNet combines both the regularization techniques (i.e., Lasso and Ridge) by learning from their deficits, and thus, the regularization of statistical models is improved. Equation (3) shows the mathematical formulation of ElasticNet regularization.

$$L_{\text{lasso}}(\beta) = \sum_{i=1}^n (y_i - x_i'\beta)^2 / 2n + \lambda((1 - \alpha)) / 2 \sum_{j=1}^m |\beta_j^2| + \alpha \sum_{j=1}^m |\beta_j| \quad (3)$$

Random forest can be defined as an ensemble technique that can perform classification and regression tasks via decision trees and bootstrap aggregation or bagging. Gradient boosting can be defined as grouping together multiple weak machine

learning models to in turn get a stronger model. Usually, decision trees are used in the ensemble of weak models. Light gradient boosting or LightGBM is a gradient boosting framework which is fast and highly efficient. It is based on decision tree algorithm. LightGBM is capable of handling large-scale data. It converges faster than gradient boosting method. XGBoost stands for extreme gradient boosting. It is an advanced version of gradient boosting. XGBoost learns from its mistakes and contains large number of hyperparameters for fine-tuning.

4 Dataset Description

The dataset used in this project/paper is the Ames housing dataset from Kaggle [5]. The dataset was originally used for house price prediction using advanced regression techniques in Kaggle competition. The motivation for choosing this dataset comes from the fact that it is an extended version of the Boston housing dataset and consists of 79 explanatory variables. It consists of both numerical and categorical variables. The target variable is the Sale Price which represents the actual cost of the property/house given the independent variables.

5 Experimental Setup and Analysis

We have started the experiment with pre-processing of the data. The experiments are carried out on the Google Colab platform, where the models are developed and tested using Keras, TensorFlow, NumPy, matplotlib libraries. On the univariate time series dataset, we implemented 6 distinct models and compared their results with each other.

Dataset was taken from the Kaggle Website [5]. We visualized the numerical attributes via scatter plots and categorical attributes with box plots. Figure 1 shows the scatter plots for the 36 numerical attributes. It can be seen that certain attributes contain outliers which are ought to be removed after the outlier detection process. Next step was carrying out the imputation process. For normally distributed categorical variables, missing values with filled with 'none'. For example, a missing entry in 'Mas Vnr Type' practically meant that veneer is not present for that property. Two attributes: 'Garage Qual' and 'Garage Cond' were imputed with the maximum occurrence of a particular category in these columns. All numerical attributes were imputed with the median of the values in those columns. After this process, the resultant shape of the dataset was (183,982). Following this step, it was necessary to look out the distribution of the target variable, i.e., the 'Sale Price' attribute. Figure 2 shows distribution of the target variable. It is seen that the variable right skewed (positively skewed) and has the skewness measure of 1.297067. After log transformation, it is transformed close enough to a Gaussian or normal distribution. Figure 3 shows the log-transformed variable and has the skewness measure of -0.269862 .

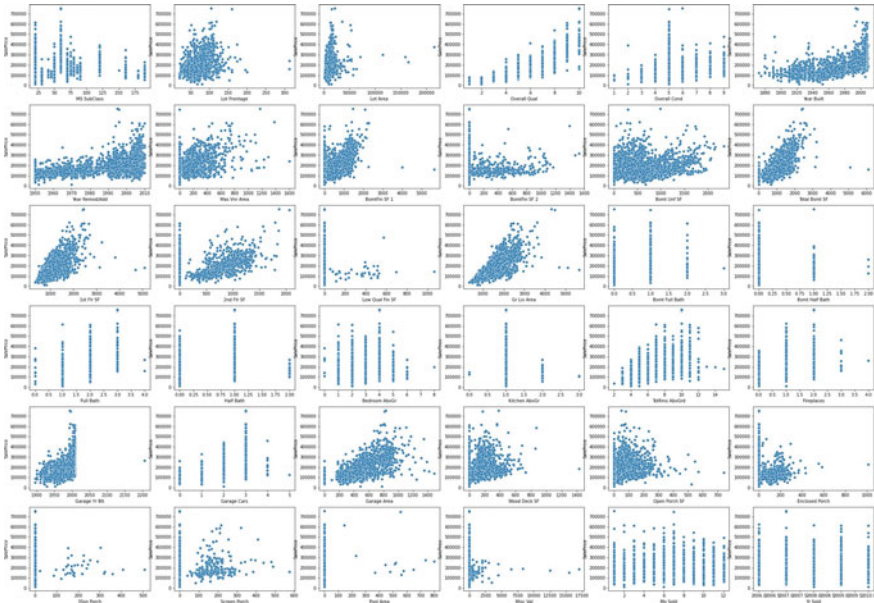


Fig. 1 Numerical attributes

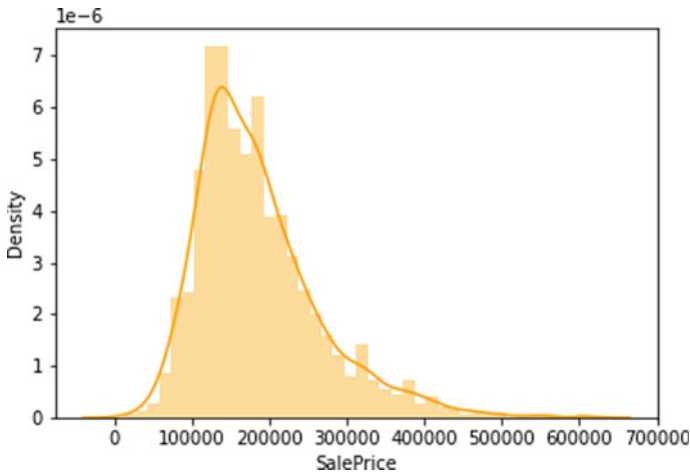


Fig. 2 Distribution of target variable

We applied `get dummies()` method in order to one-hot encode the categorical variables. After splitting the dataset into train and test set with test size = 0.3, we applied three feature selection methods, namely feature-based method, wrapper-based method, and intrinsic or embedded method. Using feature-based method, we extracted the most important features using Pearson correlation function. For the

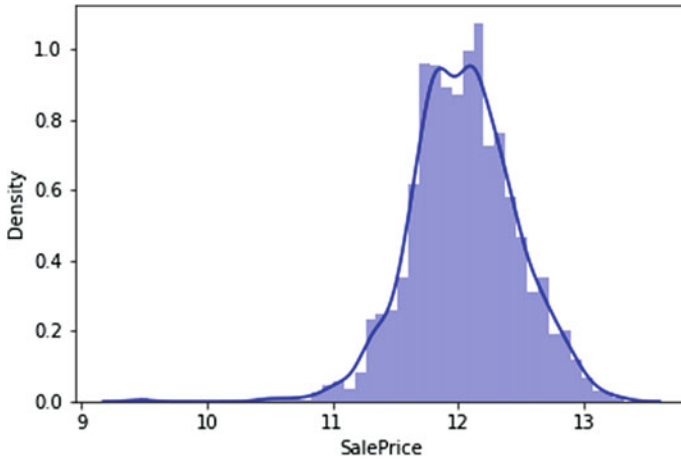


Fig. 3 After log transformation

wrapper-based method, we implemented recursive feature elimination (RFE) method which would recursively select half the number of total features, i.e., 138 features that best defines the target variable. The features were selected identified using Rank 1 and with support marked as 'true'. We used decision tree regressor as the estimator for RFE() function. Lastly, we implemented embedded/intrinsic method using tree-based model random forest regressor. The model was found to select 16 most important features out of the rest. Figure 4 shows a plot of feature importance method applied on the RF model. It can be observed that the feature 'overall Qual' strongly determines the Sale Price of the house. It defines the rating given to the house such as excellent, good, and poor, and these categories are known to affect price of the house. For all the methods, we assigned the parameter max features = 0 when looking for the best fit.

(1) Linear Regression with Ridge Regularization

Upon trying different values of the hyperparameter alpha, best value of alpha was found to be 3.0. Ridge regularization is also called L2 penalty which prevents the model from overfitting and thus makes the model a good fit for the data. Figure 5 shows the plot of residuals vs prediction for the test set. It can be analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is a good fit for our data. Figure 5 shows the coefficients of the top 20 variables among which the first 10 represent attributes with the highest coefficients, and the last 10 signify the lowest coefficients.

(2) Linear Regression with Lasso Regularization

Upon trying different values of the hyperparameter alpha, best value of alpha was found to be 0.003. Lasso regularization is also called L1 penalty which prevents the model from overfitting and thus makes the model a good fit for the data. Figure 5 shows the plot of residuals vs prediction for the test set. It can be analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is a good fit for our

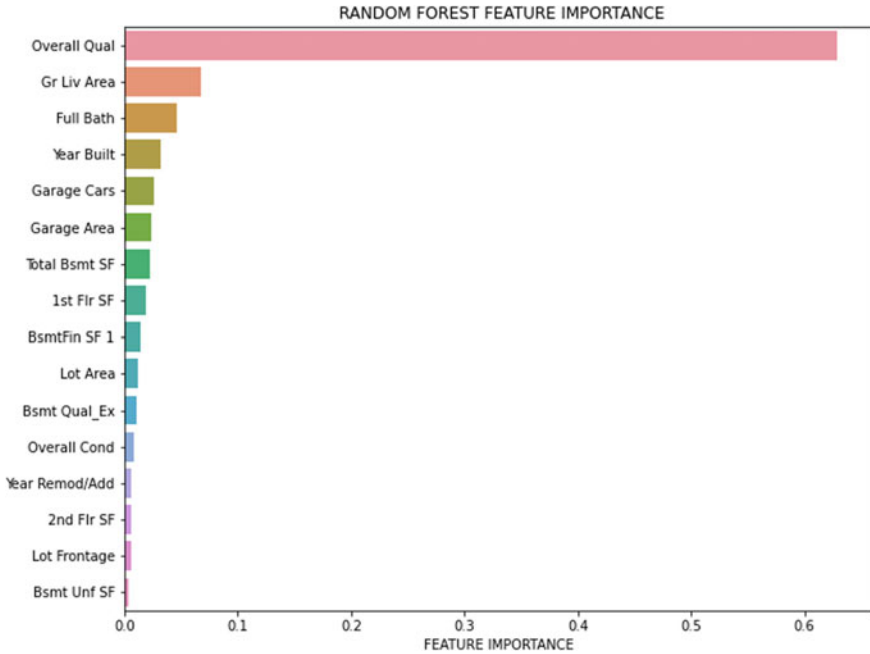


Fig. 4 Feature relevance

data. Figure 5 shows the coefficients of the top 20 variables among which first 10 represents attributes with highest coefficients, and last 10 signifies lowest coefficients or least relevance.

(3) Linear Regression with ElasticNet Regularization

Upon trying different values of the hyperparameter alpha, best value of alpha was found to be 0.0003 and best value of l1 ratio turned out to be 0.95. ElasticNet regularization is a combination of L1 and L2 penalty which prevents the model from overfitting and thus make model a good fit for the data. Figure 6a shows the plot of residuals vs prediction for the test set. It can analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is the good fit for our data. Figure 6b shows the coefficients of the top 20 variables among which first 10 represents attributes with highest coefficients, and last 10 signifies lowest coefficients.

(4) Random Forest Regressor

We experimented the model with the n estimators = 600 which indicated the number of trees to be considered while training the model. Figure 8 shows the plot of residuals versus prediction for the test set. It can analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is the good fit for our data. Evaluation was carried out using repeated K -fold on both test and train set with n splits = 10 and n repeats = 5. The actual value of first entry in test set was 254,900.0000000001,

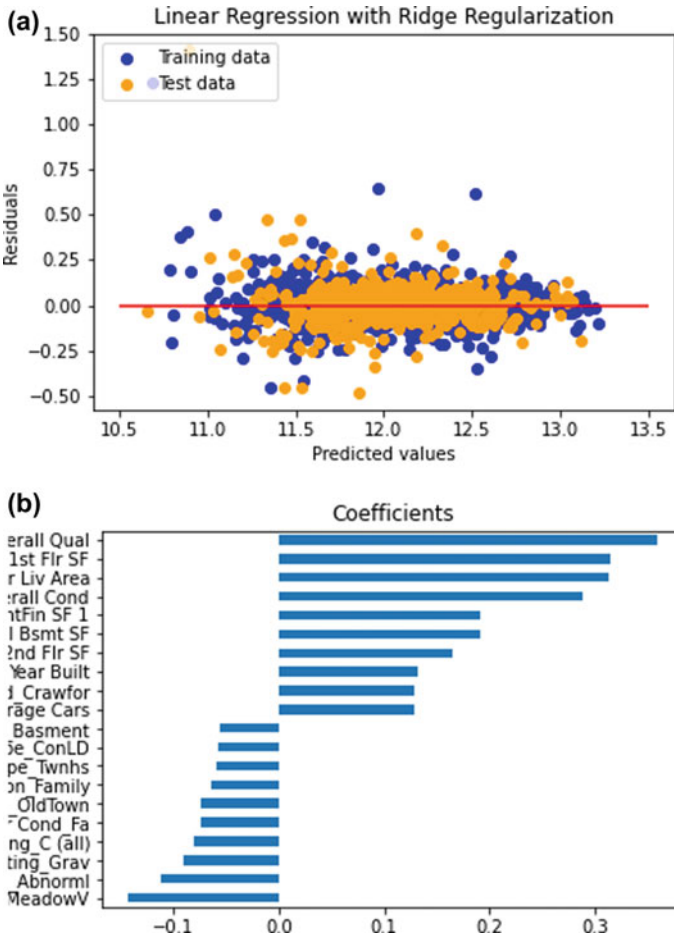


Fig. 5 Residuals (linear regression with ridge regularization)

while the corresponding predicted value turns out to be 239,590.8998588072 with random forest regressor which is quite close to the actual value. This step requires inverse log transformation.

(5) Gradient Boosting Regressor

We experimented the model with the n estimators = 500 which indicates the number of boosting stages to perform. Figure 9 shows the plot of residuals versus prediction for the test set. It can be analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is a good fit for our data. Evaluation was carried out using repeated K -fold on both test and train set with n splits = 10 and n repeats = 5. The actual value of the first entry in the test set was 254,900.0000000001, while the

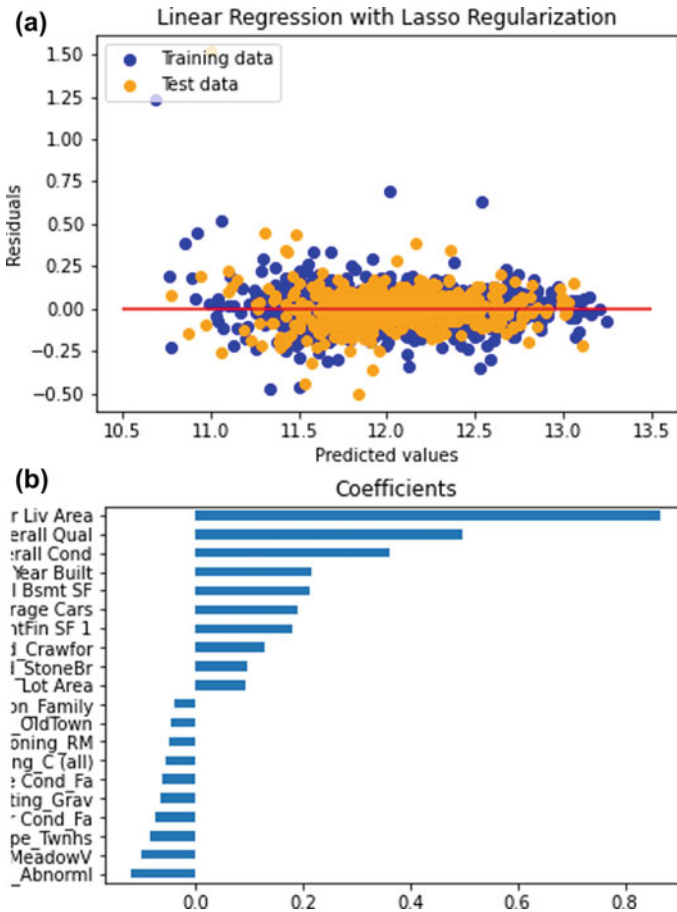


Fig. 6 **a** Residuals (linear regression with Lasso regularization), **b** coefficients (linear regression with Lasso regularization)

corresponding predicted value turns out to be 240,095.55266364999 with random forest regressor which is quite close to the actual value. This step requires inverse log transformation.

(6) Light Gradient Boosting Regressor and XGBoost—Extreme Gradient Boosting

We experimented the model with the n estimators = 500 which indicates the number of boosting stages to perform. Figure 10 shows the plot of residuals versus prediction for the test set. It can analyzed from the plot that as the residuals are near to the origin, i.e., zero, the model is the good fit for our data. Evaluation was carried out using repeated K -fold on both test and train set with n splits = 10 and n repeats = 5. The actual value of first entry in test set was 254,900.0000000001, while the corresponding predicted value turns out to be 242,191.26291967864 with random

forest regressor which is quite close to the actual value. This step requires inverse log transformation XGBoost has the best performance of all the models mentioned. We can infer from Fig. 11 that as the house condition moves towards getting excellent, the value price of that place indeed increases.

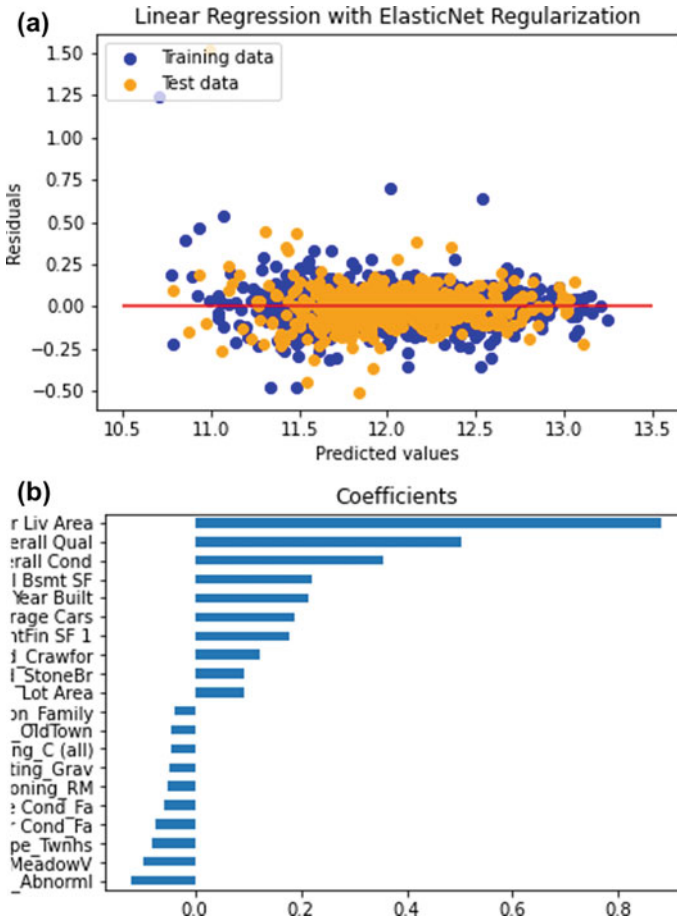


Fig. 7 a Residuals (linear regression with ElasticNet regularization), **b** coefficients (linear regression with ElasticNet regularization)

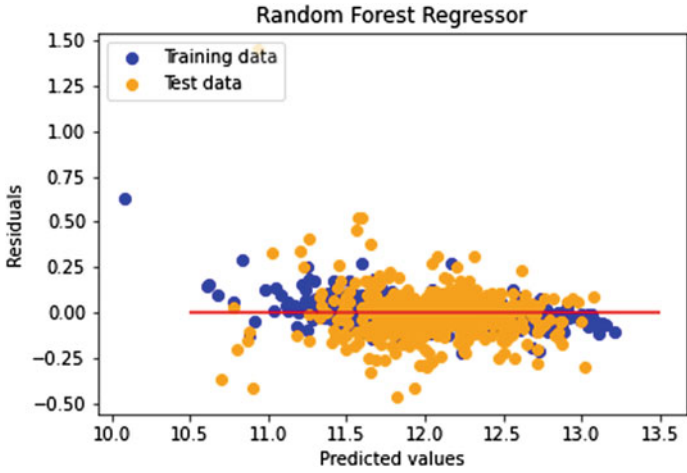


Fig. 8 Residuals (linear regression with random forest regressor)

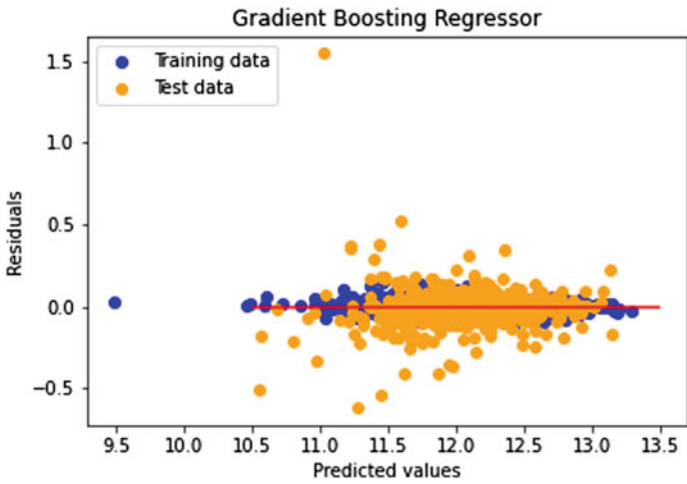


Fig. 9 Residuals (linear regression with gradient boosting regressor)

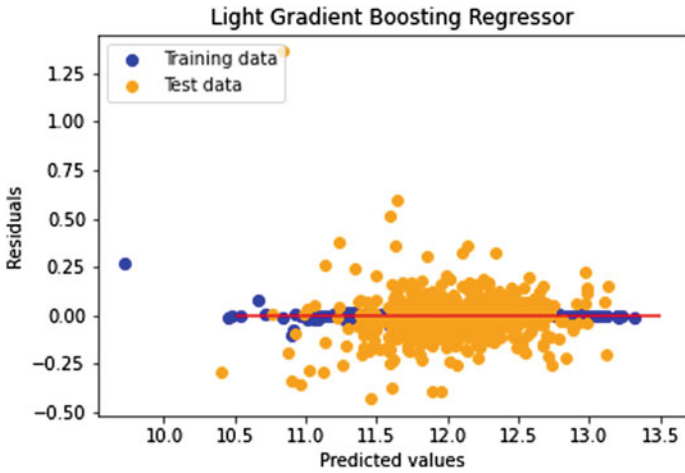


Fig. 10 Residuals (linear regression with XGBoost)

Table 1 Evaluation performed on the validation set

Model	RMSE	MAE	r2gcore
RidgeCV	0.126140	-0.085	0.913816
LassoCV	0.127350	-0.080	0.917159
ElasticNetCV	0.128725	-0.081	0.917537
Random forest regressor	0.135762	-0.144	0.903091
Gradient boosting regressor	0.121118	-0.145	0.915101
LightGBM	0.128655	-0.144	0.909732
XGBoost	0.128740	-0.148	0.921972

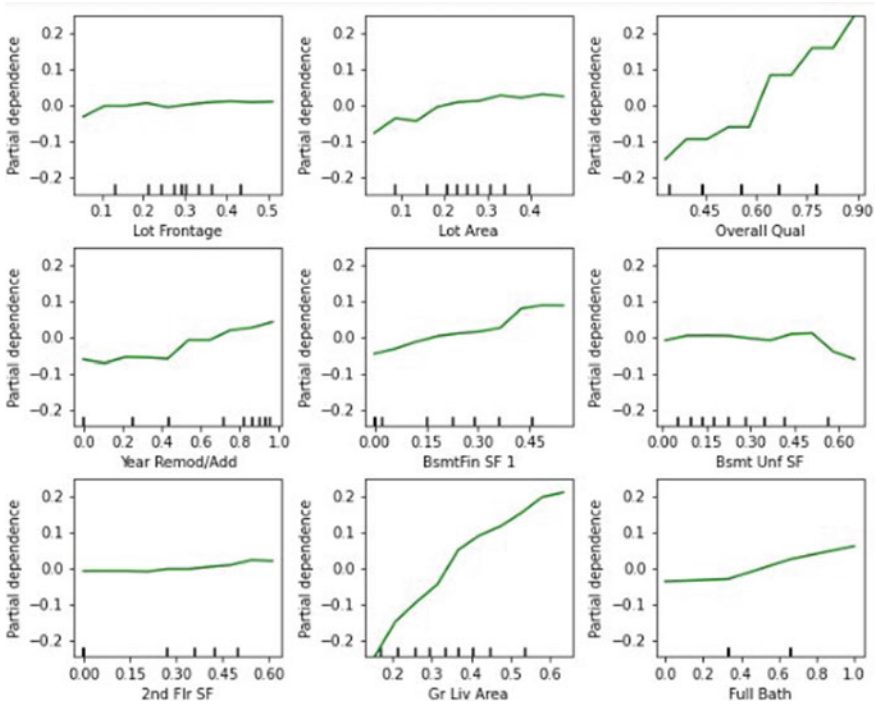


Fig. 11 Partial dependence (PD) plots

References

1. Kuvalekar A, Manchewar S, Mahadik S, Jawale S (2020) House price forecasting using machine learning. In: ICAST. <https://doi.org/10.2139/ssrn.3565512>
2. Lu S, Li Z, Qin, Yang X, Goh RSM (2017) A hybrid regression technique for house prices prediction. In: IEEM, pp 319–323. <https://doi.org/10.1109/IEEM.2017.8289904>
3. Truong Q, Nguyen M, Dang H, Mei B (2020) Housing price prediction via improved machine learning techniques. *Procedia Comput Sci* 174:433–442, ISSN 1877–0509
4. Tulio Ribeiro CGM, Singh S (2016) Model-agnostic interpretability of machine learning. [Pre-print: arxiv 1606.05386V](https://arxiv.org/abs/1606.05386)
5. Dataset Source: <https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data>
6. Sotiris Kotsiantis PEP, Kanellopoulos D (2006) Data preprocessing using supervised learning. *Int J Computer Sci* 1(1), ISSN: 1306-4428
7. Abbasi S (2020) Advanced regression techniques based housing price prediction model. <https://doi.org/10.13140/RG.2.2.18572.87684>

Product Integrity Maintenance and Counterfeit Avoidance System Based on Blockchain



Sagar Ramesh Pujar, Girish R. Deshpande, S. T. Naitik,
Raghavendra Vijay Pail, and B. Naveenkumar

1 Introduction

Since the evident growth of globalization, many big organizations follow a globalized approach for manufacturing in light of reducing costs required for this purpose. This transition, however, comes at a cost as rendering a general model of quality control becomes complicated due to a multitude of social and political factors. With avid globalization, the supply chain also becomes more complex and thus becomes a contributing factor to the complicated nature of quality control. This situation, thus, leaves the door open for an increase in counterfeiting. Counterfeiting is not restricted to a particular industry sector but rather affects a wide array of industries and it impacts the brand values of organizations that manufacture the genuine products and, in extension, impacts the global economy.

Many studies that have observed the growth of counterfeiting have observed a sound increase in sales of these products; thus, it is quite observable that counterfeiting has become a true threat to many organizations and has continued to escalate over the years. Based on the report on the enforcement of Intellectual Property Rights

S. R. Pujar (✉) · G. R. Deshpande · S. T. Naitik · R. V. Pail
KLS, Gogte Institute of Technology, Belagavi, Karnataka, India
e-mail: srpumar@git.edu

G. R. Deshpande
e-mail: grdeshpande@git.edu

S. T. Naitik
e-mail: ntsuryavanshi@git.edu

R. V. Pail
e-mail: rvpatil@git.edu

B. Naveenkumar
Sahyadri College of Engineering and Management, Mangalore, Karnataka, India
e-mail: naveen.aptra@sahyadri.edu.in

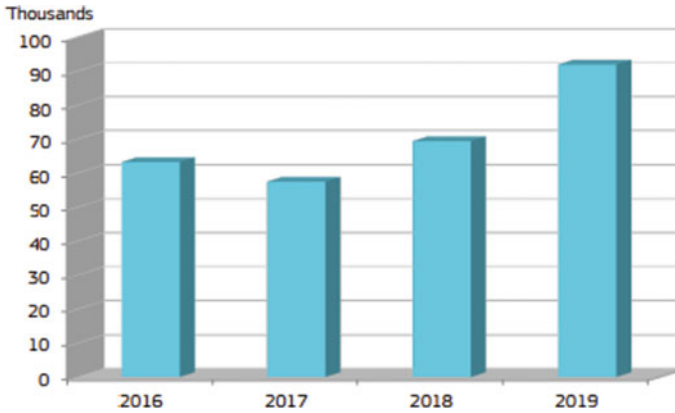


Fig. 1 Number of cases registered across the years by the EU customs [1]

by EU customs [1], it was observed that the total number of cases of counterfeiting increased by 32% in 2019 in the European Union. As seen in Fig. 1, there is a sharp increase in the number of counterfeit products being produced every year, and thus, it can be established that counterfeiting is a serious and escalating problem that is impacting the global economy.

Counterfeiting has an impending impact on various companies, governments and, in extension, to society as a whole. The direct impact of counterfeiting is observed by the companies that are victim to this who face a loss in sales and brand value. However, counterfeiting also impacts customers who cannot quite realize the difference between these products and genuine products and this can result in them having a poor quality product. Furthermore, counterfeit products undermine the efforts of Intellectual Property Rights and Copyright standards, thereby directly affecting the capabilities of governments in enforcing these standards across industries.

The inherent immutable nature of Blockchain along with traceability could be considered as a possible technological solution to solve the counterfeiting problem. By using Blockchain with barcode-based authentication, a precise and robust product authentication system can be created. The decentralized nature of Blockchain further improves the robustness of the system and also helps in maintaining trust across the supply chain. The proposed solution uses Blockchain as a solution to address the issue of counterfeiting by providing a product authentication system based on this novel technology.

2 Literature Review

Product authentication is a comprehensive approach to tackle counterfeiting by allowing verification of genuine products from counterfeit products. Although there

are many approaches that address this problem, secure authentication still stands as a recurring problem for physical products. There has been previous work that provides a system for product authentication. Turcu et al. [2] proposes the use of an Radio Frequency Identification (RFID)-based integrated system approach to solve this issue by providing a low-cost solution for rendering traceability and a software-based solution for controlled distribution across the supply chain. Rotunno et al. [3] provided a clear analysis about the impact of traceability systems in the context, both positive and negative, by considering the pharmaceutical supply chain. The observations made in [3] provide insights into the impact of having traceability systems on the operations side of the supply chain. Apart from these, [4] illustrated the Implementation of Serialization and Traceability in a Pharmaceutical Packaging Company. Shi et al. [5] illustrated the design and implementation of a secure traceability system for supply chains that use an RFID integrated system, based on the EPC global network. Brock and Sanders [6] also presented a representation of such a system along with a method for Anti-Counterfeit Barcode Labeling. Pun et al. [7] explains how Blockchain can be adapted for combating Deceptive Counterfeit in those markets where customers have intermediate distrust about products. Toyoda et al. [8] used Blockchain as the core philosophy to build a Product Ownership Management System (POMS) for RFID tagged products in order to tackle counterfeiting in the context of the post-supply chain.

A general idea shared among most of the implementations of a product authentication system is the usage of RFID-based systems. Various implementations have previously been done to authenticate RFID tagged products. A general review of RFID-based product authentication was presented in Lehtonen et al. [9]. A general approach based on what the product is the object-specific features-based authentication. This provides a certain genuinity to the product and makes the cloning of products more difficult. Object-specific features such as physical or chemical features, called as *unique product identifiers*, are used for creating tags for different products, and these tags are used for authentication [10]. The RFID tags store a signature value based on the unique product identifier, unique tag identifier, signature method and a private key [9]. An individual who wants to verify the authenticity of the product can thus use the public validation key to do so. Another approach for an RFID-based authentication system is tag authentication. In this approach, rather than the assumption of object-specific features being hard to clone, the focus is on security features that are hard to clone. In this scenario, the reader device just has to verify whether the tag has knowledge about a certain secret key. Such a system generally uses authentication protocols based on cryptographic primitives such as hashing or symmetric key operations [9]. Location-based authentication is another approach for product authentication. Such type of authentication has a targeted approach, restricted to a certain size of the environment. Therefore, the outcome of such an approach does not prevent cloning of products on a large scale but it avoids cloning in the limited scope. This approach is sometimes referred to as track and trace-based plausibility check [9]. Serialization is another way of combating counterfeiting. An example for this is provided in [11] where an example of a Victorian painter is considered who uses

serialization to combat counterfeiting. With a high level of certainty, it was observed that this works as a powerful anti-counterfeiting tool [9].

Although RFID-based authentication systems for products have addressed many issues related to counterfeiting, they still face a myriad of challenges. RFID-based systems rely on Electromagnetic Waves for providing functionality and object-specific features such as materials, and chemical composition can indirectly impact these waves and thus can affect the overall functionality and accuracy of such a system. Furthermore, while RFID-based systems are cost-effective, they are still more expensive as compared to general Barcode Scanners. Furthermore, implementation of RFID-based systems is generally more difficult and time consuming.

3 Blockchain as a Solution

Blockchain has become a trending term in the technical ecosystem. With the introduction of Ethereum Blockchain, Blockchain comprehensively shifted into a prevalent technology that could help tackle real-world problems. This section gives an intricate view into Blockchain, Ethereum and the concept of Decentralized Applications (DApp).

3.1 Blockchain

Blockchain is a distributed and decentralized database that is consensually shared, replicated and synchronized across multiple nodes [12]. A Blockchain generally consists of a chain of multiple records, called blocks, with each record composed of multiple transactions. These blocks are chained together by using cryptographic techniques such as hashing. Each block, apart from the transactions, consists of a cryptographic hash of the previous block, a timestamp and Merkle hash calculated over the set of transactions. A defining feature of Blockchain is its decentralized nature. By decentralization, Blockchain does not rely on a centralized source for maintenance of the database but rather is maintained by storing a copy of itself in each node which is a part of the Blockchain network. Whenever new transactions have to be added to the Blockchain, consensus algorithms are used. By these observations, it can be established that Blockchain is not owned by a central authority but rather is a distributed system that is continuously maintained and improvised by the participants in the Blockchain network. Thus, the level of trust established in Blockchain is higher than that of a centralized system as it removes the necessity of a trusted third party.

Identities in Blockchain are pseudo-anonymous in nature. Blockchain uses the concept of asymmetric key cryptography and digital signatures for this purpose. Each node in the system has a “public key,” which is publicly known to all the nodes in the network and “private key” which is secret to the node. Digital signatures are further used to ensure authenticity for transactions in the network.

3.2 *Ethereum*

Ethereum is a decentralized cryptocurrency, based on Blockchain that allows code execution. This is done by the virtue of “Smart Contracts.” A weaker form of Smart Contracts was introduced first in Bitcoin [13]. However, the functionalities provided by these Smart Contracts were extensively limited, and thus, [14] presented a novel use of Smart Contracts in the form of Ethereum. Ethereum generally consists of two types of accounts: Externally Owned Accounts (EOA) and Contract Accounts (CA). EOA are general accounts owned by users of Ethereum who manage their own balance. CA, on the other hand, are used to execute Smart Contracts and have their own storage. Smart Contracts in Ethereum are written in a programming language called Solidity which is compiled into a stack-based programming language that the CA can execute [8].

To avoid malicious code such as infinite loops, aimed at wasting computing resources and energy, Ethereum uses the concept of “gas.” Code executions that change the storage require the sender contract to spend this gas which is calculated based on the data amount and the computational steps. When the gas runs out, the state is reverted back to the original but the cost for gas is not returned to the sender [8]. Thus, having such a punishment-based system reduces malicious use of Smart Contracts in the network.

3.3 *Decentralized Applications (DApps)*

Through the extensive benefits of Blockchain, a novel type of applications, called Decentralized Applications or DApps, gained a lot of traction. Decentralized Applications are applications that are on a Blockchain, thus using the P2P computation, instead of relying on a centralized server. This prevents total control on the application by a single authority. Decentralized Applications on Ethereum execute Smart Contracts on Ethereum and are executed by consensus across the entire network. Furthermore, these applications have access to a global state as its “hard drive.” The evolution of Decentralized Applications has now reached a point that one can provision high-level graphical user interfaces for general customers to interact with their system, thus allowing the development of novel customer-facing applications.

4 System Architecture

The previous section presented a clear view of Blockchain technology along with its real-world implementations. Blockchain can be used as a solution to tackle counterfeiting as it inherently exhibits trust, thus preventing a single point of failure or reliance on a trusted third party.

The proposed solution uses Blockchain as a service to tackle counterfeiting across the supply chain. Each product is identified with a *unique product identifier* similar to [10]. The tag information for these identifiers is stored in the Blockchain. Furthermore, we consider four types of users of the system, namely Company, Distributor, Retailer and Customer. Company refers to the organization that produces the product and has rights to add the *unique product identifier* to the Blockchain. All the other types of users can query the Blockchain to verify the authenticity of a particular product.

A general limitation of Blockchain is the available storage. Assume that we store the entire product information in the Blockchain. In such a situation, since Blockchain is replicated across all the nodes in the network, a high amount of storage is required at each node to account for the great size of this information. Therefore, we use a centralized database to store product information along with the *unique product identifier*. Whenever a particular user wishes to validate a particular product, the system queries both the database and the Blockchain and achieves provenance. Therefore, each product goes through a 2-step authentication process.

By considering both these setups, we can visualize the system to be made of two major modules: Authentication Module, that moderates the entire authentication process, and a Blockchain Module, that validates the *unique product identifier* against the Blockchain.

This section presents a clearer view into the system architecture for the proposed solution, covering the finer details of each of the modules mentioned previously.

4.1 Authentication Module

When a user scans a tag present in the product, a SCAN request is initiated and sent to the Authentication Module. The *unique product identifier* is extracted in this module, and the identifier is queried to both the centralized database and the Blockchain controller. Once the Blockchain controller receives this identifier, it triggers the Blockchain Module by transferring the identifier. The Authentication Module then waits for confirmation from both the Blockchain controller and the centralized database. Based on these confirmations, the Authentication Module will confirm to the user whether a particular product is genuine or not. Figure 2 shows a clear representation of interactions of the Authentication Module with the centralized database and the Blockchain Module.

4.2 Blockchain Module

When the Blockchain Module receives the *unique product identifier*, a new transaction is requested from an EOA which is then initiated by the CA on passing the identifier as the payload. This transaction invokes the *authentication procedure* present

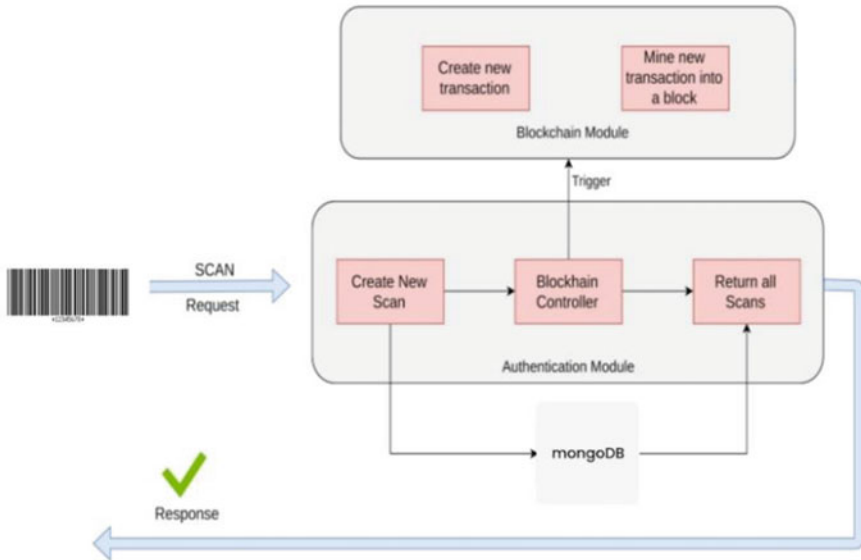


Fig. 2 Architectural design for the system

in the Smart Contract with the payload. This procedure then verifies the availability of the identifier in the Blockchain and returns *true* if the identifier is present in the Blockchain or returns *false* if the identifier is not present in the Blockchain. Once the transaction is completed, it is mined and added to the Blockchain.

5 End User Interaction

As mentioned in the previous section, there are generally four types of users in the system: Company, Distributor, Retailer and Customer. This section observes the interaction of these various types of users with the system.

1. **Company:** Company refers to the organization that produces a particular product and uses the proposed system as an anti-counterfeiting tool. The sequence of operations and the interactions of this user with the system is presented in Fig. 3. A user of type Company first has to register for the system and authenticate themselves for each session. Once authentication is successful, the user can add various products in their inventory to the system. It is worth noting that whenever the Company adds the product information, the general product information is stored in the centralized database and the *unique product identifier* is stored in the Blockchain which is then used for authentication.
2. **Distributor:** Distributors are generally the next step after manufacturing. These users are responsible for distribution of products among retailers. Similar to

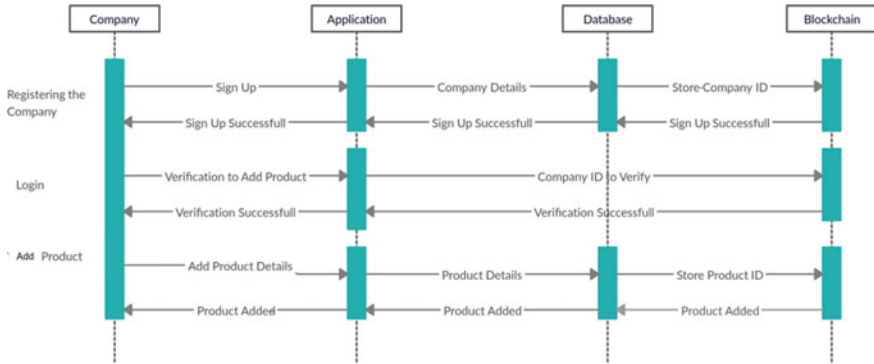


Fig. 3 Sequence of operations for the user type company

the Company, even the Distributor has to be registered to the system to use it. The Distributor uses the tag in the product and extracts the *unique product identifier* and uses this to authenticate the product (see Sect. 4). If the product is authenticated successfully, the distributor sends the product further in the supply chain. Otherwise, the product is rejected. Figure 4 shows the interactions of the Distributor with the proposed system.

3. **Retailer:** Retailers are the next step in the supply chain. These users receive the product from the Distributors and place them in the market. The interactions of the Retailer are similar to that of a Distributor. Retailer scans the tag of the product and uses the *unique product identifier* to verify the authenticity of the product. Based on the result, the Retailer decides whether to place the product in the market or not. Figure 5 represents a clear view of the interaction of Retailer with the proposed system.
4. **Customer:** Customer is the last and the most important step in a supply chain. Unlike the previous user types, Customers do not need to authenticate themselves

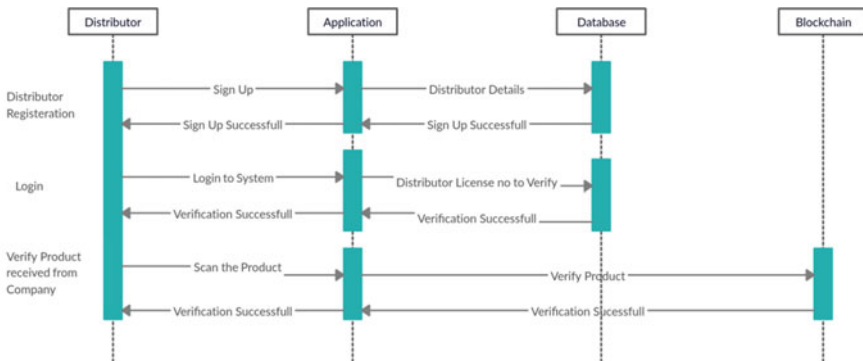


Fig. 4 Sequence of operations for the user type distributor

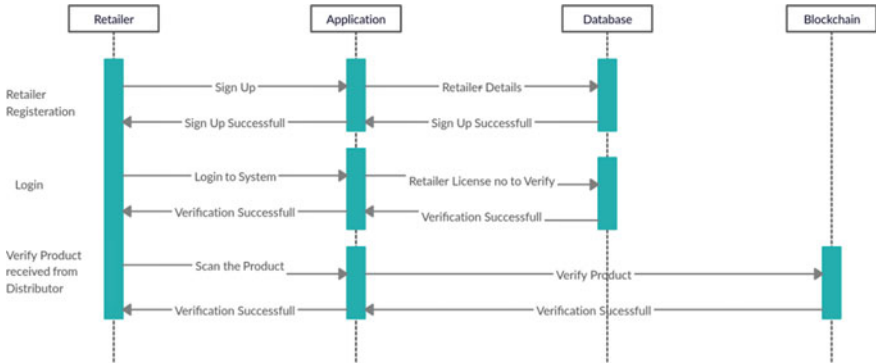


Fig. 5 Sequence of operations for the user type retailer

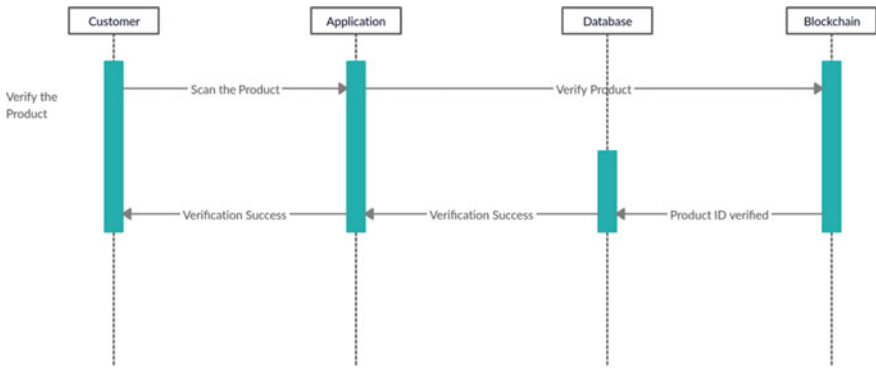


Fig. 6 Sequence of operations for the user type customer

in the system. Customers extract the unique product identifier from the product tags and verify them against the proposed system. The system then responds with a positive or a negative result for the authenticity of the product. Therefore, the customer can verify if the product is genuine or not. Figure 6 represents the interaction between the Customer and the proposed system.

6 Result and Observation

The proposed system was developed practically and checked for functionality. The visual results of the system are present in this section along with a few observations made during implementation. Most important cases are covered in order to represent the functionality of the proposed system.

1. **Addition of Products to the system:** An authenticated Company can access the form to add the products into the system. Using this form, the Company can

provide various information about the product. The representation of this portal is presented in Fig. 7.

The system then generates a *unique product identifier* in the form of a barcode and displays it as shown in Fig. 8.

2. **Verification of Products by Authenticated Users:** Once the product is added to the system, any authenticated user in the supply chain can verify the authenticity of the product. By authenticated users, we mean the users of types Company, Distributor and Retailer (see Sect. 5). The users of these types use the tag (barcode) to extract the *unique product identifier* to verify the authenticity of the product as explained in Sect. 4. Figure 9 presents the scanning process. This step is the same for all the three types of authenticated users.

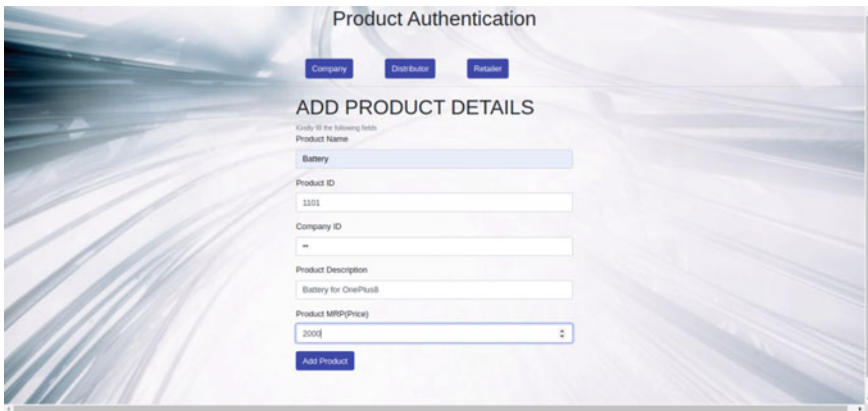


Fig. 7 Form to add products into the system

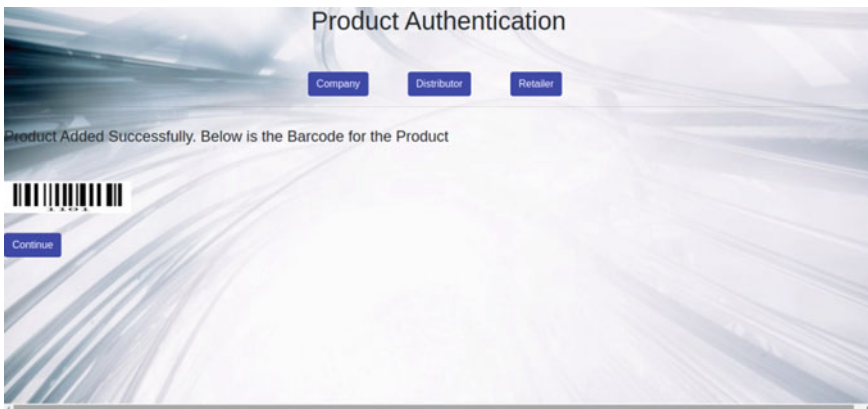


Fig. 8 Barcode generated for the product

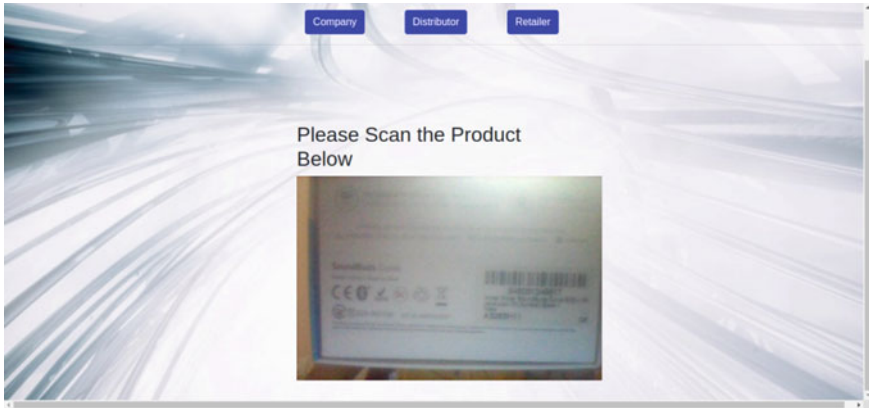


Fig. 9 Scanning process for authenticated users

- 3. **Verification of Products by Customers:** Customers are the last type of users who are presented with a different portal for authentication of the products. Since the Customer is not expected to be authenticated to use the system, they can directly use the tag (barcode) of the product to extract the *unique product identifier* and verify the authenticity of the product as presented in Fig. 10. After this verification process, the Customer can figure out if the product is genuine or not. If the product is genuine, the customer can also view information about the product that is scanned as shown in Fig. 11.

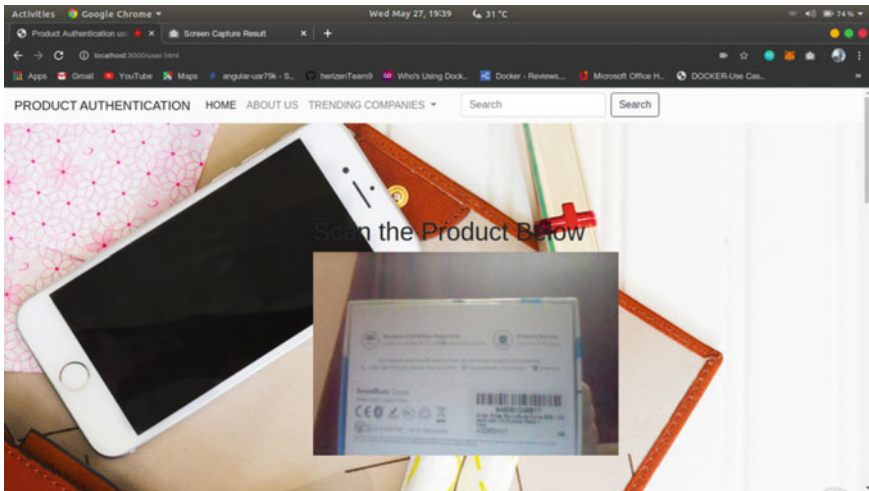


Fig. 10 Scanning process for customers

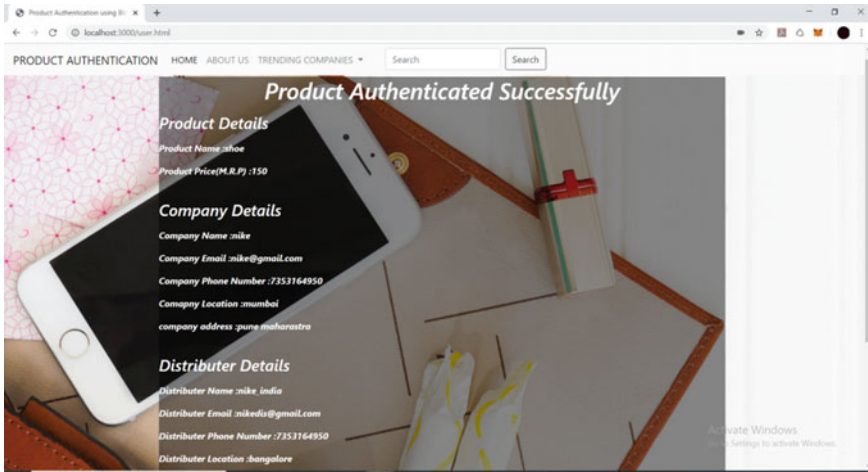


Fig. 11 Verification result for the customer

7 Conclusion

The proposed system presents a novel approach to tackling counterfeiting in the real world. By the virtue of multi-level authentication, the system provides a strong authentication and integrity protection for products. Apart from this, the system also accounts for different types of users in the supply chain and ensures that the authenticity of products is maintained across the supply chain. By providing such a robust system, organizations can expect to tackle counterfeiting to a great extent. Furthermore, another defining characteristic of the proposed system is the high level of traceability in the supply chain. Apart from the organizations, the proposed system also helps prevent Customers from falling victim to counterfeiting. The solution provided here is robust and cost-effective and has high accessibility.

References

1. Report on the EU customs enforcement of intellectual property rights-2019, Luxembourg: Publications Office of the European Union, 2020. ISBN: 978-92-76-25371-6. <https://doi.org/10.2778/92319>. Available at: [srujar@git.eduhttps://ec.europa.eu/taxation_customs/system/files/2020-12/ipr_report_2020.5464_en_04.pdf](https://ec.europa.eu/taxation_customs/system/files/2020-12/ipr_report_2020.5464_en_04.pdf)
2. Turcu CE, Turcu CO, Cerlinca M, Cerlinca T, Prodan R, Popa V (2013) An RFID-based system for product authentication. In: Department of Computer, Electronics and Automation, Stefan cel Mare University of Suceava 13, University Street, 720229–Suceava, Romania, IEEE
3. Rotunno R, Cesarotti V, Bellman A, Introna V, Benedetti M (2014) Impact of track and trace integration on pharmaceutical production systems. Int J Eng Bus Manage Spec Issue: Innov Pharm Indus 6

4. AgaraMalleesh D, Sawhney R, De Anda EM (2015) Implementation of serialization and traceability in a pharmaceutical packaging company. In: IIE Annual conference norcross
5. Shi J, Li Y, He W, Sim D (2012) SecTTS: a secure track and trace system for RFID-enabled supply chains. School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore, Singapore Institute of Manufacturing Technology, Singapore
6. Brock C, Sanders R (2007) Method and system for anti-counterfeit barcode label. Symbol Technologies Inc., Holtville, NY (US)
7. Pun H, Swaminathan JM, Hou P (2018) Blockchain adoption for combating deceptive counterfeits. Kenan Institute of Private Enterprise Research Paper No. 18–18
8. Toyoda K, Mathiopoulos PT, Sasase I, Ohtsuki T (2017) A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain. IEEE Access 5
9. Lehtonen MO, Michahelles F, Fleisch E (2007) Trust and security in RFID-based product authentication systems. IEEE Syst J 1(2):129–144. <https://doi.org/10.1109/JSYST.2007.909820>
10. Nocht Z, Staake T, Fleisch E (2006) Product specific security features based on RFID technology. In: International symposium on applications and the internet workshops (SAINTW'06), pp 4–75. <https://doi.org/10.1109/SAINT-W.2006.34>
11. Juels A (2006) RFID security and privacy: a research survey. IEEE J Sel Areas Commun 24(2):381–394. <https://doi.org/10.1109/JSAC.2005.861395>
12. National Archives and Records Administration (NARA) (2019) Blockchain White Paper
13. Nakamoto S (2009) Bitcoin: a peer-to-peer electronic cash system. Available at: <https://bitcoin.org/bitcoin.pdf>
14. Buterin V (2013) Ethereum white Paper. Available at: <https://ethereum.org/en/whitepaper/#a-next-generation-smart-contract-and-decentralized-application-platform>
15. International Counterfeit Marketing: Success without Risk (2000) Delener N, Associate Dean for Academic Affairs, The Peter J. Tobin College of Business, St. John's University, NY, Review of Business, Vol 21(No 1), Spring-Summer
16. Poonja AR, Ashish SK, Pujar SR, Kini S (2019) A study on blockchain technology and its applications. IJERT Department of Information Science and Engineering, Mangalore Institute of Technology and Engineering, Mangalore, India

Efficient Building Fire Detection Using Synergistic Interaction of Activation Functions in Artificial Neural Network



Tanushree Roy and Saikat Kumar Shome

1 Introduction

Safety from fire hazards is one of the most important parts of a city as well as a country. It is a very emergency and quick response services in the country and in constitution, it comes under the 12th schedule dealing with municipal functions [1]. The Indian insurance companies estimate that among the major losses reported in the year 2007–2008, about 45% of the claims are due to fire losses. According to another study, it is estimated that about Rs. 1000 crores are lost every year due to fire hazards [2]. The other interesting fact, from 2011 to 2012 fire and rescue services (FRSs) in Britain received 584,500 callouts, among which 53.4% were false alarms. So, not only fire hazards but also the frequency of false alarms is matter of concern.

In all the previous work, we can notice different types of approach to detect fire, smoke and also to reduce false alarm. In some recent works, two different types of alarming system for fire safety are popular—(1) traditional sensor-based fire alarm (consists thermal sensor, smoke sensors, and heat sensors) and (2) machine learning-based fire detection system (using CNN, ANN, etc.). For the first type of system, sensors require a sufficient intensity of fire to detect it correctly. The main drawback of this type of system is it needs certain amount of time for detection which may cause sufficient damage. Second type of fire detection system overcomes this drawback efficiently.

In a recent research work, by Valikhujaev and Abdusalomov proposed an algorithm that uses a dilated CNN to remove the time-consuming efforts [3]. Their method extracts some practical features to train the model automatically. Jadon and Omama established a new computer vision-based fire detection model, called FireNet, which

T. Roy · S. K. Shome (✉)

Ministry of Science and Technology, CSIR-Central Mechanical Engineering Research Institute, Government of India, Durgapur, West Bengal 713209, India
e-mail: saikatkshome@cmeri.res.in

is basically light-weight and suitable for mobiles also [4]. But the main issue for video or image-based system is it needs sufficient light to detect fire or smoke and may not be work well in dark. Kharisma and Setiyansah used three different types of sensors and a microcontroller for their system, i.e. LM35 for heat, TGS2600 for smoke, and QM6 for gas [5]. Their system has a limiting value for heat, smoke, and gas, above which it triggers alarm. This may tend to trigger false alarm frequently. They are providing SMS alert system as well as alarm, which is much appreciated. Dubey and Kumar research sends information about fire early detection in case of forest fire by detecting heat, smoke, and using a flame sensor connected to the raspberry pi microcontroller and predicted by a fully connected feed forward neural network [6]. In Park and Lee research, they introduce a fire detection framework, which composes a set of multiple machine learning algorithms as well as a fuzzy algorithm which is also adaptive and also a Direct-MQTT based on SDN is developed to solve the traffic concentration problems, which is basically a problem of traditional MQTT [7]. Evalina and Azis mainly focused on the fire incidents happens due to LPG leakage, so they described how the MQ-6 gas sensor using the ATmega8 microcontroller can be used to detect LPG gas leakage. Here is also a limit set for the gas sensor, above which a buzzer will buzz and LCD will show “leakage” as an alarm [8]. Bahrepour and Meratnia used a set of ionization, temperature, CO, and photoelectric sensor as an optimal set of sensors and they assumed that every sensor is present at the sensor node, i.e. a centralized system. For the algorithm, they showed a comparison among feed forward neural network, Naïve Bayes, and D-FLER. In their paper, it is established that Naïve Bayes is suitable for centralized system, but D-FLER is useful for distributed system [9]. In 2019, Sarwar and Bajwa worked on an IoT-based application, which warns about fire, using adaptive neuro-fuzzy inference system (ANFIS) to get minimum false alarm. For the experiment, they used microcontroller Arduino UNO R3, which is based on the atmega328p [10]. Wu and Zhang proposed a combination of transpose CNN and long short-term memory model for a real-time forecast of tunnel fire and smoke. They used numerical dataset and images to train the layers of their model [11]. Angayarkkani and Radhakrishnan established a method for forest fire detection using spatial data and artificial intelligence. They used radial basis function neural network on analysed spatial data. But it is mostly useful when fire is already spread and does not reduce false alarm [12]. Abdusalomov and Barotov approached to improve the fire detection system using classification and surveillance system. They used YOLOv3 algorithm. By this approach, they got 77.8% testing accuracy for 57 h training time as well as 82.4% testing accuracy. But this is also an image-based system which occupies more space and need sufficient light to detect fire [13]. In august 2020, Suhas and Kumar built a fire detection model using transfer learning. They have used different models (ResNet-50, InceptionV3, and Inception-ResNetV2) for feature extraction and different machine learning algorithm (decision tree, Naïve Bayes, logistic regression, and SVM) for prediction. Their result shows that features extracted by ResNet 50 and then trained by SVM algorithm shows highest accuracy (97.80%) [14].

After this survey, it can be mentioned that beside new image and video-based fire detection system, traditional sensor-based system is also improving day by day

with the use of artificial intelligence. As we have already mentioned, image-based detection is useful for a wide area with sufficient light. It is also difficult to implement everywhere. But sensor-based system is easy to implement and also works well in almost every situation. In maximum cases, a simple fuzzy logic-based algorithm has been used to implement these type of sensor-based systems, which is too generalized to detect fire. In some cases, ANN and CNN have been used to reduce false alarms using sensor data. So, we have approached an algorithm based on neural network which can learn from a numerical dataset consist of multiple sensor data. The remaining paper contains Sect. 2 methodology including a Sect. 2.1 explaining “importance of activation functions in ANN”, then three results and discussion with the description of dataset (Sect. 3.1) and explanation of outputs (Sect. 3.2) following a conclusion (Sect. 4).

2 Methodology

The main focus of this paper is to build a machine learning-based techniques which will detect fire more accurately. For this purpose, we have used an artificial neural network (ANN) which can learn and improve the results by itself. Here, sequential model, which is basically a feed forward neural network, has been used. In our work, we have tried to use different activation function in different layers (input, hidden, and output) of the neural network to get more accuracy. In a neural network, most of the time, only one activation function is used in every layer. Detailed workflow is presented in the following subsections (Fig. 1)

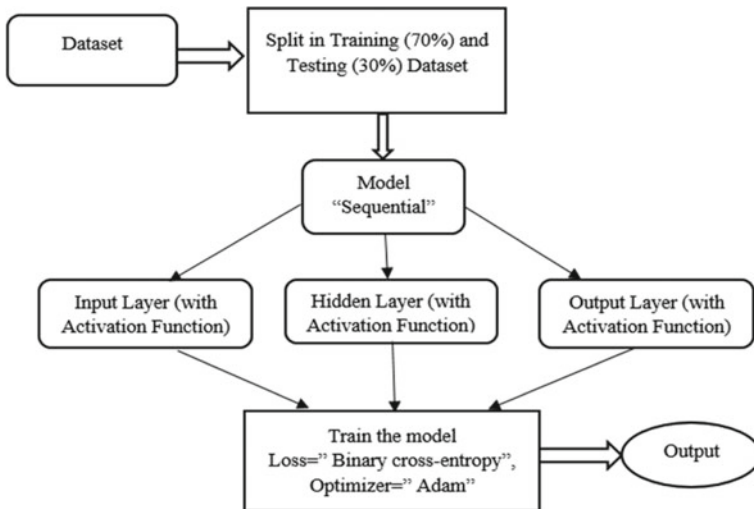


Fig. 1 Work flow diagram

2.1 Importance of Activation Functions in ANN

Accuracy and performance of a neural network mostly depend on the number of layer of the model and the activation function used in that model. Activation functions introduce nonlinearity in an ANN. Without an activation function, a neural network behaves like a linear regression model [15] (Figs. 2 and 3).

Equation of linear regression: $Y = mX + C$

Equation of ANN (without activation function): $\sum (Xi * Wi) + Bias$

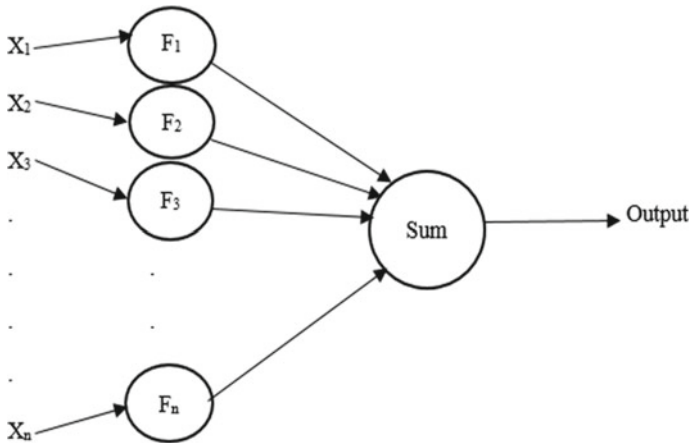


Fig. 2 ANN without activation function

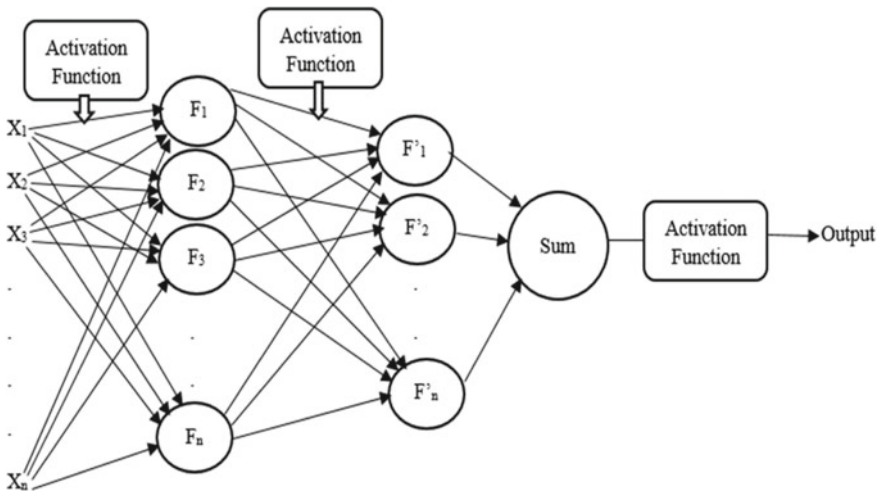


Fig. 3 ANN with activation function at each layer

Equation of ANN (with activation function): $\sum \text{Activation}((Xi * Wi) + \text{Bias})$

So, basically an activation function improves an input signal to an output signal and then that output signal fed its next layer as an input signal. It acts in stack of layers. If the output is far away from the desired, then it calculates the error and updates the weights and biases value of every neuron.

There are different types of activation functions for neural network. We have chosen four most popular of them—(1) sigmoid function, (2) linear function, (3) ReLU function, and (4) Softmax function [15].

Sigmoid function—It is an ‘S’-shaped nonlinear function. It is a common choice as an activation function for neural network. Sigmoid function gives values from 0 to 1.

Linear function—Linear function defines a straight line that passes through the origin. It is simply a form of linear regression. That is why it is not as helpful in case of neural networks as it cannot update the values of weights and biases.

ReLU function—It stands for rectified linear unit. It is also a nonlinear function and works very fast. Advantage of this function is it does not activate all the neurons at the same time. But, like linear function, it does not update the values of weights and biases because sometimes gradient of ReLU is 0.

Softmax function—It is an updated combination of multiple sigmoid functions and very useful for classification problems. Sigmoid function is mostly useful for binary classification, but Softmax is useful for both binary and multiple classification (Table 1).

Table 1 Different activation functions, their derivatives and range

Name	Function	Derivative	Range
Linear	$Y = aX$	$dY = a$	$(-\infty, \infty)$
Sigmoid	$\sigma(x) = \frac{1}{1 + e^{-x}}$	$\sigma(x) * (1 - \sigma(x))$	$(0, 1)$
ReLU	$f(x) = \max\{0, x\}$	0 if $x < 0$ 1 if $x > 0$ Undefined if $x = 0$	$(0, \infty)$
Softmax	$\sigma(x) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$	$\sigma_i(x) * (\delta_{ij} - \sigma_j(x))$ ^a	$(0, 1)$

^a δ_{ij} is Kronecker delta function

3 Results and Discussion

3.1 Dataset

We have made a numerical sensor dataset by collecting data from the NIST website <https://www.nist.gov/el/nist-report-test-fr-4016>. In that experiment, different kind of fire hazard situations including flaming and smouldering of different things (chair, mattress, cooking oil, etc.) were observed in a controlled experimental environment. O₂, CO₂, and CO gas concentrations, temperature at multiple positions, and smoke obscuration in the structure were recorded. From seven datasets (sdc01–sdc07), a total number of 5450 entries were obtained to make our dataset. 70% of that was used to train and 30% to test the model. Only six sensor data (TCB_1, TCFIRE, SMB_1, GASB_1, GASB_3 and GASB_6) were selected on the basis of their importance to feed the model. The description of the column headings and units can be found on <https://www.nist.gov/document/hookupmhl.csv>.

3.2 Explanation

In this work, we have tried to compare the results obtained from some popular activation function after feeding the model with our dataset. Also, a new method of applying different activation function at each layer, and how they behave, has been shown. To build our model, we have used an initializer called “uniform”. For the output layer, a loss function called “binary cross-entropy” [16, 17] has been chosen among some popular loss functions like “mean squared error” and “mean absolute error”. To optimize the error between the true value and predicted value, an optimizer called “Adam” has been adopted [18, 19].

For the activation function, at first we ran our code using same activation function in all the layers (input, hidden, and output) and then, we combined them and used different activation functions in different layer. For “SSL” combination, we have adopted “sigmoid” in input layer, “Softmax” in hidden layer, and “linear” in output layer. For “RLS” combination, “ReLU”, “linear”, and “sigmoid” have been used as activation function in input, hidden, and output layer, respectively. Comparison among all the results of our computation is described in Table 2.

It is clear from the above table and plots, first combination SSL, i.e. input layer activated by sigmoid, hidden layer activated by Softmax, and output layer activated by linear function, shows better testing accuracy (97.61%). Sigmoid, linear, and the second combination RLS (input layer ReLU, hidden layer linear, and output layer sigmoid) is also showing satisfactory results for both training and testing accuracy. In Fig. 4a and b, confusion matrix of SSL and RLS has been shown. Confusion matrix shows us that how many times our model predicted the desired class correctly from a testing dataset. It is clear from that “False positive” is 0 for both the combinations, and our aim was to reduce the false positive value in case of fire detection. “True

Table 2 Performance results of different methods

	Softmax-based layer	Sigmoid-based layer	Linear-based layer	Sigmoid-softmax-linear (SSL)	ReLU-linear-sigmoid (RLS)
Training accuracy (%)	75.81	97.85	94.71	97.67	98.19
Testing accuracy (%)	76.21	97.37	94.37	97.61	97.49
Mean squared error	0.2379	0.02629	0.05626	0.02385	0.02507
Mean absolute error	0.2379	0.02629	0.05626	0.02385	0.02507
Precision	0.76	0.99	0.97	1.00	1.00
Recall	1.00	0.97	0.96	0.97	0.97
F1-score	0.86	0.98	0.96	0.98	0.98

positive” and “True negative” are 1218 and 378, respectively, for combination SSL. Combination RLS shows 1197 “True positive” and 397 “True negative” value after testing the model, which is also satisfactory. Mean squared error [MSE] and mean absolute error [MAE] are also lower by at least 0.3–0.1% in these two combinations (Table 2).

Loss and accuracy curve during the training is shown in Fig. 5a and b.

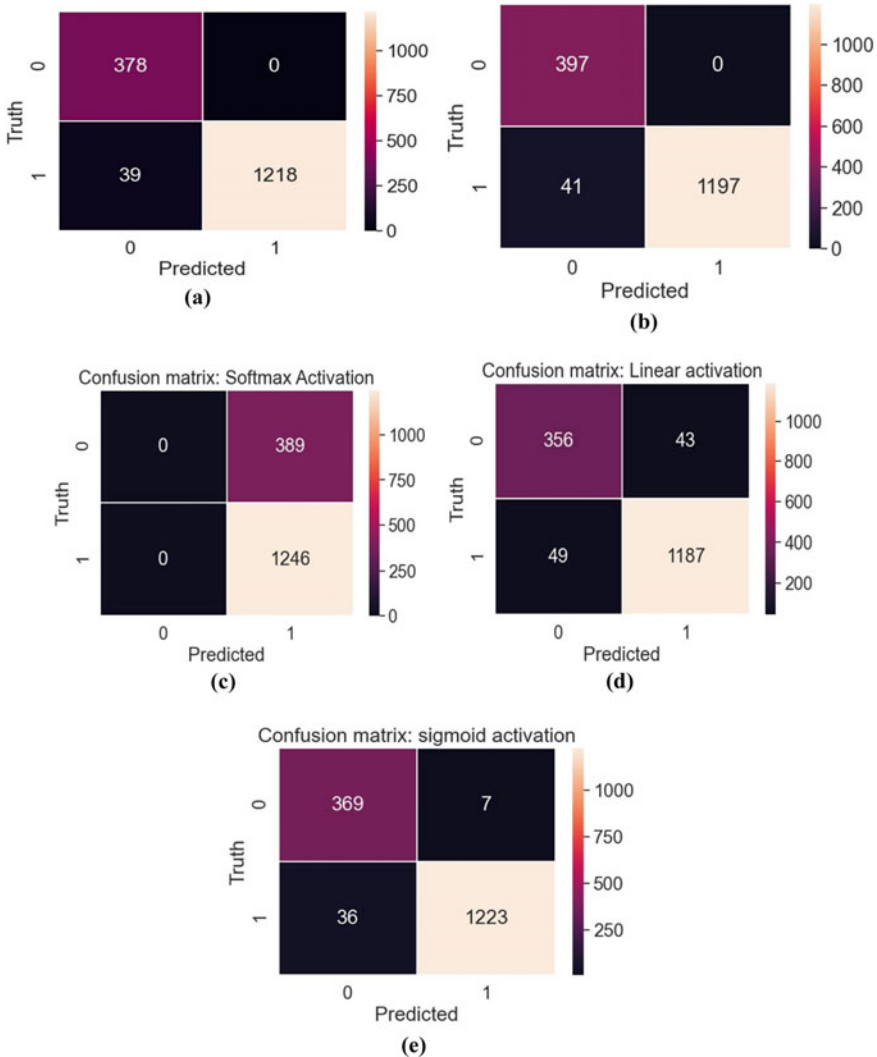
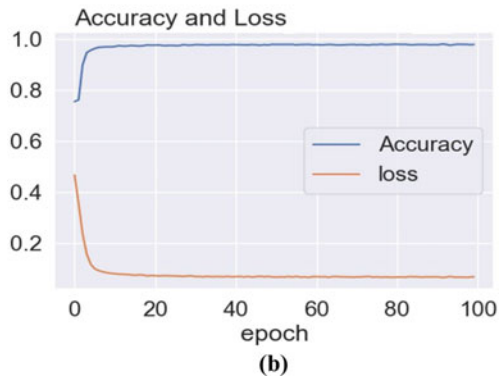
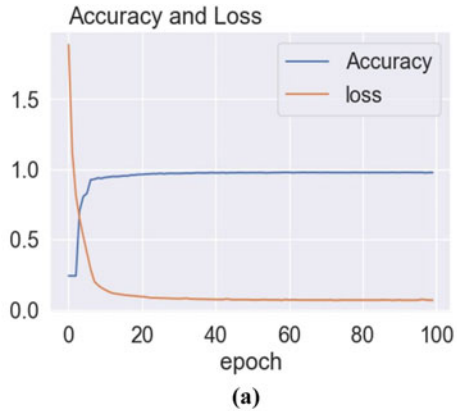


Fig. 4 a Confusion matrix obtained by using sigmoid–softmax–linear, b Confusion matrix obtained by using ReLU–linear–sigmoid, c Confusion matrix: softmax-based layer, d Confusion matrix: linear-based layer, e Confusion matrix: sigmoid-based layer

Fig. 5 a Accuracy and loss curve of SSL combination, **b** Accuracy and loss curve of RLS combination



Loss and accuracy curve of combination SSL is shown in Fig. 5a, and of combination RLS is shown in Fig. 5b. In the Fig. 5a, loss is quite high at the initial stage, but then it decreased logarithmically and also accuracy increased to almost 1 with the increasing epoch. In Fig. 5b, we can see a symmetric but opposite curve of loss and accuracy, which is also showing a satisfactory result.

4 Conclusion

In this paper, a new approach that uses selective different activation functions in different layer of an ANN to detect fire has been shown. Comparison with the existing methods and explanation of the results is mentioned accordingly. It is observed that the proposed approach reduces false positivity score as well as increases true positive value. But, it can vary with the different types of dataset. We have used a two-layered ANN with an output layer which in case of fire detection, is showing promising results and will be useful and reliable for a sensor-based fire detection system. As a future

scope of research, different results and improvement may be done by adding layers and shuffling the activation functions.

References

1. Directorate general NDRF and civil defence (Fire) ministry of home affairs, N.D (2011) Fire hazard and risk analysis in the country for revamping the fire services in the country (New Delhi)
2. Goplani S, AP (2021) To study factors governing fire safety aspect of highrise building in Ahmedabad region. IJCRT
3. Valikhujaev Y, Abdusalomov A (2020) Automatic fire and smoke detection method for surveillance systems based on dilated CNNs. Atmosphere
4. Jadon A, Omama M (2019) FireNet: a specialized lightweight fire and smoke detection model for real-time IoT applications. arXiv
5. Kharisma RS, Setiyansah A (2021) Fire early warning system using fire sensors, microcontroller, and SMS gateway. J Robot Control (JRC) 2(3)
6. Dubey V, Kumar P (2018) Forest fire detection system using IoT and artificial neural network, Springer
7. Park JH, Lee S (2019) Dependable fire detection system with multifunctional artificial intelligence framework. Sensors (MDPI)
8. Evalina N, Azis HA (2020) Implementation and design gas leakage detection system using ATMega8 microcontroller. In: IOP conference series: materials science and engineering, IOP Publishing
9. Bahrepour M, Meratnia N (2009) Use of AI techniques for residential fire detection in wireless sensor networks. AIAI-2009 Workshops Proceedings
10. Sarwar B, Bajwa IS (2019) An intelligent fire warning application using IoT and an adaptive neuro-fuzzy inference system. Sensors (MDPI)
11. Wu X, Zhang X (2021). A real-time forecast of tunnel fire based on numerical database and artificial intelligence. Springer-Verlag GmbH, Germany
12. Angayarkkani K, (2010) An intelligent system for effective forest fire detection using spatial data. Int J Comput Sci Inf Secur
13. Abdusalomov A, Baratov N (2021) An improvement of the fire detection and classification method using YOLOv3 for surveillance systems. Sensors (MDPI)
14. Suhas G, Kumar C (2020) Fire detection using deep learning. Int J Progressive Res Sci Eng
15. Sharma S, Sharma S (2020) Activation functions in neural networks. Int J Eng Appl Sci Technol
16. Sun M, Raju A (2016) Max-pooling loss training of long short-term memory networks for small-footprint keyword spotting. In: spoken language technology workshop (SLT), IEEE
17. Diederik P, Kingma JL (2015) ADAM: a method for stochastic optimization. ICLR
18. Cleary TG. An analysis of the performance of smoke alarms. In: Gaithersburg, MD 20899 USA: fire research division, Engineering Laboratory, National Institute of Standards and Technology
19. Chaurasia D, Shome SK (2021) Intelligent fire outbreak detection in wireless sensor network, Springer, Singapore

Performance Evaluation of Spectral Subtraction with VAD and Time–Frequency Filtering for Speech Enhancement



G. Thimmaraja Yadava , B. G. Nagaraja , and H. S. Jayanna 

1 Introduction

The field of speech enhancement has progressed and developed on its own throughout the years. Various algorithms were suggested during this time period in response to the growing demands of our technology oriented way of life. Spectral subtraction (SS) [1–4] is by far the most popular method in speech enhancement, possibly due to its simplicity. A well-known shortcoming of the SS algorithms is the resulting residual noise consisting of musical tones. Spectral smoothing has been proposed as a solution to the musical noise problem; however, it results in low resolution and variance [5].

Over the decades, the VAD-based noise estimation technique is employed for estimating noise in speech enhancement algorithms. Though much advancement has been done in the VAD methods, expanding VAD techniques that are precise in real-time scenarios and can work well for low signal-to-noise ratio (SNR) is still challenging. A combination of SS-VAD and linear predictive coding scheme was developed in prior work to increase the SNR and audibility features of encoded audio recordings [6]. It was observed that the resulting musical noise due to SS had an adverse effect on encoding performance. The work in [3] described a noise

G. T. Yadava (✉)

Nitte Meenakshi Institute of Technology, Bengaluru, Karnataka, India

e-mail: thimrajyadav@springer.com

B. G. Nagaraja

K. L. E. Institute of Technology, Hubballi, Karnataka, India

e-mail: nagarajbg@springer.com

H. S. Jayanna

Siddaganga Institute of Technology, Tumkur, Karnataka, India

e-mail: jayannahs@springer.com

suppression algorithm using SS. For various SNR circumstances, experimental findings revealed a 10 dB decrease in background noise (-6 – 16 dB). Also in comparison with the SS method, the proposed method achieved improved speech quality and reduced noise artifacts.

In recent studies, several speech enhancement techniques independent of VAD and SNR have been reported [7, 8]. The performance evaluation of SS and different minimum mean square error (MMSE)-based algorithms, viz. MMSE short time spectral amplitude, β -order MMSE, Log MMSE, and adaptive β -order MMSE was done in [8]. The objective, subjective, and composite objective measures showed that the adaptive β -order MMSE technique outperforms the others. To reduce the background noise, a zero-frequency filtering-based foreground speech separation front-end enhancement scheme was introduced into the automatic speech recognition system [9]. It was observed an absolute reduction of 6.24% word error rate is achieved in comparison with the previously reported spoken query system performance [10]. In [11], a modified cascaded median (MCM)-based speech enhancement algorithm was implemented using the TMS320C6416T processor. The performance of MCM-based system in terms of speech quality, memory consumption, and execution time showed superior performance than the dynamic quantile tracking and cascaded median techniques.

Another way to tackle the musical noise problem in SS is to combine an over subtraction factor with a spectral floor [5]. This approach has the shortcoming of degrading the required speech data when musical tones are adequately reduced. To overcome this, we propose the combination of the SS-time–frequency (SS-TF) filtering method. In the proposed method, a continuous recursive algorithm is employed to compute the spectral magnitude of noise. Also, a time–frequency filtering is used in place of residual noise reduction technique to condense the additive noise. The remainder of the paper is organized as follows: Sect. 2 explains the implementation of proposed SS-TF filtering method. Section 3 describes the speech quality and intelligibility evaluations and discusses the results. Finally, conclusion and future work are mentioned in Sect. 4.

2 Spectral Subtraction with Time–Frequency (SS-TF) Filtering: A Proposed Technique

In this Section, for completeness, we precisely describe the SS-VAD and implement the proposed SS-TF for speech enhancement.

2.1 Implementation of SS-VAD

The VAD in SS plays an important role by detecting the speech activity in the degraded signal [12, 13]. The degraded speech data $y(n)$ is the sum of original speech data $s(n)$ and noise model $n(n)$ can be mathematically represented as

$$y(n) = s(n) + n(n) \tag{1}$$

The Fourier transform of the above equation is

$$Y(w) = S(w) + N(w) \tag{2}$$

The block diagram of SS-VAD is shown in Fig. 1. The energy, E , the normalized linear prediction error, NLPE, and the zero-crossing rate, ZCR, were calculated for each speech segments. All these parameters were used for calculating the factor Z is given by

$$Z = E(1 - ZCR)(1 - NLPE) \tag{3}$$

The parameter Z_{max} was calculated for all the frames in speech signal and the ratio of Z/Z_{max} was used to determine whether the signal has speech activity or not. The frames without speech activity were considered for the estimation of noise spectrum. Let μ is defined as the noise spectral magnitude estimate, and then the spectral subtraction output can be written as follows:

$$|\tilde{S}_i(w)| = |X_i(w)| - \mu_i(w); \quad w = 0, 1, \dots, L - 1 \quad \text{and} \quad i = 0, 1, \dots, L - 1 \tag{4}$$

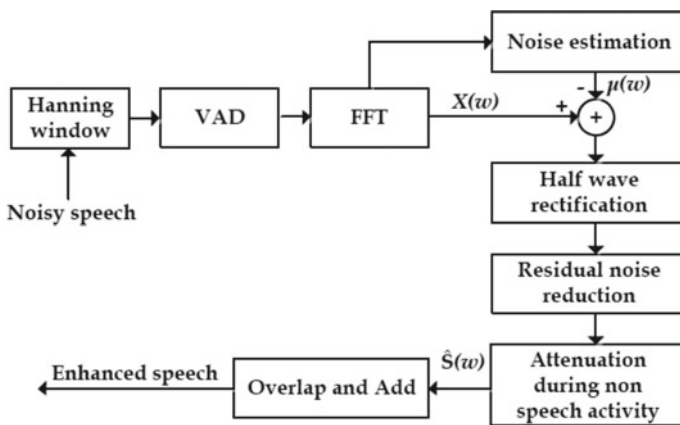


Fig. 1 Block diagram of SS-VAD

where length of the FFT is L and the number of frames are represented as M . After the process of subtraction, the differenced values are set to zero if they are negative. The residual noise reduction can be done by the mathematical modeling shown below.

$$|\tilde{S}_i(w)| = \begin{cases} |S_i(w)|; & |S_i(w)| \geq \max|N_R(w)| \\ \min\{|S_j(w)|; & |S_i(w)| < \max|N_R(w)| \end{cases} \quad (5)$$

where $\max |N_R(w)|$ is the maximum residual noise during the absence of speech activity. To reduce the noise further in the absence speech activity regions, the process of attenuation was performed.

2.2 Implementation of SS-TF

To estimate noise in the degraded speech data, we consider three important assumptions are as follows:

- The noise model or background noise is additive, slowly varying and not dependent with the clean speech signal.
- The expected values of spectral magnitude of background noise (noise model) during speech activity remain same as prior to the speech activity.
- The noise model is long-time stationary compared to the original speech signal.

The block diagram of proposed technique for background noise reduction is shown in Fig. 2.

The noisy input speech data is framed at 15 ms and windowed using Hanning window. The 50% of overlapping rate is considered with window length of 256

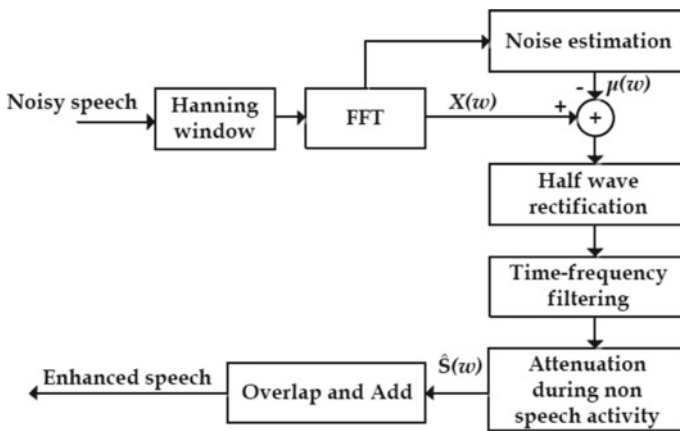


Fig. 2 Block diagram of proposed speech enhancement technique

points. For each frequency bin, the estimation of noise is computed by considering the average signal magnitude spectrum of $|Y|$ from the non-speech frames. The estimate of the spectral subtraction can be calculated as the same way presented in “Eq. (5)”. The negative magnitudes are set to zero after subtraction, and this step is called half wave rectification. In [6, 14–16], many of the SS algorithms have been implemented and showed that, there is a less suppression of musical noise when the noise model is non-stationary. To overcome this problem, we use a continuous recursive algorithm to compute the spectral magnitude of noise, and it can be mathematically represented as follows:

$$N_i(k) = (1 - \alpha)Y_i(k) + \alpha N_i(k - 1) \tag{6}$$

where $Y_i(k)$ represents the spectral magnitude of degraded speech sequence samples at frame k in i th subband. A threshold, $\beta N_i(k - 1)$ has been set to separate the speech and non-speech frames. The values of $\alpha = 0.9$ and β is in the range of 1.5–2.5. When the estimated spectral value of $Y_i(k)$ is greater than the threshold, then this can be considered as closest speech detection and recursive system which stops its execution. The TF filtering is performed to minimize the musical noise in the corrupted speech data using preceding frames and several frames following the frames of interest. To analyze the TF filtering on particular frame(s), we consider two regions shown in Fig. 3.

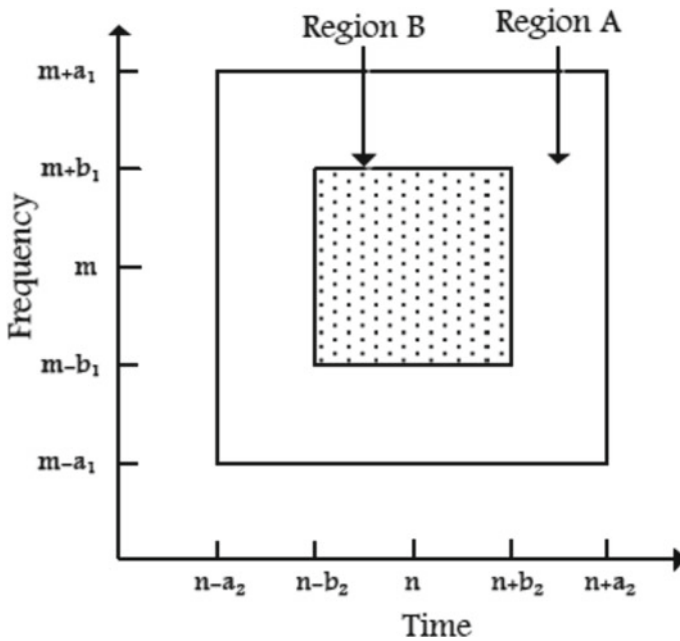


Fig. 3 Time–frequency analysis region

The values of a_1 , a_2 , b_1 , and b_2 are 7, 7, 4, and 4, respectively. If $P_B(m, n) \geq \lambda \times P_A(m, n)$, then region B contains a peak otherwise region A has the musical signal. The term λ denotes ratio of energies in both regions A and B and its value for the computation of isolation of musical signal is 5. If the region B has an isolated peak, then,

$$P_B(m, n) = \sum_{i=n-b_2}^{n+b_2} \sum_{w=m-b_1}^{m+b_1} Y_i(k) \quad (7)$$

$$P_A(m, n) = \sum_{i=n-a_2}^{n+a_2} \sum_{w=m-a_1}^{m+a_1} Y_i(k) - P_B(m, n)$$

$$Y_i(w) = 0; \quad \text{for } i = n - b_2, \dots, n + b_2 \quad \text{and} \quad w = m - b_2, \dots, m + b_2 \quad (8)$$

Otherwise, we keep the original spectral components unchanged. To estimate the noise spectrum, we multiply η with $N_i(k)$. The values of η are in the range of [1.5–2.5]. Larger the value of η , more the noise reduction and speech distortion in corrupted speech data. The smoothing operation is performed for m th frame to convert from frequency domain to time domain to reconstruct the signal.

$$P_{s,m}(w) = (1 - \lambda)P_{s,m-1}(w) + \lambda P_{s,m}(w) \quad (9)$$

where λ is ranging from [0.5–0.9]. We have used $\lambda = 0 : 75$.

3 Experimental Results and Analysis

In order to assess the effectiveness of speech enhancement, of the SS-VAD and proposed SS-TF filtering methods, we use a noisy speech corpus (NOIZEUS) [17]. In this work, the perceptual evaluation of speech quality (PESQ) and mean SNR performance metrics for speech quality evaluation [18, 19] and normalized covariance metric (NCM) performance metric for evaluating the speech intelligibility [19, 20] is considered. The assessment of speech quality and intelligibility has been conducted for different types of noises (airport, exhibition, restaurant, and station) and SNR levels (0, 5 and 10 dB). The Tables 1 and 2 give the performance evaluation of proposed and existing methods for the assessment of speech quality and intelligibility, respectively. From the experimental results, it can be observed that the proposed method has given better performance in terms PESQ and NCM for low SNR levels compared to the SS-VAD. Further, it is observed that the SS-VAD has given better performance at higher SNRs. To sum up, the proposed SS-TF filtering performed comparably to the SS-VAD for high SNR conditions and achieved a meaningful improved performance especially for low SNRs.

Table 1 PESQ values for proposed and existing methods for the assessment of speech quality

Algorithm	Types of noise	0 dB	5 dB	10 dB
SS-VAD in [6]	Airport	1.9085	2.1752	2.4814
	Exhibition	1.6571	1.9992	2.3984
	Restaurant	1.9950	2.0314	2.4362
	Station	1.6517	2.1396	2.5386
Proposed (SS-TF)	Airport	1.9501	2.1758	2.4778
	Exhibition	1.6614	2.0123	2.3811
	Restaurant	2.0900	2.0364	2.4101
	Station	1.6801	2.1294	2.5111

Table 2 NCM values for proposed and existing methods for the assessment of speech intelligibility

Algorithm	Types of noise	0 dB	5 dB	10 dB
SS-VAD in [6]	Airport	0.5641	0.6757	0.8286
	Exhibition	0.4886	0.6072	0.7736
	Restaurant	0.4963	0.6223	0.8337
	Station	0.5061	0.7528	0.8466
Proposed (SS-TF)	Airport	0.5744	0.6777	0.8247
	Exhibition	0.4887	0.6099	0.7514
	Restaurant	0.5010	0.6314	0.8247
	Station	0.5179	0.7491	0.8310

4 Conclusions

We proposed an algorithm alternative to SS-VAD under various noisy conditions. The residual noise, musical, and other types of noises were suppressed by TF filtering. Consistent speech quality and speech intelligibility performance for different noise types and SNR levels were observed. It would be also interesting to see whether SS-TF filtering would lead to better speech quality and intelligibility under negative SNRs.

References

1. Paliwal K, Wojcicki K, Schwerin B (2010) Single-channel speech enhancement using spectral subtraction in the short-time modulation domain. *Speech Commun* 52(5):450–475
2. Udrea RM, Vizireanu ND, Ciochina S (2008) An improved spectral subtraction method for speech enhancement using a perceptual weighting filter. *Digit Signal Process* 18(4):581–587
3. Gustafsson H, Nordholm SE, Claesson I (2001) Spectral subtraction using reduced delay convolution and adaptive averaging. *IEEE Trans Speech Audio Process* 9(8):799–807

4. Kumar B (2015) Spectral subtraction using modified cascaded median based noise estimation for speech enhancement. In: Proceedings of the sixth international conference on computer and communication technology 2015, pp 214–218
5. Jelinek M, Salami R (2004) Noise reduction method for wideband speech coding. In: 2004 12th European signal processing conference, IEEE, pp 1959–1962
6. Thimmaraja YG, Nagaraja BG, Jayanna HS (2021) Speech enhancement and encoding by combining SS-VAD and LPC. *Int J Speech Technol* 24(1):165–172
7. Kumar B (2021) Comparative performance evaluation of greedy algorithms for speech enhancement system. *Fluctuation Noise Lett* 20(02):2150017
8. Kumar B (2018) Comparative performance evaluation of MMSE-based speech enhancement techniques through simulation and real-time implementation. *Int J Speech Technol* 21(4):1033–1044
9. Shahnawazuddin S, Thotappa D, Dey A, Imani S, Prasanna SRM, Sinha R (2017) Improvements in IITG assamese spoken query system: background noise suppression and alternate acoustic modeling. *J Signal Proc Syst* 88(1):91–102
10. Shahnawazuddin S, Deepak KT, Sarma BD, Deka A, Prasanna SRM, Sinha R (2015) Low complexity on-line adaptation techniques in context of assamese spoken query system. *J Signal Proc Syst* 81(1):83–97
11. Kumar B (2019) Real-time performance evaluation of modified cascaded median based noise estimation for speech enhancement system. *Fluctuation Noise Lett* 18(04):1950020
12. Ramirez J, G´orriz JM, Segura JC (2007) Voice activity detection. fundamentals and speech recognition system robustness. *Robust Speech Recogn Underst* 6(9):1–22
13. Jainar SJ, Sale PL, Nagaraja BG (2020) VAD, feature extraction and modelling techniques for speaker recognition: a review. *Int J Sign Imaging Syst Eng* 12(1–2):1–18
14. Kumar B (2016) Mean-median based noise estimation method using spectral subtraction for speech enhancement technique. *Indian J Sci Technol* 9(35)
15. Tan Z-H, Dehak N et al (2020) rVAD: An unsupervised segment-based robust voice activity detection method. *Comput Speech Lang* 59:1–21
16. Yadava GT, Jayanna HS (2019) Speech enhancement by combining spectral subtraction and minimum mean square error-spectrum power estimator based on zero crossing. *Int J Speech Technol* 22(3):639–648
17. Hu Y, Loizou PC (2007) Subjective evaluation and comparison of speech enhancement algorithms. *Speech Commun* 49:588–601
18. Rix AW, Beerends JG, Hollier MP, Hekstra AP (2001) Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221) vol 2, pp 749–752
19. Hu Y, Loizou PC (2007) Evaluation of objective quality measures for speech enhancement. *IEEE Trans Audio Speech Lang Process* 16(1):229–238
20. Hu Y, Loizou PC (2006) Subjective comparison of speech enhancement algorithms. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, vol 1, IEEE, pp I–I
21. Yuan W (2021) Incorporating group update for speech enhancement based on convolutional gated recurrent network. *Speech Commun*
22. ITU-T Recommendation (2001) Perceptual evaluation of speech quality (PESQ): an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. *Rec. ITU-T P.* 862

A Unified Libraries for GDI Logic to Achieve Low-Power and High-Speed Circuit Design



Jebashini Ponnian, Senthil Pari, Uma Ramadass, and Ooi Chee Pun

1 Introduction

Standard cells (consist of well-known logic function) which is the rudimentary building blocks for combinational and sequential logic design. Subsequently, they have been in existence for over 20 years. The design of the actual standard cells is often considered as a challenging and significant activity in ASIC design. As technology gets shifted from one generation to next generation the standard cell library is scaled down. Standard cell for full adder using GDI is reported in [1]. In [2] modified GDI full adder standard cells are present with full swing restoration with additional transistors. Design of various 3 T XOR cells realization is demonstrated in [3]. The signal connectivity and buffer insertion procedure is elucidated in [4]. Primitive gate design using modified GDI is represented with and without full swing support in [5]. Energy-efficient full adder standard cells implemented is illustrated in [6] with addition transistor to support full swing output using modified GDI. Extensive discussion about multiplier standard cell is presented in [7]. An improved version of XOR/XNOR, Full adder cells with full swing output is reported in [8]. GDI basic cell representation and its characteristics are elucidated in [9–14].

J. Ponnian (✉)
Infrastructure University Kuala Lumpur, Hulu Langat, Malaysia
e-mail: jebashiniraj@gmail.com

S. Pari · O. C. Pun
Multimedia University, Cyberjaya, Malaysia
e-mail: c.senthilpari@mmu.edu.my

O. C. Pun
e-mail: cpooi@mmu.edu.my

U. Ramadass
Jeppiaar Institute of Technology, Kanchipuram, India

The primary contribution of this work offers the signal connectivity of GDI cell, creation of GDI library in four different ways. The former presents basic GDI cell without buffer, next presents the GDI cell creation including buffer, GDI cell creation using F1 and F2 and the latter specifies GDI cell using level restoration. All these libraries are created using Silterra 130 nm process mentor graphics Pyxis software and the parameter like rise time, fall time, delay power and dynamic power have been analysed. These four library cells are compared with the existing counterpart CMOS technology and reveal the significant improvement in terms of transistor count, delay and power.

2 GDI Library Creation

This session illustrates the basics of GDI cell representation, signal connectivity and various library cell creations.

2.1 Basics of GDI

GDI is one of the profound logic techniques that support low power design. The fundamental design is derived from CMOS inverter configuration having shorted gate input, and the diffusions are associated with inputs like variable inputs, power supply VDD or ground VSS.

This modified configuration offers minimal transistor count utilization for a design; the combinational delay-power factor can be minimized. Though this technique supports for low-power and high-speed design, output of the circuit is degraded due to threshold variation and complexity in fabrication process. The basic structure of GDI cell is demonstrated in Fig. 1.

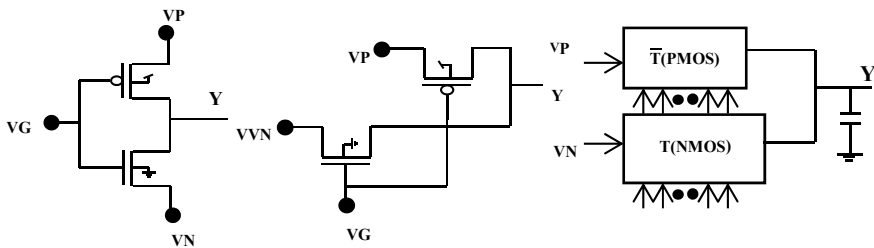


Fig. 1 Different representation of GDI cell

```

MUX Mapping Algorithm (GDI)
Algorithm for AnyGate with 2 inputs and 1 output
MUX (gate,input1,input2)
Step 1: Consider variables, // A and B are the input1 and input2 respectively.
        A - input1
        B - input2
        P, N - drain diffusion of PMOS and source diffusion of NMOS
        Y - gate output
Step 2: Assign,
        A<- Control Signal
        B<- Select Signal
Step 3: Construct 2x2 matrix with complement B and B as row, P and N as
        column. Map the truth table output of the corresponding gate in the
        constructed matrix
Step 4: Check for conditions,
        Step 4a: If (P, B) = 1 and (P,Bc) = 1, then P ← 1
                Else If (P, B) = 1 and (P,Bc) ≠ 1, then P ← B
                Else If (P, B) ≠ 1 and (P,Bc) = 1, then P ← Bc
                Else If (P, B) ≠ 1 and (P,Bc) ≠ 1, then P ← 0

                Step 4b: If (N, B) = 1 and (N,Bc) = 1, then N ← 1
                        Else If (N, B) = 1 and (N,Bc) ≠ 1, then N ← B
                        Else If (N, B) ≠ 1 and (N,Bc) = 1, then N ← Bc
                        Else If (N, B) ≠ 1 and (N,Bc) ≠ 1, then N ← 0
Step 5: Construct GDI realization with A, P and N values derived at step 4a
        and 4b respectively.
Step 6: Return Y

```

Fig. 2 Signal connectivity model for GDI Logic

2.2 Signal Connectivity of GDI Cell

The general structure of GDI cell depicts MUX-based structure. Therefore, any Boolean function GDI cell can be derived using MUX mapping algorithm which demonstrated in Fig. 2.

The algorithm defines mapping the output of particular gate to be designed in 2×2 K-map and observing the literal entities in the map along column wise. For case, if P -diffusion column is (0, 0) or (1, 1) the P -diffusion node is connected with 0 or 1. If case is (0, 1) or (1, 0), then the assigned literal along the row will be connected to P -diffusion. For elucidation, AND gate realization is shown in Fig. 3.

2.3 Library for Basic GDI Cell Without Buffer

Basic GDI cell created using MUX mapped signal connectivity is shown in Fig. 4. The patterns are created for all the primitive gates and the parameter values are tabled using mentor graphics software. This library is optimal in terms delay and power but

suffers from threshold variation and full swing problem. This can be surrogated nut inserting a buffer at each node. But, this increases the transistor count but provides full swing.

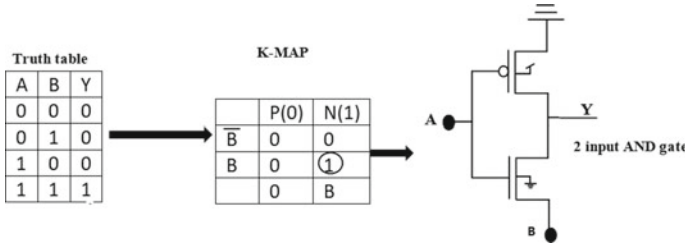


Fig. 3 2-input AND GDI gate using Mux signal connectivity

Gate	Truth Table	MUX Based Mapping	Circuit Realization In GDI	Gate	Truth Table	MUX Based Mapping	Circuit Realization In GDI																																										
AND	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>$\overline{A}B$</td> <td>0</td> </tr> <tr> <td>0</td> <td>$A\overline{B}$</td> </tr> </tbody> </table>	P(0)	N(1)	$\overline{A}B$	0	0	$A\overline{B}$		NOR	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}	
A	B	Y																																															
0	0	0																																															
0	1	0																																															
1	0	0																																															
1	1	1																																															
P(0)	N(1)																																																
$\overline{A}B$	0																																																
0	$A\overline{B}$																																																
A	B	Y																																															
0	0	1																																															
0	1	0																																															
1	0	0																																															
1	1	0																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
OR	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}		XOR	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}	
A	B	Y																																															
0	0	0																																															
0	1	1																																															
1	0	1																																															
1	1	1																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
A	B	Y																																															
0	0	0																																															
0	1	1																																															
1	0	1																																															
1	1	0																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
NAND	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}		XNOR	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}	
A	B	Y																																															
0	0	1																																															
0	1	1																																															
1	0	1																																															
1	1	0																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
A	B	Y																																															
0	0	1																																															
0	1	0																																															
1	0	0																																															
1	1	1																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
F1	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	1	1	0	0	1	1	0	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}		MUX	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>B</td> </tr> <tr> <td>0</td> <td>1</td> <td>B</td> </tr> <tr> <td>1</td> <td>0</td> <td>C</td> </tr> <tr> <td>1</td> <td>1</td> <td>C</td> </tr> </tbody> </table>	A	B	Y	0	0	B	0	1	B	1	0	C	1	1	C	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>B</td> </tr> <tr> <td>A</td> <td>C</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	B	A	C	
A	B	Y																																															
0	0	0																																															
0	1	1																																															
1	0	0																																															
1	1	0																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
A	B	Y																																															
0	0	B																																															
0	1	B																																															
1	0	C																																															
1	1	C																																															
P(0)	N(1)																																																
\overline{A}	B																																																
A	C																																																
F2	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	1	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>\overline{A}</td> <td>0</td> </tr> <tr> <td>0</td> <td>\overline{B}</td> </tr> </tbody> </table>	P(0)	N(1)	\overline{A}	0	0	\overline{B}		INV	<table border="1"> <thead> <tr> <th>A</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	Y	0	1	1	0	<table border="1"> <thead> <tr> <th>P(0)</th> <th>N(1)</th> </tr> </thead> <tbody> <tr> <td>A(0,1)</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	P(0)	N(1)	A(0,1)	1	1	0										
A	B	Y																																															
0	0	1																																															
0	1	1																																															
1	0	0																																															
1	1	1																																															
P(0)	N(1)																																																
\overline{A}	0																																																
0	\overline{B}																																																
A	Y																																																
0	1																																																
1	0																																																
P(0)	N(1)																																																
A(0,1)	1																																																
1	0																																																

Fig. 4 2-input GDI library cells using Mux signal connectivity

Gate	Truth Table	MUX Based Mapping	Circuit Realization In GDI	Gate	Truth Table	MUX Based Mapping	Circuit Realization In GDI																																														
AND	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>0</td></tr> <tr><td>B</td><td>0</td></tr> <tr><td>0</td><td>B</td></tr> </table>	P(0)	N(1)	\bar{B}	0	B	0	0	B		NOR	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>1</td></tr> <tr><td>B</td><td>0</td></tr> <tr><td>\bar{B}</td><td>0</td></tr> </table>	P(0)	N(1)	\bar{B}	1	B	0	\bar{B}	0	
		A	B	Y																																																	
0	0	0																																																			
0	1	0																																																			
1	0	0																																																			
1	1	1																																																			
P(0)	N(1)																																																				
\bar{B}	0																																																				
B	0																																																				
0	B																																																				
A	B	Y																																																			
0	0	1																																																			
0	1	0																																																			
1	0	0																																																			
1	1	0																																																			
P(0)	N(1)																																																				
\bar{B}	1																																																				
B	0																																																				
\bar{B}	0																																																				
OR	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>0</td></tr> <tr><td>B</td><td>1</td></tr> <tr><td>B</td><td>1</td></tr> </table>	P(0)	N(1)	\bar{B}	0	B	1	B	1		XOR	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>0</td></tr> <tr><td>B</td><td>1</td></tr> <tr><td>B</td><td>\bar{B}</td></tr> </table>	P(0)	N(1)	\bar{B}	0	B	1	B	\bar{B}	
		A	B	Y																																																	
0	0	0																																																			
0	1	1																																																			
1	0	1																																																			
1	1	1																																																			
P(0)	N(1)																																																				
\bar{B}	0																																																				
B	1																																																				
B	1																																																				
A	B	Y																																																			
0	0	0																																																			
0	1	1																																																			
1	0	1																																																			
1	1	0																																																			
P(0)	N(1)																																																				
\bar{B}	0																																																				
B	1																																																				
B	\bar{B}																																																				
NAND	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>1</td></tr> <tr><td>B</td><td>1</td></tr> <tr><td>1</td><td>\bar{B}</td></tr> </table>	P(0)	N(1)	\bar{B}	1	B	1	1	\bar{B}		XNOR	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>1</td></tr> <tr><td>B</td><td>0</td></tr> <tr><td>\bar{B}</td><td>B</td></tr> </table>	P(0)	N(1)	\bar{B}	1	B	0	\bar{B}	B	
		A	B	Y																																																	
0	0	1																																																			
0	1	1																																																			
1	0	1																																																			
1	1	0																																																			
P(0)	N(1)																																																				
\bar{B}	1																																																				
B	1																																																				
1	\bar{B}																																																				
A	B	Y																																																			
0	0	1																																																			
0	1	0																																																			
1	0	0																																																			
1	1	1																																																			
P(0)	N(1)																																																				
\bar{B}	1																																																				
B	0																																																				
\bar{B}	B																																																				
F1	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	Y	0	0	0	0	1	1	1	0	0	1	1	0	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>0</td></tr> <tr><td>B</td><td>1</td></tr> <tr><td>B</td><td>0</td></tr> </table>	P(0)	N(1)	\bar{B}	0	B	1	B	0		MUX	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>B</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>C</td></tr> <tr><td>1</td><td>1</td><td>C</td></tr> </table>	A	B	Y	0	0	B	0	1	0	1	0	C	1	1	C	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>B</td></tr> <tr><td>B</td><td>C</td></tr> <tr><td>B</td><td>C</td></tr> </table>	P(0)	N(1)	\bar{B}	B	B	C	B	C	
		A	B	Y																																																	
0	0	0																																																			
0	1	1																																																			
1	0	0																																																			
1	1	0																																																			
P(0)	N(1)																																																				
\bar{B}	0																																																				
B	1																																																				
B	0																																																				
A	B	Y																																																			
0	0	B																																																			
0	1	0																																																			
1	0	C																																																			
1	1	C																																																			
P(0)	N(1)																																																				
\bar{B}	B																																																				
B	C																																																				
B	C																																																				
F2	<table border="1"> <tr><th>A</th><th>B</th><th>Y</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	A	B	Y	0	0	1	0	1	1	1	0	0	1	1	1	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>\bar{B}</td><td>1</td></tr> <tr><td>B</td><td>1</td></tr> <tr><td>1</td><td>B</td></tr> </table>	P(0)	N(1)	\bar{B}	1	B	1	1	B		INV	<table border="1"> <tr><th>A</th><th>Y</th></tr> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	A	Y	0	1	1	0	<table border="1"> <tr><th>P(0)</th><th>N(1)</th></tr> <tr><td>A,(0,1)</td><td>1</td></tr> <tr><td></td><td>0</td></tr> <tr><td></td><td>1</td></tr> <tr><td></td><td>0</td></tr> </table>	P(0)	N(1)	A,(0,1)	1		0		1		0								
		A	B	Y																																																	
0	0	1																																																			
0	1	1																																																			
1	0	0																																																			
1	1	1																																																			
P(0)	N(1)																																																				
\bar{B}	1																																																				
B	1																																																				
1	B																																																				
A	Y																																																				
0	1																																																				
1	0																																																				
P(0)	N(1)																																																				
A,(0,1)	1																																																				
	0																																																				
	1																																																				
	0																																																				

Fig. 5 2-input GDI library cells using buffer

2.4 Library for Basic GDI Cell with Buffer

As predominant problem in GDI library is threshold variation which would cause the output to degrade. This adverse effect can be eliminated using level restoration circuit. One of the level restoration circuits is buffer. By including buffer at the output node of each gate, it surrogates the output swing problem with additional transistor. For larger circuit, more number of buffers are accumulated which would increases the area of IC which is the draw of utilizing this library. One of the solutions provided could be adding the buffer for every consecutive three nodes will be sufficient to pull the output of logic gate [4]. The circuit design is depicted in Fig. 5.

2.5 Library for Basic GDI Cell with F1 and F2

For GDI logic F1 and F2 are the functional component similar to NAND/NOR for CMOS logic. All primitive structure can be implemented using F1 and F2. The

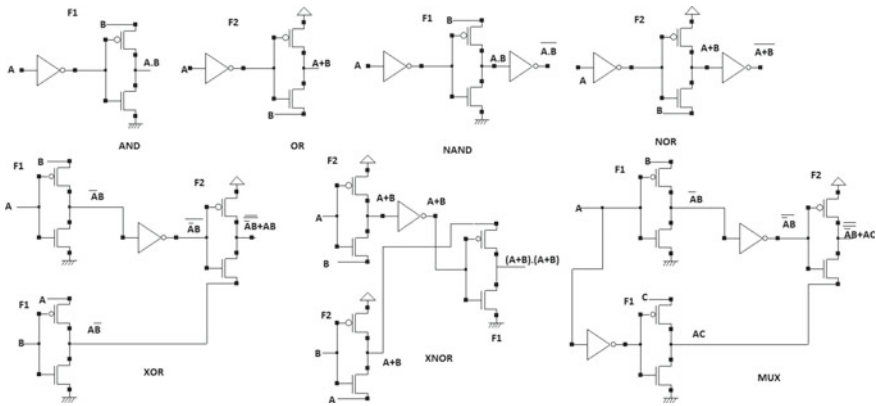


Fig. 6 2-input GDI library cells using F1 and F2

functional description of F1s \overline{AB} and for F2 is $\overline{A} + B$. Any Boolean function implemented using F1 and F2 occurs better output swing when compared to GDI cell in Fig. 4. Slight deformation is observed for certain inputs. If proper level restoration is included for F1 and F2 the circuit produces full swing output. The catalogue of primitive cells implemented only with F1 and F2 without level restoration is shown in Fig. 6.

2.6 Library for Basic GDI Cell with F1 and F2 Using Level Restoration

As elucidated above, any Boolean function can be implemented using F1 and F2 of GDI cell. But this library provides optimal solution with slight degradation. To provide full swing output level restoration circuit which is shown in Fig. 7 can be deployed. For F1 cell, an NMOS is included so for the input 01, the output is completely pulled up by NMOS keeper circuit. Similarly for F2, a PMOS is added so that for the input 10, the output is completely pulled down by PMOS keeper circuit. The catalogue of primitive cells implemented only with F1 and F2 without level restoration is shown in Fig. 8.

3 Experimentation and Results

The experimentation is done using Silterra 130 nm process mentor graphics Pyxis software and the parameter like rise time, fall time, delay power and dynamic power have been analysed. The simulation setup is defined with 0–1.2 V in steps of 0.2 V and tested for all design corners. Extensive simulation is performed to cover every

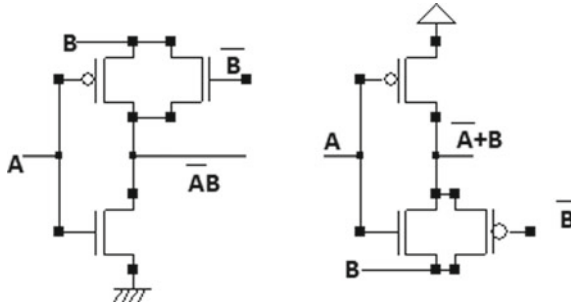


Fig. 7 Level restoration circuit for F1 and F2

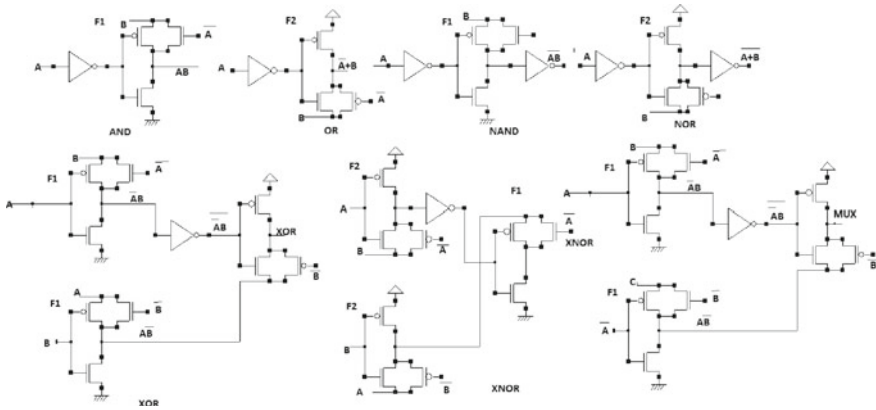


Fig. 8 2-input GDI library cells using F1 and F2 with restoration circuit

possible input patterns for all logic gates. The simulated results of 4 GDI library along with CMOS logic is shown in Tables 1, 2, 3, 4 and 5.

4 Discussion

The comparative analysis with respect to delay and PDP is shown in Fig. 9. From this study, it is observed that GDI with F1 and F2 using restoration circuit produces optimal delay and PDP in accordance with the other three libraries with additional transistor count.

For CMOS logic only for XOR and XNOR, it is slightly comparable with GDI with F1 and F2 using restoration. For GDI library listed in Table 1 utilizes less transistor count but the delay aspect is high when compared to the other three libraries.

Table 1 Simulated Input for 2-input GDI library cells using Mux signal connectivity

GDI	RS (PS)	FT (PS)	Delay (nS)	Tr	PD (PW)	Tr * Delay	PDP
AND	1.55	3.06	12.67	2	282.9	25.34	3584.343
OR	20.86	910.63	12.66	2	330.56	25.32	4184.8896
NAND	180.44	136.06	24.9	4	390.69	99.6	9728.181
NOR	137.76	148.83	24.96	4	480.07	99.84	11,982.547
XOR	180.49	29.48	28.89	4	942.84	115.56	27,238.648
XNOR	31.69	174.49	28.03	4	912.07	112.12	25,565.322
MUX	21.11	19.21	19.96	2	164.97	39.92	3292.8012
F1	173.2	19.21	14.87	2	222.88	29.74	3314.2256
F2	11.33	173.16	14.08	2	230.09	28.16	3239.6672

Table 2 Simulated Input for 2-input GDI library cells including buffer

GDI	RS (PS)	FT (PS)	Delay (nS)	Tr	PD (PW)	Tr * Delay	PDP
AND	76.19	116.02	11.67	6	284.23	70.02	3316.9641
OR	103.84	169.8	11.66	6	332.56	69.96	3877.6496
NAND	39.68	36.92	22.9	8	393.61	183.2	9013.669
NOR	44.62	20.81	22.96	8	484.07	183.68	11,114.247
XOR	40.47	120.97	24.89	8	944.82	199.12	23,516.57
XNOR	110.1	115.75	24.03	8	921.07	192.24	22,133.312
MUX	68.93	115.69	17.96	8	166.97	143.68	2998.7812
F1	37.91	169.81	12.87	6	226.88	77.22	2919.9456
F2	77.21	22.99	12.08	6	234.09	72.48	2827.8072

Table 3 Simulated Input for 2-input GDI library cells including F1 and F2

GDI	RS (PS)	FT (PS)	Delay (nS)	Tr	PD (PW)	Tr * Delay	PDP
AND_F1	136.06	180.49	10.17	4	286.23	40.68	2910.9591
OR_F2	148.83	31.69	10.66	4	336.56	42.64	3587.7296
NAND_F1	29.48	21.11	20.9	6	394.62	125.4	8247.558
NOR_F2	137.76	21.11	20.96	6	486.22	125.76	10,191.171
XOR_F1, F2	180.49	173.2	22.82	8	946.41	182.56	21,597.076
XNOR_F1, F2	31.69	40.47	22.32	8	923.71	178.56	20,617.207
MUX_F1, F2	21.11	110.1	15.96	10	166.96	159.6	2664.6816
F1	173.2	19.21	14.87	2	222.88	29.74	3314.2256
F2	11.33	173.16	14.08	2	230.09	28.16	3239.6672

Table 4 Simulated input for 2-input GDI library cells including F1 and F2 with restoration

GDI	RS (PS)	FT (PS)	Delay (nS)	Tr	PD (PW)	Tr * Delay	PDP
AND_F1	396.79	795.56	9.98	5	288.32	49.9	2877.43
OR_F2	474.82	122.83	9.99	5	338.55	49.95	3382.11
NAND_F1	134.41	65.83	18.81	7	396.72	131.67	7462.30
NOR_F1	50.72	106.33	18.98	7	488.24	132.86	9266.79
XOR_F1 + F2	60.93	62.96	19.96	17	948.44	339.32	18,930.8
XNOR_F1 + F2	245.68	780.83	19.95	17	925.72	339.15	18,468.1
MUX_F1 + F2	367.77	59.44	10.78	19	168.22	204.82	1813.41
F1	811.08	178.54	10.98	5	224.73	54.9	2467.53
F2	179.14	796.64	10.99	5	232.1	54.95	2550.77

Table 5 Simulated Input for 2-input CMOS library cells

GDI	RS (PS)	FT (PS)	Delay (nS)	Tr	PD (PW)	Tr * Delay	PDP
AND	36.92	110.1	11.79	6	360.73	70.74	4253.007
OR	20.81	68.93	16.4	6	653.21	98.4	10,712.64
NAND	120.97	37.91	30.93	4	506.98	123.72	15,680.89
NOR	115.75	77.21	31.07	4	647.49	124.28	20,117.51
XOR	115.69	173.16	19.86	8	750.22	158.88	14,899.37
XNOR	29.48	180.44	19.7	8	966.21	157.6	19,034.34
MUX	174.49	137.76	20.78	12	978.32	249.36	20,329.49
F1	19.21	180.49	28.7	8	468.85	229.6	13,456
F2	19.21	31.69	28.8	8	479.38	230.4	13,806.14

*RT = Rise Time, *FT = Fall Time, *Tr = Transistor count, *PD = Power dissipation, Tr * delay = product of delay and transistor, *PDP = Product of power and delay

5 Conclusion

This work, presents a regimented creation of four different GDI libraries to implement combinational and sequential circuit with minimal power and delay. The proposal also defines a unified connectivity model for GDI logic using MUX mapping algorithm. The experimentation is done using Silterra 130 nm process mentor graphics Pyxis software and the parameter like rise time, fall time, delay power and dynamic power. From this work, it is observed that GDI with F1 and F2 using restoration circuit produces optimal delay and PDP in accordance with the other three libraries with additional transistor count.

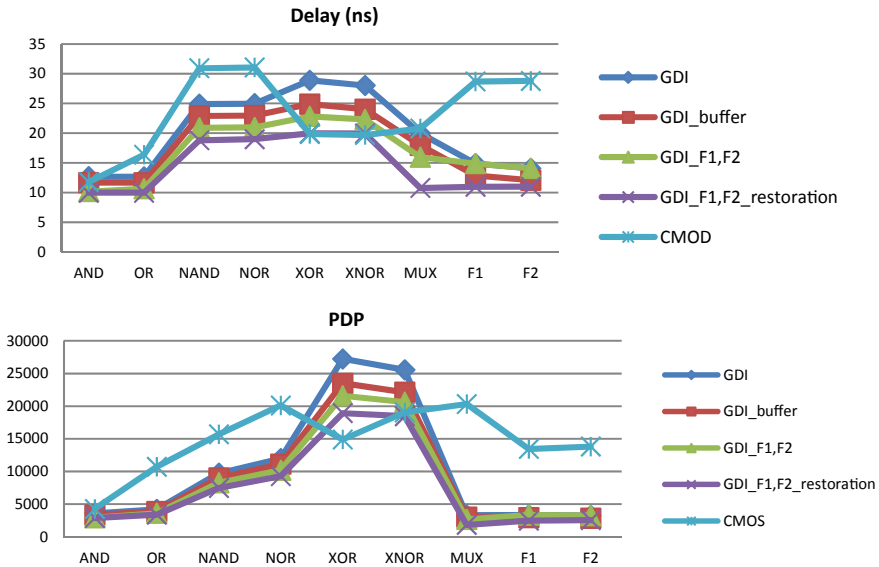


Fig. 9 Performance analysis of libraries with respect delay and PDP

References

1. Geetha S, Amritvalli P (2019) Design of high speed error tolerant adder using gate diffusion input technique. *J Electron Test* 10836–019–05802–2, May 2019
2. Kishore P, Koteswaramma KC, Chalapathi Rao Y (2020) Design of High Performance Adder Using Modified Gdi Based Full Adder. *J Mech Cont Math Sci* 15(8)
3. Sarkar S, Chatterjee H, Saha P, Biswas M (2020) 8-Bit ALU Design using m-GDI Technique. In: *Proceedings of the fourth international conference on trends in electronics and informatics (ICOEI 2020)* IEEE Xplore Part Number: CFP20J32-ART, pp 17–22
4. Morgenshtein A, Fish A, Wagner IA (2002) Gate-diffusion input (GDI): a power-efficient method for digital combinatorial circuits. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 10(5):566–581
5. Uma R, Dhavachelvan P (2012) Modified gate diffusion input technique: a new technique for enhancing performance in full adder circuits, 2nd International conference on communication, computing and security [ICCCS-2012]. *Proc Technol* 6:74–81
6. Amini-Valashani M, Mirzakuchaki S (2020) New MGDI based full adder cells for energy-efficient applications. *Int J Electron*. <https://doi.org/10.1080/00207217.2020.1818296>
7. Praveen Kumar YG, Kariyappa BS, Shashank SM, Bharath CN (2020) Performance analysis of multipliers using modified gate diffused input technology. *IETE J Res*. <https://doi.org/10.1080/03772063.2020.1782778>
8. Shoba M, Nakkeeran R (2016) GDI based full adders for energy efficient arithmetic applications. *Eng Sci Technol Int J* 19:485–496
9. Morgenshtein A, Fish A, Wagner IA (2002) Gatediffusion input (GDI)-a power-efficient method for digital combinatorial circuits. *IEEE Trans VLSI Syst* 10(5)
10. Rabaey JM, Chandrakasan A, Nikolic B (2002) *Digital integrated circuits In: A design*. 2nd 2002, prentice Hall, Englewood Cliffs, NJ

11. Ponnian J, Pari S, Ramadass U, Pun OC (2021) A new systematic GDI circuit synthesis using MUX based decomposition algorithm and binary decision diagram for low power ASIC circuit design. *Microelectron J* 108:104963
12. R.Uma, Ponnian J, Dhavachelvan P (2017) New low power adders in self resetting logic with gate diffusion input technique. *J King Saud Univ Eng Sci* 29(2):118–134, April 2017, Elsevier
13. Uma R, Dhavachelvan, P (2012) Modified gate diffusion input technique: a new technique for enhancing performance in full adder circuits, international conference on communication, computing and security (ICCCS-2012). *Proc Technol* 6:74–81, September 2012, Elsevier
14. Uma R, Vigneshwarababu P, Nakkeeran R, Dhavachelvan P () New low-power reversible logic gates using gate diffusion input technique. In: International conference on emerging research in computing, information, communication and applications, ERCICA-2013, pp 31–36

Detection of Diabetic Retinopathy Using Convolution Neural Network



K. S. Swarnalatha, Ullal Akshatha Nayak, Neha Anne Benny, H. B. Bharath, Daivik Shetty, and S. Dileep Kumar

1 Introduction

During the initial years of discovery of this disease, most of the patients were diagnosed with diabetes and greater than 65% of the patients were diagnosed with diabetes were discovered with retinopathy. From 2015 to 2019, there were more than 27% of cases of diabetic retinopathy. Person who is diagnosed with diabetic retinopathy is prone to blindness. One out of 3 people diagnosed with diabetes will have diabetic retinopathy and 1 out of 10 people will suffer from vision loss. Diabetic retinopathy can eventually lead to vision loss. Diabetic retinopathy causes damage in the blood vessels in the retinal region of the eye [1]. There are four various stages in diabetic retinopathy, namely mild non-proliferative retinopathy also known as microaneurysms; it is the earliest stage which is clinically visible changes in diabetic retinopathy. A round localized capillary dilatations are found. They are usually small red dots usually found in clusters and sometimes can also occur in isolation. But it does not affect the vision. The number of microaneurysms has strong predictive value in the progression of the diabetic retinopathy. Blockage in blood vessels is caused due to moderate non-proliferative retinopathy. Severe non-proliferative retinopathy which causes more proliferative retinopathy results in the growth of abnormal blood vessels on the retina. An algorithm for the detection of diabetic retinopathy might help doctors and researchers to recognize the symptoms of diabetic retinopathy in the people and reduce the burden of clinical trials on the specialists and researchers. An effective way of DR detection is through image processing (Fig. 1).

K. S. Swarnalatha · U. A. Nayak (✉) · N. A. Benny · H. B. Bharath · D. Shetty · S. D. Kumar
Nitte Meenakshi Institute of Technology, Bangalore 560064, India
e-mail: akshatha.n@nmit.ac.in

K. S. Swarnalatha
e-mail: swarnalatha.ks@nmit.ac.in

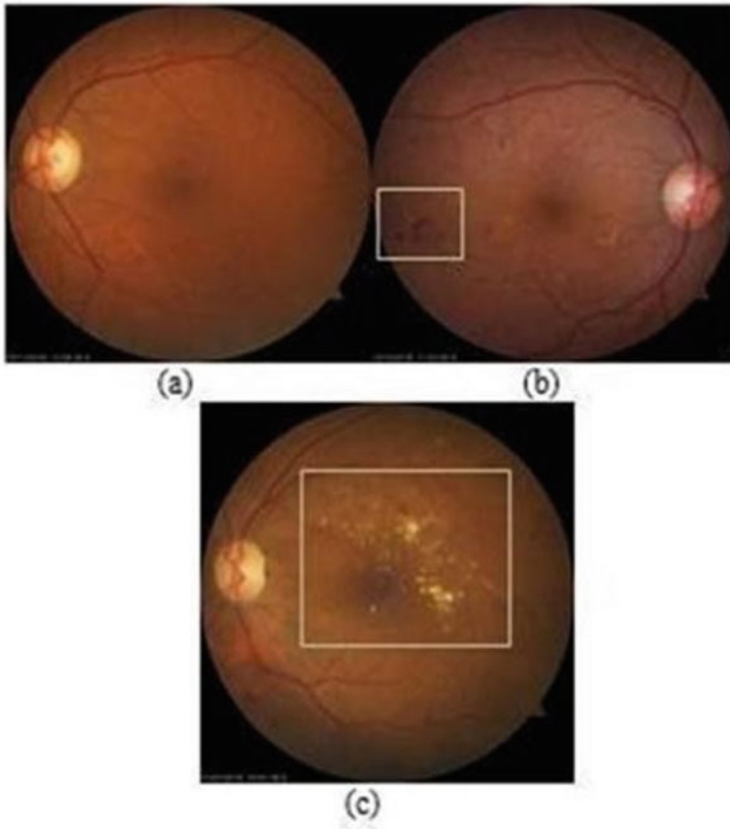


Fig. 1 **a** Normal fundus image with no sign of DR, **b** fundus image with red lesions, **c** fundus image with exudates

The research for automatic detection of DR becomes more and more crucial in the past few years. In our study, we are focusing on anomalies in the retina in the form of exudates and red lesions. It would be challenging to locate red lesions using normal image processing techniques because of its equivalent color characteristics. It is hard to considerate work needs to be done in blood vessel extraction and optic disk removal, both of which may also result in a false detection. The above-discussed problems can be overcome through CNN techniques. During training phase, CNN is capable to grasp the features. Different features with different complexities can be learned in various layers of the network. As features are extracted automatically, there is no room for manual feature extraction. The hidden layer may include sigmoid activation layer (logistic function), which is an activation function, convolutional layer and fully connected layer.

2 Related Work

The work by Carson lam uses an automatic DR grading system which is capable of classifying the images with respect to different stages of the disease based on severity [2]. The convolutional neural network (CNN) takes an input image and extracts specific features of the image without losing information on spatial arrangement. Initially, different architectures is evaluated to determine the best CNN for classification of binary task and to achieve standard performance levels. Then, training of different models is done that enhances sensitivities for different classes, which include data pre-processing and augmentation to improve accuracy of the test and also increases effective dataset size of sample. The different concerns of data quality and fidelity is checked images verified specialist doctor. Finally, the issue is addressed regarding insufficient size of samples by utilizing deep-layered CNN on color space for the recognition of task. Then, training and testing of CNN architectures AlexNet and GoogLeNet [2] a done as 2-ary, 3-ary and 4-ary. They are also trained using many techniques which include batch normalization, L2 regularization, learning rate policies and gradient descent update rules. Other studies were done using the publicly available Kaggle dataset of 36,000 retinal [3] images with 5-class labels (normal, mild, moderate, severe and end stage) and MESSIDOR-1 dataset of 1200 color fundus images verified by physician with 4-class labels. Throughout the study, the main aim has been to garner a more effective means of classifying diabetic retinopathy at early stage and thus giving better chance to patients for treatment.

The work done by Nikhil and Rose involves three CNN designs with features to develop a classification system for different stages of DR based on color fundus images [4]. The classification is done based on the 5-stage severity of DR. Here, deep learning-based CNN networks is deployed. There were many medical studies conducted in field of designing algorithm for classification of DR from fundus images of retina. But usually they were only binary classifiers differentiating two stages of DR which included normal and DR [4]. But in here, they check the three prediction accuracies of different deep CNN architectures and combinations of these networks when deployed as DR stage classifier. The study is done using Kaggle dataset having 500 images of retinas. Here, they found that after using VGG16, AlexNet and Inception Net V3; they got highest accuracy of around 80.1.

The work by Wang and Yang adopts CNN as predicting algorithm with the aims to develop more efficient CNN architecture that can be specifically useful for dataset of large scale. The CNN built by them has no fully connected layer but pooling layers and convolutional layers [5]. This method reduces the parameters number significantly (fully connected layers have more parameters than convolutional layers in CNN) and thus provides better conditions for neural network interpretability. They have shown in their experiments that with less parameters and no fully connected layers CNN architecture will have better performance in terms of prediction. The advantage of the network structure proposed is that it can provide a RAM of input image to show that each pixel of image that is input for DR detection. This RAM output somehow will mitigate the known shortcomings of CNN as black box method [5]. They are

of the opinion that RAM output makes proposed solution better self-explained to motivate the doctors to trace the cause of disease for the patient and explain clearly regarding which region of the fundus color image is the main cause of the disease. By analyzing the insights of CNN, they have yielded good performance in terms of prediction of the diseases facing challenges due to CNN as it has non-convex character.

In this paper, Arcadu and Benmansour have utilized artificial intelligence (AI) to offer a solution to the problem. Deep learning (DL), specifically deep convolutional neural networks (DCNNs), has been used for the assessment of images to get a particular target for prediction of the outcome. The use of DCCN algorithms has been right now used in numerous areas in health care like dermatology, radiology. Particularly in ophthalmology, significant work has recently been conducted. Due to automation of DR prediction and grading and of risk factors by DCNN [6]. The main goal of the work is to go past use of DL for DR diagnostics and for assessment of feasibility of DCNNs operating on 7-field CFPs for the prediction future threats of significant worsening of DR of patient over a given span of 2 years. The DCNNs are trained on high-quality 7-field CFPs which have been graded for DR severity by and highly trained experts masked using the diabetic retinopathy severity scale and early treatment diabetic retinopathy study from many clinical trials. Earlier studies have limited the use of DCNNs to optic nerve [7] or fovea critical CFPs [6]. The findings have highlighted the significance of signal prediction located in the patients peripheral retinal [3] fields with DR and tell that such an algorithm on further development and correct validation can and will help us fight blindness by quickly identifying the DR progressors for referring to a retina specialist or using it in a clinical trial which is intended to target early stage of diabetic retinopathy.

3 Architecture

In this section of implementing the system architecture, we are going to deliberate on every component which completes the deployment. The components are as follows: (1) machine learning model, (2) web services and (3) end-user application (Fig 2).

3.1 *Machine Learning Model*

The machine learning model detects the presence of diabetes retinopathy. It implements the deep learning algorithm. The machine learning model takes input from the user and reveals the probability of diabetic retinopathy. For spotting the existence of retinopathy, the model should be trained repeatedly.

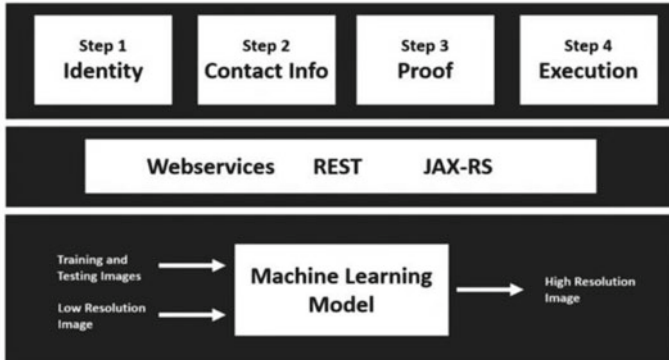


Fig. 2 Machine learning model

3.2 Web Services

Implementing web services exposes the model to the end-users. The user would be uploading the fundus image [8] and then would place a request to execute the model. Once the request is received by Webservice API, the input image will be stored in the receiver machine and the Python code would be invoked by stating the folder location which then processes the image. The output will be displayed after processing in a user-friendly graphical interface. For testing the implementation of web services, REST client is used.

3.3 End-User

The end-user/third-party application has been implemented to demonstrate the usage of the web service to the patients. In this application, we implement four steps,

- Step1: User’s identity, we collect the users first name and the last name.
- Step2: Contact info, we collect the email ID and mobile number of the client.
- Step3: Proof, we will send the user an OTP to ask them to prove his or her identity.
- Step4: Execution, the user uploads an input image, and clicking on the run algorithm will invoke the web service implemented as discussed above, and then, the result is made available to the client.



Fig. 3 Dataset samples

4 Methodology

4.1 Dataset

The high-resolution fundus image database consisting of more than 1000 images with dimension 1050×1050 is used for training the CNN algorithm. And around 100 fundus images are used for testing. Sample fundus images are shown in Fig. 3.

4.2 Data Pre-processing

Initially, the unique fundus images are resized to a measurement of 224×224 . Due to the immense information and varying contrast of images taken from the fundus cameras, pre-processing is very necessary. Without pre-processing then the images suffer from vignetting effects and image distortion. The pre-processing method is necessary because of nonlinearity in fundus images.

4.2.1 Greyscale Conversion

At first, all the fundus images which are in Red, Green, Blue (RGB) are reconstructed to greyscale from the weighted average of Red, Green, Blue pixels within which 0.299 of the red part, 0.114 of the blue part and 0.587 of the green part are considered.

$$\mathbf{I} = \mathbf{R} * 0.299 + \mathbf{B} * 0.114 + \mathbf{G} * 0.587 \text{ where } \mathbf{I} = \text{resultant pixel}$$

4.2.2 Resizing

The images which are transformed to greyscale are rescaled to a fixed size of $336 * 448$ pixels.

4.2.3 Pixel Rescaling

The fundus image is rescaled to a pixel value between 0 and 1 divided by 255 for better computation. The pixels with lesser than a threshold value are rescaled to 0, and others are rescaled to 1.

CNN is extensively-used for various applications such as image processing, pattern recognition and video recognition. CNN has been widely recognized for image classification. It takes in an image as input and classifies into the appropriate category. CNN has couple of unseen layers in which convolution is done to extract features and other valuable information from the image. The output is obtained from the classification layer. CNN is the class of deep learning neural networks that combines the accomplished attributes with the input data and uses a 2D convolutional layer. A database with around 1000 high definition retinal image was taken for training the CNN model. The database undergoes several pre-processing procedures. And, 30 fundus images are used for testing the trained CNN model. Comparatively, CNN uses lesser pre-processing. This is because they can learn the features of an image during the pre-processing and training session itself (Fig. 4).

CNN is a blend of divergent layers like I/P layer, an O/P layer and the hidden layers. The hidden layer consists of convolution layers, sigmoid layers, pooling layers and fully connected layers. Convolution operation is applied to the convolution layer. The product between filters and image patches is been computed using convolutional features. The features and information from the convolution layer is passed on to the activation layer, where it performs the threshold operation to each element of the image. Element-wise activation is applied to the output of the previous layer. The

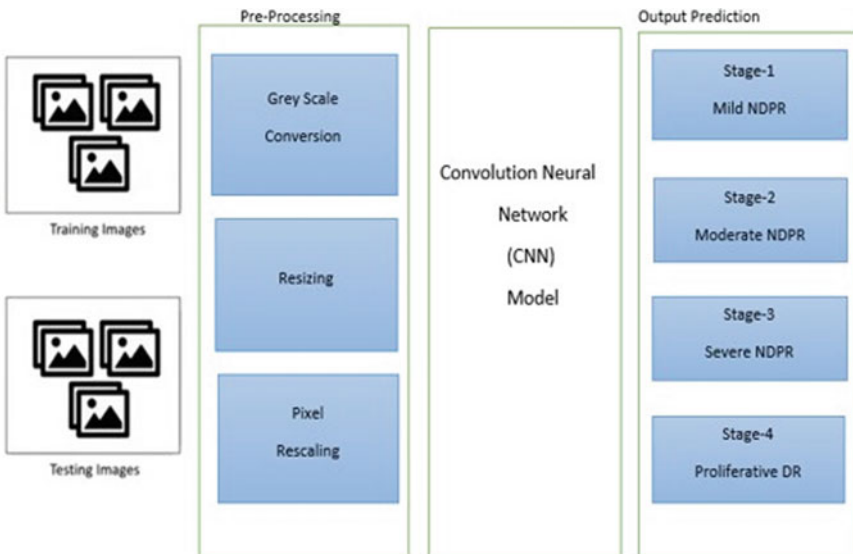


Fig. 4 CNN model

information of the sigmoid layer is passed on to the pooling layer. The average pooling layer down-converts the volume of the pixels of the image and lessens the memory used. The data collected is passed on to the fully connected layer, which is the final layer of the convolution network. This layer has all the features such as the edges, contrast, blobs and shapes. This layer collects all the accumulated information of all the preceding layers.

4.3 CNN Classification

The overall pixels in the input fundus image is similar to the count of I/P layer neurons. The CNN layer employs the CNN features and generates the outcome between the filter the image patches. Sigmoid linear can be utilized for the activation layer. Sigmoid activation function is also called as logistic function. The sigmoid function exists between the range of 0–1. Therefore, it is especially used in models that need to predict as output. The probability is in the range of 0–1. For the outcome of the convolutional layer, segment-wise activation is been applied. To bring down the memory needs and to elevate the computation, the volume of the pooling layer has been down converted. The properties like contrast, shapes, edges and blobs are been held as an information in fully connected layer. For the terminating layer of CNN softmax layer is been used. For individual class, decimal probabilities would be assigned. The I/P images is categorized into two different classes like abnormal (with DR) normal (without DR).

A fully connected layer takes the output of all the neurons from the former layer. Diabetic retinopathy is classified into four stages:

- (i) Mild non-proliferative retinopathy: This is considered as expeditious stage of diabetic retinopathy. Microaneurysms occurs in this stage. They are nothing but the tiny areas swellings within the retinal tiny blood vessels.
- (ii) Moderate non-proliferative retinopathy: The blood vessel which nurtures the retinal region would lead to swelling, and in some instances, they are even blocked.
- (iii) Severe non-proliferative retinopathy: This is the newfangled stage. In this phase, there are chances of blood vessels [9] being increased in numbers which would block the retinal regions, and it would lead to denying not many areas of the retina with their blood supplies.
- (iv) Proliferative retinopathy: This considered as the newfangled stage of diabetic retinopathy. Here, the blood vessels which are growing are abnormal and fragile. This stage is more dangerous and is incurable.

The CNN model is trained to spot and categorize the fundus image into different stages of diabetic retinopathy. Based on the severity of hemorrhages, microaneurysms, distortion in the inner layer of the retina, and red lesions in the fundus image, the classification of diabetic retinopathy is observed.

5 Performance Parameters

The parameters which are used to gauge the performance of the proposed model are accuracy and loss. To calculate the accuracy, we have the quadratic weighted kappa, also known as Cohen’s kappa, the official evaluation metric. For our filter, we will use a custom callback to monitor the score and plot it at the end.

The definition of Cohen kappa (κ) is:

$$K = \frac{P_o - P_e}{1 - P_e}$$

where P_o is designated as the relative observed agreement identical to accuracy, whereas P_e is the hypothetical probability of chance of agreement, using the observed data to calculate the probabilities of each observer randomly seeing each categories of output (Fig. 5).

Taking an example of a binary classification problem, say we have constructed the following table:

The “observed proportionate agreement” is calculated the same way as accuracy:

$$p_o = acc = \frac{tp + tn}{all} = 2 + 26 = 0.66$$

Fig. 5 Custom callback

true	pred	agreement
1	1	true positive
0	0	true negative
1	0	false negative
1	0	false negative
0	0	true negative
1	1	true positive

$$K = \frac{P_o - P_e}{1 - P_e}$$

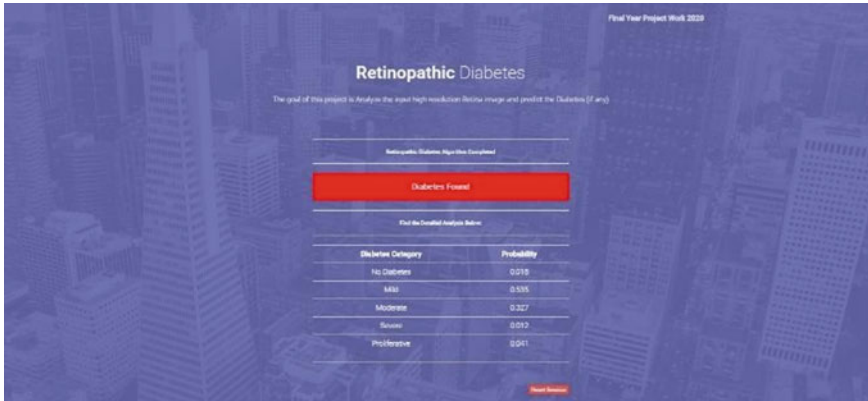


Fig. 6 Experiment result 1

6 Experimental Result

In the project, the experiment is done on several number of images obtained from the online dataset and also on the images obtained from various Web sites. The experiment is done on both the fundus images with diabetic retinopathy and without diabetic retinopathy. After pre-processing and extraction, the images were trained with the CNN algorithm which consisted of multiple layers. The training was done on more than 3000 images for ten epochs. More the number of epochs (iteration), more the accuracy. Testing the trained model resulted in an accuracy of 94.5%.

Here, we evaluate the performance, efficiency and result of the CNN trained model with various parameters like speed, accuracy, etc.

The input image is diagnosed and determined whether the person has DR or not. And, the result is analyzed to determine and classify in which stage the disease is probably present and displayed (Fig. 6).

In this above image, the result is shown as diabetic retinopathy found as the probability of the disease is more for mild stage (Fig. 7).

In this above image, the result is shown as diabetic retinopathy not found as the probability of the disease is more for no disease according to the model accuracy.

In this experiment, the diagnosis is performed to determine whether the person has DR or not; then if the person has DR, the result is analyzed to determine the probability in which stage the disease might be present more.

7 Challenges

The major challenge was about dealing with the computational complexity that the cloud service that we have with us could not handle. The program had to deal with

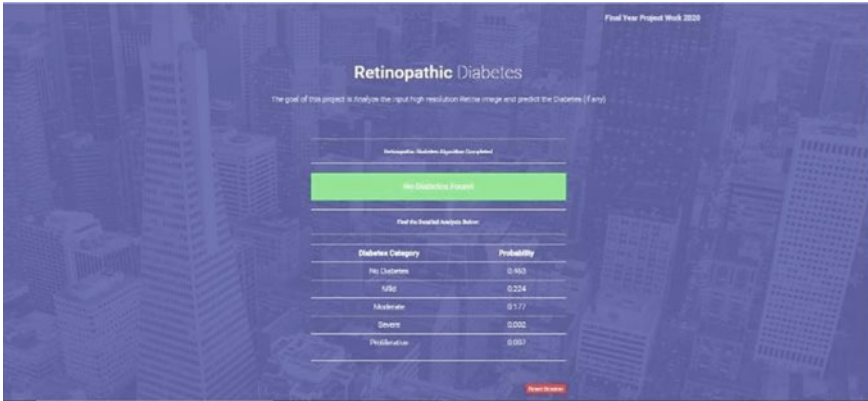


Fig. 7 Experiment result 2

a diverse number of neurons. Initially while working with a laptop that was running on an AWS did not give the expected results. Then, we moved on to use the Digital Ocean cloud service which handles machine learning workloads better than AWS. With this, we were able to test more images and acquired the correct predictions.

8 Conclusion

Today, due to a huge population of diabetic patients present and the possibility of them suffering from diabetic retinopathy has made automatic DR systems to be of great demand; automated detection provides an opportunity for us to prevent loss of vision among patients due to diabetic retinopathy. In this paper, the objective is to make a clinically usable system. Here, the systems use the fundus images captured from screening to detect diabetic retinopathy. This paper uses CNN architecture for the detection and classification of diabetic retinopathy. For clinical applications, the trained models are deployed on cloud computing models. The studies in this paper are supposed to assist medical doctors to easily detect diabetic retinopathy and reduce the number of reviews of doctors and also help patients to be aware of their medical condition.

In the future, a larger dataset will be included and a broader study will be done. The data we get will be further used to modify and improve the accuracy, efficiency of the models. As per the observation made, it is understood that neural networks which are a machine learning technique have ensuing scope in disease detection. The competency of R-CNN technique is been proven in the field of object detection. As per our work, it is proven that CNN is been useful to identify knee-high features. When it comes to lesion detection, CNN is considered to be more accurate with its 93.8% accuracy rate.

References

1. Aslani C (1999) *The Human Eye: Structure and Function*. Sinauer Associates, Sunderland, MA
2. Lam C, Yi D, Guo M, Lindsey T (2018) Automated detection of diabetic retinopathy using deep learning
3. Zhang L, Fisher M, Wang W (2015) Retinal vessel segmentation using multi-scale textons derived from keypoints. *J Comput Med Imaging Grap* 45:47–56
4. Nikhil MN, Rose AA Diabetic retinopathy stage classification using CNN
5. Wang Z, Yang J (2018) Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation. In: *The workshops of the thirty-second aaai conference on artificial intelligence*, Microsoft corporation zhiguang.wang@microsoft.com
6. Arcadu F, Benmansour F (2019) Deep learning algorithm predicts diabetic retinopathy progression in individual patients 2, 92 (2019), sep 20. <https://doi.org/10.1038/s41746-019-0172-3>. eCollection 2019
7. Hassana C, Boyce JF, Cook HL, Williamson TH (1999) Automated localization of the optic disc, fovea and retinal blood vessels from digital color fundus images. *British J Ophthal* 83(8):902–910
8. Sinthanayothin C, Boyce JF, Williamson TH, Cook HL, Mensah E, Lal S, Usher D (2002) Automated detection of diabetic retinopathy on digital fundus images. *Diab Med J* 19
9. Akram MU, Khan SA (2013) Multilayered thresholding-based blood vessel segmentation for screening of diabetic retinopathy. *J Eng Comput* 29(2):165–173
10. Vermeer KA, Vos FM, Lemij HG, Vossepoel AM (2004) Model based method for retinal blood vessel detection. *Comps Bio Med* 34(3):209–219
11. Chaudhuri S, Chatterjee S, Katz N, Nelson M, Goldbaum M (1989) Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Trans Med Imaging* 8(3):263–269
12. Chanwimaluang T, Fan G (2003) An efficient algorithm for extraction of anatomical structures in retinal images. In: *Proceedings of ICIP*, pp 1193–1196
13. Hoover AD, Kouznetsova V, Goldbaum M (2000) Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Trans Med Imaging* 19(3):203–211
14. Martinez-Perez ME, Hughes AD, Stanton AV, Thom SA, Bharath AA, Parker KH (1999) Segmentation of retinal blood vessels based on the second directional derivative and region growing. In: *Proceedings of ICIP*, pp 173–176
15. Martinez-Perez ME, Hughes AD, Stanton AV, Thom SA, Bharath AA, Parker KH (1999) Scale-space analysis for the characterization of retinal blood vessels. In: Taylor, Colchester A (eds.) *Medical Image Computing and Computer-assisted intervention (Lecture notes computer science)* vol 16794, Springer, New York, pp 90–97
16. Wang Y, Lee SC (1998) A fast method for automated detection of blood vessels in retinal images. In: *IEEE Computer Society Proceedings of Asilomar Conference*, pp 1700–1704
17. Jiang X, Mojon D (2003) Adaptive local thresholding by verification based multi-threshold probing with application to vessel detection in retinal images. *IEEE Trans Patt Anal Mach Intell* 25(1):131–137
18. Zana F, Klein JC (2001) Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Trans Med Imag* 11(7):1111–1119
19. Niemeijer M, Staal J, Van Ginneken B, Loog M, Abramoff MD (2004) Comparative study of retinal vessel segmentation methods on a new publicly available database. In: Fitzpatrick M, Sonka M (eds) *Proceedings of SPIE Medical Imageing*, vol 5370, pp 648–656
20. Staal J, Abramoff MD, Niemeijer M, Viergever MA, van Ginneken B (2004) Ridge-based vessel segmentation in color images of the retina. *IEEE Trans Med Imag* 23(4):501–509
21. Zhou L, Rzeszutarski MS, Singerman LJ, Chokreff JM (1994) The detection and quantification of retinopathy using digital angiograms. *IEEE Trans Med Imag* 13(4):619–626

22. Toliaş Y, Panas SM (1998) A fuzzy vessel tracking algorithm for retinal images based on fuzzy clustering. *IEEE Trans Med Imag* 17(2):263–273; 1998. (2):105–112; 2002

Drone-Based Security Solution for Women: DroneCop



Sukanya Bharati, T. R Vinay , M. G Prasanna, N. M Sangeetha, and Shreya Roy

1 Introduction

India tops as the most dangerous place for women even in the twenty-first century. As per NCRB Report 2019, 88 rape and sexual harassment cases happen on an average day [1]. It means almost 4 cases per hour. Less than 0.14% of cases are solved in India. 99% of these cases go unreported according to government reports [2]. The existing conventional ways and the police provide help mostly after the crime is committed, and therefore, help is not offered in real time to the victim. If it was possible to decrease this time lag, most of the victims can be saved from these incidents.

Many apps and smart gadgets were developed for women's safety in the past. But there are many drawbacks. The apps keep crashing, or it needs to be opened to send an SOS to emergency contacts which are not possible at all times. The location taken by these apps is not accurate. Few apps are paid and have excessive advertisements on the screen. The features like voice-activated SOS, pressing the power button/volume button 3 times to send SOS to emergency contacts, and detecting screams to send SOS or alert fail to work sometimes. Smart gadgets for women's safety like gloves, wearable bracelets, shoes, etc., have been developed. But these gadgets are bulky and cannot be carried everywhere. So, they are not portable. These gadgets require more hardware, which in turn increases the implementation cost.

To overcome the above problems, the proposed solution is an amalgamation of the use of drones, a smartwatch, and a mobile application which will provide help to the users in real time, thus decreasing the time lag. Drones are unmanned aerial vehicles (UAVs) with mounted sensors, GPS, navigation systems and cameras, and other features that are primarily used for surveillance, security, and other purposes. Unmanned aerial vehicles (UAVs) are planes that fly without a crew or passengers.

S. Bharati (✉) · T. R. Vinay · M. G. Prasanna · N. M. Sangeetha · S. Roy
Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: sukanyabharati823@gmail.com

They might be remotely controlled cars or autonomous ‘drones’. Drones are used to monitor climate change, execute rescue operations in the aftermath of natural catastrophes, take photographs and video, and deliver commodities, among other things [3].

Being virtually connected on a platform to access help in real time makes a woman feel safer. Here, the user can send SOS through a mobile application or smartwatch. The mobile application connects the IOT safety device module (smartwatch) and triggers the drone. The drone on activation reaches the location and tracks the movement of the victim/attacker. Following the drone, the location of the attacker may be traced, and the situation thus can be taken under control.

2 Literature Survey

The literature survey was conducted in three different directions: analysis of existing apps for women’s safety, research papers on solutions proposed for women’s safety, and a real-time survey on current security situations (including women, children).

Existing apps for women’s safety: There are many trending apps in the market for women’s safety. Few apps were analyzed based on the details of the application, their working, benefits, drawbacks, UI, and reviews. The apps are as follows (Table 1).

Research papers related to women’s safety:

In the paper [10], the author has developed a solution using Internet of Things devices and a mobile application is built which keeps track of user’s live coordinates and sends it to nearby law enforcement agencies. The application user gets to know about the nearest available secured places. The disadvantage is that the IoT device is bulkier to wear. There are bugs in the application which cause SMS to deliver to different law enforcement agencies in different zones.

In the paper [11], the author has developed an embedded system having a GSM and GPS subsystem. This system sends an emergency message and live location to the predefined contacts. It also triggers an alarm. This device is easy to use and is triggered by the user on click of a button. The system does not work in offline mode, and the switch needs to be pressed every time the location is changed. Hence, it is not automated.

In the paper [12], the author has proposed an app that provides articles, reviews, and safety levels about the location by use of the camera. Users can connect with the law enforcement agencies and first responders by sending live coordinates. Users can post their reviews about the application. The drawback is that the app does not work in offline mode.

Real-time survey on current safety situations:

Real-time data was collected from family, close friends, and relatives through a Google Form to understand the current safety situation. In the form, a set of 10 questions based on women’s safety were asked. A total of 338 responses were received in 3 days from both men and women. The results are shown below (Table 2).

Table 1 App analysis of the existing women safety apps

App Name	Working of the app	Drawbacks of the app
Call for help [4]	<ol style="list-style-type: none"> 1. By pressing the power button three times, it sends SOS, video, and audio recording to the emergency contacts 2. It allows mapping safe zones by setting the radius 3. It alerts the emergency contacts, outside the radius 	<ol style="list-style-type: none"> 1. The SOS message is queued until the phone is activated
DROR—your personal safety app [5]	<ol style="list-style-type: none"> 1. By pressing the SOS button, the alert message is sent to the nearby helpline 2. The app uses GPS technology to live track the user’s trip 3. In case of route deviation, it alerts the user’s emergency contacts 4. It shows the safety ratings of a particular place 	<ol style="list-style-type: none"> 1. It has login and technical issues
Chilla [6]	<ol style="list-style-type: none"> 1. The app detects screams even if the phone is in the pocket or bag 2. The user needs to press the power button 5–6 times to send alert messages 3. When triggered, the app will send SMS with location, emails, and audio recording and automatically places calls 4. The app can be activated in 3 ways—by pressing the power button, by screaming, and by pressing SOS 	<ol style="list-style-type: none"> 1. The app fails to detect screams at times 2. The SOS button remains switched on only for 30 min
SafeON— Personal Safety App & emergency alert [7]	<ol style="list-style-type: none"> 1. SOS can be sent through three different ways -by shaking the device, by pressing the power button 5 or 6 times, and by pressing the SOS button 2. The app sends SMS alerts, records audio, and captures pictures using the rear camera 3. The app starts a siren to attain public attention 	<ol style="list-style-type: none"> 1. The location of the user is not accurate at times

(continued)

Table 1 (continued)

App Name	Working of the app	Drawbacks of the app
ProtectMii—Personal Safety App with Panic Alarm[8]	1. It sends SOS messages along with the current battery power in your phone 2. It triggers an alarm on its own by just unplugging the headphones cable or any USB device 3. Using acoustic signals, it sends alerts to nearby users for help 4. False alarm and deactivation protection enable the user to cancel an accidentally triggered alarm within 15 s	1. The user is unable to log in properly
UrSafe: Personal Security App[9]	1. The app uses voice-activated triggers to send SOS messages 2. It shares video and audio with the emergency contacts 3. It alerts emergency contacts, police stations, and the users of this app who are nearby	1. The voice-activated SOS feature does not work properly 2. It is not user-friendly 3. It does not immediately place the call to the emergency contacts

Table 2 Real-time survey results

Question No	Real-time survey (women’s safety) results
1.	What time of day or night do women and girls go out the most often and least often? (Fig. 1) <i>Inference:</i> Most often—afternoon, least often—night
2.	How do girls/women prefer to travel? (Fig. 2) <i>Inference:</i> In groups—301 responses
3.	Which places do you feel unsafe? (Fig. 3) <i>Inference:</i> Empty roads (305 responses)
4.	What type of incidents have you experienced(women)/seen other girls experience(men)? (Fig. 4) <i>Inference:</i> Staring (279 responses)
5.	What is the frequency of patrolling in your area? (Fig. 5) <i>Inference:</i> Most number of times—2(87 responses), least number of times—5(13 responses)
6.	Have you seen a woman being teased/harassed? (Fig. 6) <i>Inference:</i> 58.9% of the 338 responders have not seen women being teased 41.1% of the 338 responders have seen cases of women harassment

(continued)

Table 2 (continued)

Question No	Real-time survey (women’s safety) results
7.	<p>If yes, did you try to help? If not, why?</p> <p><i>Inference:</i> Few responses posed were positive as they were confident enough to take a stand for the people in need which includes their relatives, friends, and other public people. Meanwhile, few responses were negative because they feared not being supported by the public or were hesitant of putting themselves in danger along with the victim</p>
8.	<p>Of the existing apps, how much have they been useful?/do you use any of them? (Fig. 7)</p> <p><i>Inference:</i> 52.7% of responders have not used an app for women’s safety</p>
9.	<p>What do you think will be the best way to defend yourselves from an unusual situation using technology?</p> <p><i>Inference:</i> The majority of the responses were to use the technology at its best by creating an app that uses real-time GPS, sends SOS to emergency contacts, and alerts nearby police stations</p>

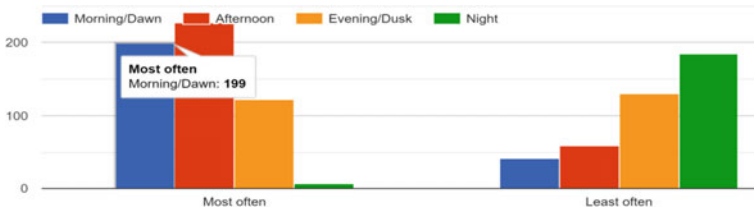


Fig. 1 Number of users versus time of the day

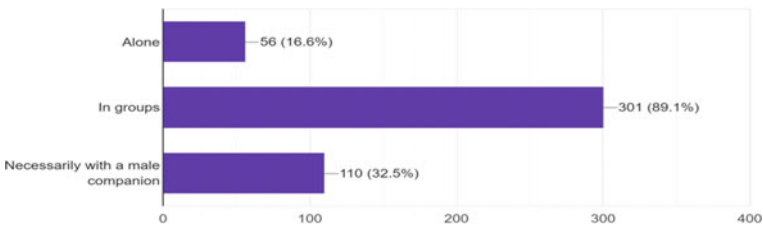


Fig. 2 Mode of traveling (alone, in groups, necessarily with a male member) versus the number of responses

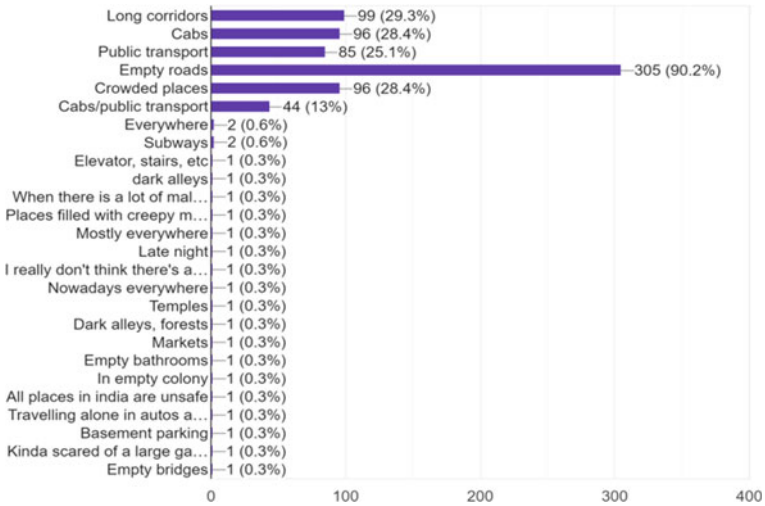


Fig. 3 Unsafe places versus number of responses

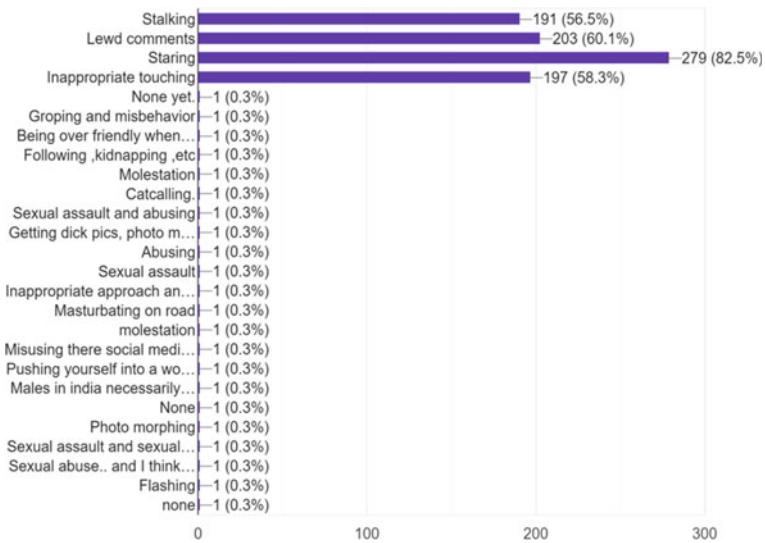


Fig. 4 Incident experienced/seen versus number of responses

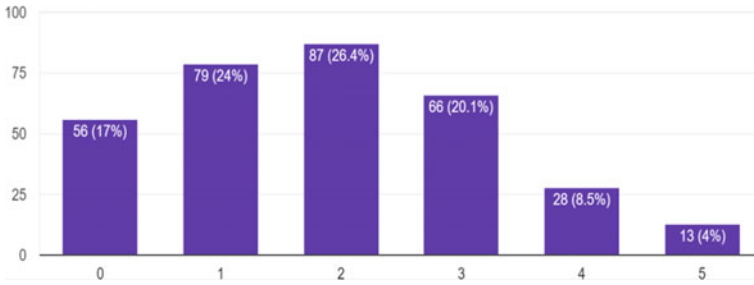


Fig. 5 Number of responses versus frequency of patrolling

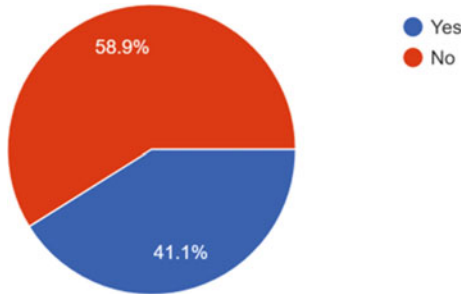


Fig. 6 Women harassment cases in the form of a pie chart

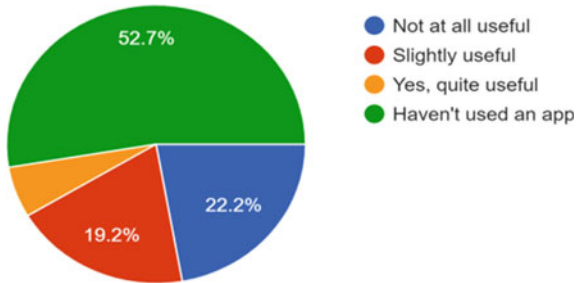


Fig. 7 The usefulness of the apps existing in the market for women's safety

3 Proposed Methodology

The proposed methodology is an amalgamation of three different technologies which include a drone, a smartwatch, and a hybrid mobile application. This system will help the user to get help in real time much faster in comparison with the conventional and the existing solutions in the market (Fig. 8).

3.1 Use Case Diagram

3.2 Sequence Diagram

Here is the sequence of activities that take place when a user triggers our system. It is as depicted in Fig. 9.

3.3 Data Flow Diagram

In the level 0 data flow diagram, the abstract of the entire project is depicted in Fig. 10.

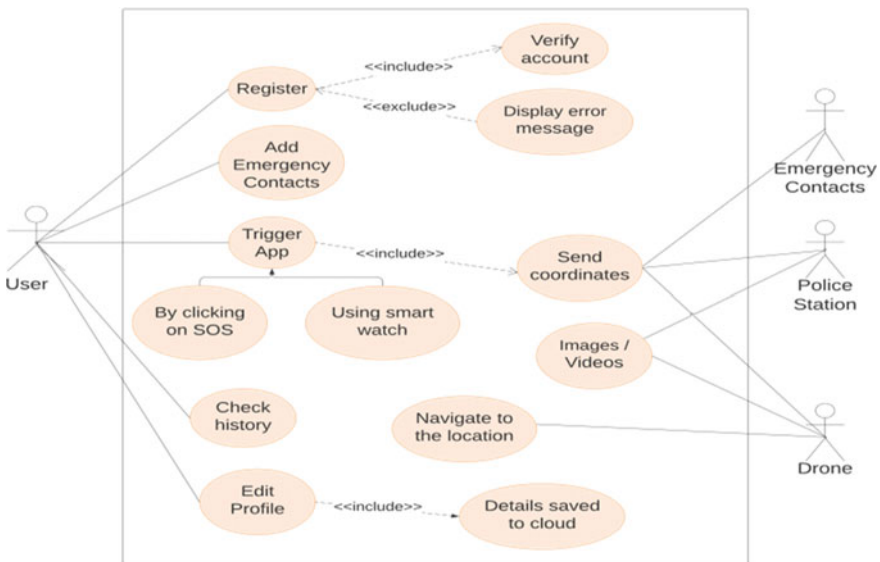


Fig. 8 Use case diagram

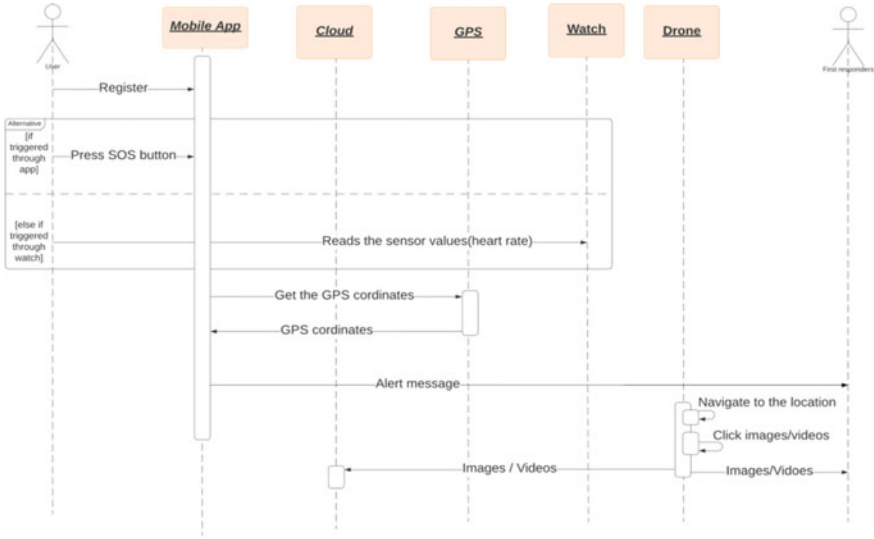


Fig. 9 Sequence diagram

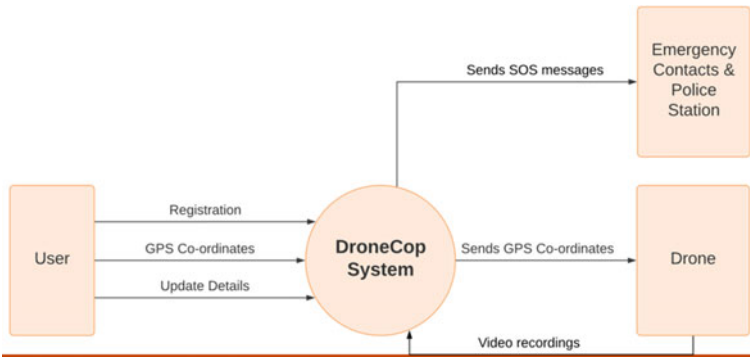


Fig. 10 Level 0 DFD

Level 1 data flow diagram represents each of the sub-processes that form the complete system. The diagram is as depicted in Fig. 11.

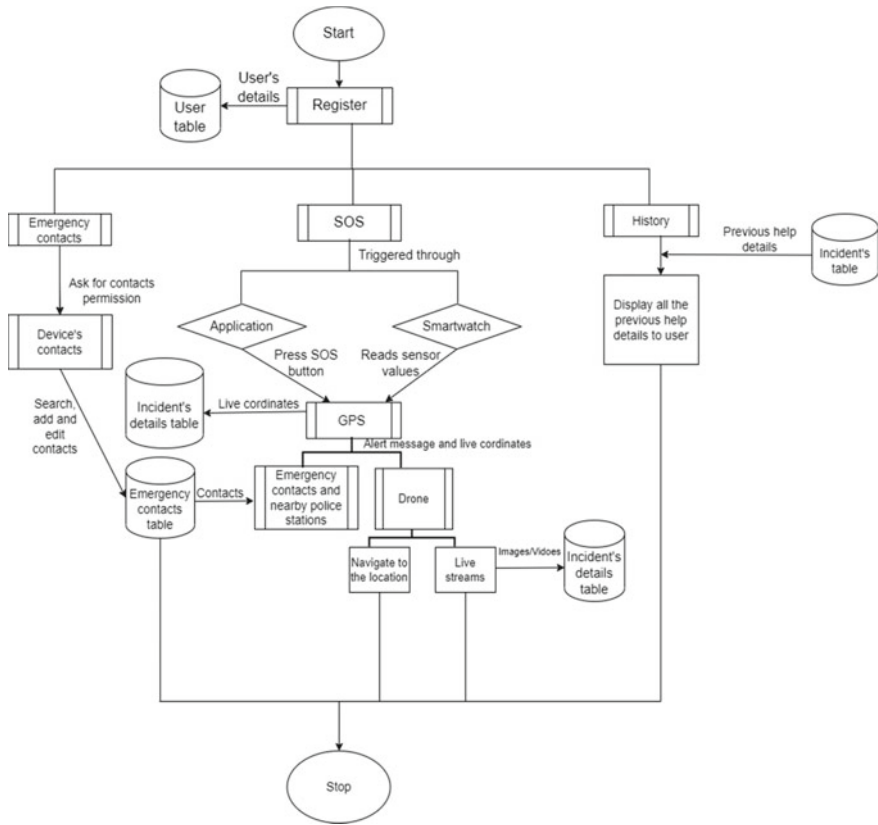


Fig. 11 Level 1 DFD

Algorithm—DroneCOP System:

Input: Emergency trigger by the user (using mobile app or the smartwatch).
 Output: Drone will reach the user’s location and livestreams the situation to the control station.

Step 0: Start.

Step 1: The user registers on the app, and the information is acknowledged. When the user logs in, the information is verified.

Step 2: Using the update module, the user can update his/her details.

Step 3: The history module stores the record of the previous help offered to the user.

Step 4: When an emergency occurs, the user triggers the application. This can be done in two ways: by pressing the SOS button or when there is a sudden surge in the sensor values of the watch, the watch will trigger an SOS.

Step 5: The application further triggers the drone by sending live geo-coordinates.

Step 6: The application shares the GPS coordinates with the drone and also the first responders to inform them about the help required.

Step 7: Once the drone is triggered, it takes off, reaches the destination, and live feeds to the ground station via the Internet.

Step 8: Stop.

4 Results and Summary

During emergencies, the novel security system developed will help the user in real time by use of drones going to the current location and sending live videos and images to the cloud and the first responders. The fear of the attacker being watched over by the drone might end up escaping from the location, and with this, the entire situation can be stopped. The system is tested under different situations, and the parameters considered here are a time to respond, the time (morning, evening, night), distance, altitude, and weather conditions. The summary is presented in Table 3, (Figs. 12 and 13).

From the above data, it can be concluded that the approximate time to initialize the system is 20 s.

Table 3 Drone testing details under the mentioned circumstances

S. No.	Situation/Area	Dimensions (distance, altitude in m)	Time taken to reach the place (s) M—morning E—evening N—night	Takeoff time (in s)	Net flight time (in s)
1	Day 1 Clear sky, no breeze	100, 10	M—14 E—16 N—12	M—58 E—57 N—61	M—72 E—73 N—73
2	Day 2 Partially cloudy, light breeze	100, 10	M—17 E—18 N—13	M—55 E—58 N—62	M—72 E—76 N—75
3	Day 3 Partially cloudy, windy	100, 10	M—22 E—23 N—18	M—61 E—58 N—68	M—83 E—81 N—86
4	Day 4 Clear sky, no breeze	200, 10	M—28 E—25 N—22	M—59 E—61 N—55	M—87 E—86 N—77

Fig. 12 Welcome page

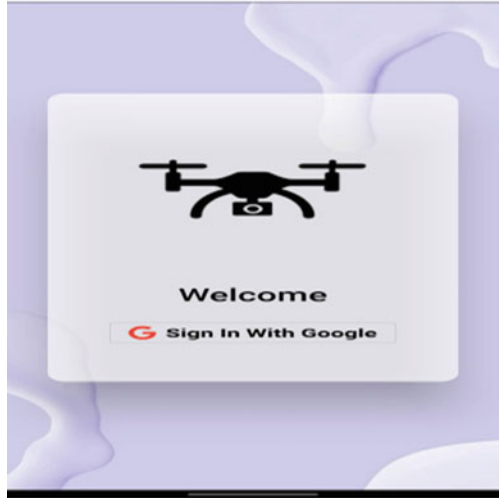
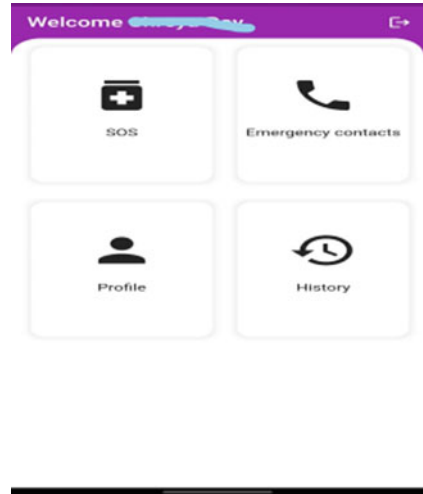


Fig. 13 Home screen



For the script to fetch the geo-coordinates, it takes around 20 s. The takeoff for the drone can be estimated to be 60 s and the peak speed to an approximation of 10 m/s, and the average speed is around 7-8 m/s (Figs. 14, 15 and 16).

Time taken to initialize the system \approx 20 s.

Time taken for the script to fetch the geo-coordinates \approx 20 s.

Time taken by the quad to take off \approx 60 s.

Peak speed of the drone = 10 m/s.

The average speed of the drone = 7-8 m/s.

Fig. 14 Adding emergency contacts

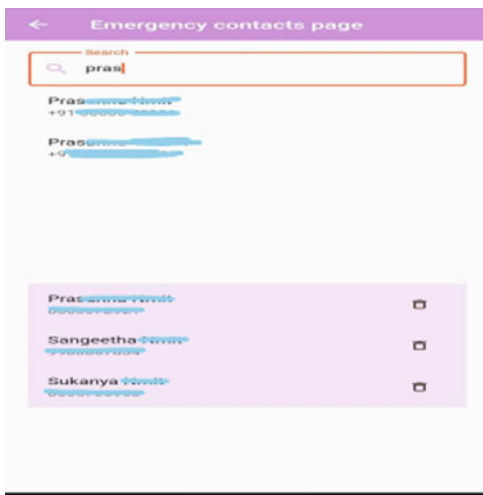


Fig. 15 Sending SOS

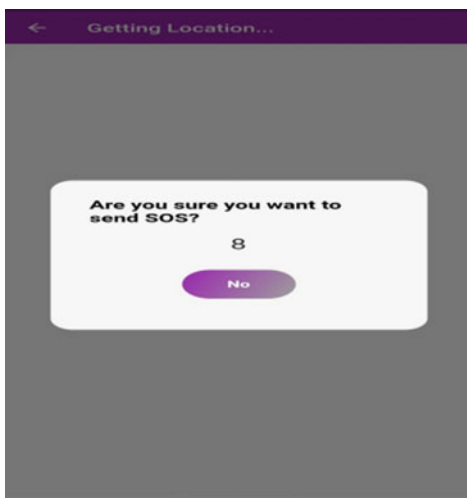


Fig. 16 Drone



5 Snapshots of the System

6 Conclusion and Future Work

DroneCop, a system built for women's safety, is described in this study report. By tapping a button on the app or sensing the heart rate using a smartwatch, users can send alert messages to first responders and law enforcement agencies. The benefit of using this technology is that a drone can get to a victim's location way faster than police and emergency personnel can, allowing the drone to trace the assailant. This system has been tested under various circumstances and environmental conditions, and the results have been studied. This system's scope could be expanded in the future to include natural disasters, survey and surveillance, road, and pole inspection. It might also be extended to include rescuing victims if the mobile network is unavailable after the initial alert, and alternative methods of triggering, such as pushing the power button or using voice to activate SOS.

References

1. <https://www.indiatoday.in/diu/story/no-country-for-women-india-reported-88-rape-cases-every-day-in-2019-1727078-2020-09-30>
2. <https://www.livemint.com/Politics/AV3sIKoEBAGZozALMX8THK/99-cases-of-sexual-assaults-go-unreported-govt-data-shows.html>
3. https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle
4. <https://economictimes.indiatimes.com/tech-life/14-personal-safety-apps-for-women/14-circle-of-6/slideshow/45451296.cms>
5. <https://inc42.com/buzz/dror-raises-funding-from-ip-ventures-to-make-safe-spaces-for-women-tourists/>
6. <https://therodinhoods.com/post/chilla-personal-safety-app-that-detects-a-scream/>
7. https://download.cnet.com/SafeON-Personal-Safety-App-Emergency-Alert/3000-31713_4-78687765.html
8. <https://protectmii.com/>
9. <https://www.usatoday.com/story/money/2019/12/10/digital-safety-ursafe-hands-free-safety-app-livestreams/2628566001/>
10. Kabir AZMT, Tasneem T (2020) Safety solution for women using smart band and CWS app. In: 2020 17th International conference on electrical engineering/electronics computer, telecommunications and information technology (ECTI-CON). IEEE, 2020
11. Ruman MR, Badhon JK, Saha S (2019) Safety assistant and harassment prevention for women. In: 2019 5th International conference on advances in electrical engineering (ICAEE). IEEE
12. Chaudhar P et al. (2018) Street smart': safe street app for women using augmented reality. In: 2018 Fourth international conference on computing communication control and automation (ICCUBEA). IEEE

Analysis of Granular Parakeratosis Lesion Segmentation: BCE U-Net vs SOTA



Sheetal Janthakal  and Girisha Hosalli 

1 Introduction

“Granular parakeratosis is a disorganized keratinization of hyperkeratotic flexural erythema that arises as a result of irritants or other reasons altering the flexural microbiota.” Granular parakeratosis is a response pattern caused by epidermal disturbance, in turn caused by a variety of factors such as detergents [1].

The basic concept employed in the diagnosis of skin lesion discoloration is automated lesion detection systems. Due to the complexities and difficulty of human interpretation, automated lesion image analysis has become an important study and research subject [2].

To identify a lesion, an automated detection system follows a step-by-step method. In the first stage, pre-processing performs the image enhancement and restoration tasks. The next stage, segmentation, is to separate the affected from an unaffected part of the skin [3]. Segmentation is a required stage in the automated detection system since it reduces data complexity and simplifies the recognition and classification process. The general architecture of the segmentation process is shown in Fig. 1.

For defining lesion boundaries, many image segmentation approaches have been improved.

“For segmentation and classification, Chen et al. [4] presented a multi-task architecture. To take advantage of beneficial features, a feature gate was created that connected the segmentation and classification networks. This method is evaluated on the ISIC 2017 challenge dataset. The results obtained were better than a single classification network with an accuracy of 80 versus 77%, but not quite as excellent as the best performer in the challenge (accuracy of 80 versus 89% [5]).”

S. Janthakal (✉) · G. Hosalli

Department of Computer Science and Engineering, Rao Bahadur Y Mahabaleswarappa Engineering College, Bellary, Karnataka, India
e-mail: sjanthakal@yahoo.co.in

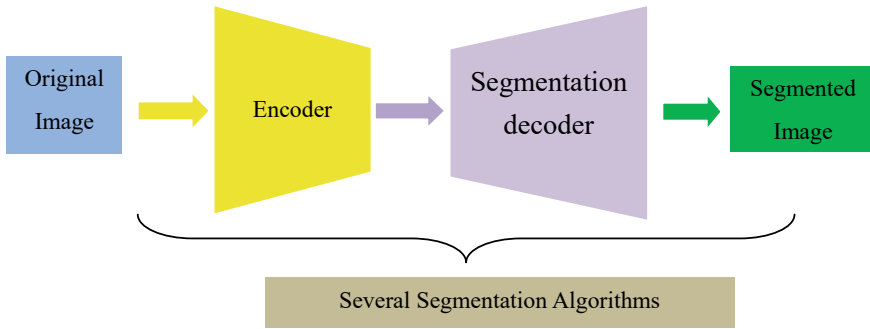


Fig. 1 Segmentation architecture

Yang et al. [6] built a multiple target CNN to segment and classify the lesions of ISIC 2017 challenge dataset. The network's encoder component used a GoogleNet CNN that had been pre-trained, while the segmentation branches used a U-Net decoder model. The results were better than a GoogleNet-based classification model with an accuracy of 89% vs. 85.7%, but not as good as the best performance in the ISIC 2017 competition (accuracy of 88.6 percent vs. 91.1 percent [7]).

The segmentation process can be carried out using a variety of techniques. As time has passed, all of the techniques have become less feasible for all types of images. Hence, a new way of implementation is to be incorporated. The suggested *U-Net* with binary cross-entropy loss function is described and compared to existing state-of-the-art image segmentation approaches on granular parakeratosis lesions in this research.

The paper is organized in the following pattern. In Sect. 2, we review the literature concerned with the segmentation of skin lesions. Thereafter, in Sect. 3, the architecture of the proposed system is given. Section 4 gives a brief description of the existing segmentation techniques and the proposed *U-Net* with BCE. Then, Sect. 5 covers the results obtained by implementing *U-Net* with BCE and comparison with other state-of-the-art techniques. Section VI is the conclusion of the paper.

2 Related Studies

Segmentation is an important stage because:

- It retrieves the features such as border irregularity, shape and asymmetry;
- Border detection accuracy is required for retrieving clinical information such as pigmented networks, uneven stripes and blue-white areas.

As a result, the most active field in an automated dermoscopic image processing is segmentation (boundary detection). A number of segmentation techniques have been developed to distinguish the lesion area from the background skin [8–11].

Ünver et al. [12] proposed a skin image segmentation pipeline that combined the GrabCut algorithm with a deep convolutional neural network (DCNN) called “You Only Look Once (YOLO).” The approach was tested on the PH2 and the ISBI 2017 datasets. This pipeline model has a sensitivity of 90%, an accuracy of 93% and a specificity of 93%.

The work of [13] simultaneously executes an auxiliary job, edge prediction, with the segmentation task. “The edge prediction branch directs the learned neural network to focus on the segmentation masks border.” During the training stage, this approach predicts both the segmented mask and its matching edge (contour), and only the segmentation mask is employed for prediction during the testing phase, yielding a sensitivity of 88.76 and an accuracy of 94.32.

Lei Bi et al. [14] introduced a new FCN approach for automatically segmenting skin lesions. The method used the primary visual aspects of skin lesions learnt and deduced from several embedded FCN stages to achieve accurate segmentation without utilizing any pre-processing procedures. On the PH2 dataset, the approach has an accuracy of 88.78, a sensitivity of 91.88 and a specificity of 89.42.

All of these approaches are less feasible for segmenting the granular parakeratosis lesion. Hence, *U-Net* with binary cross-entropy is proposed that is described in the next section.

3 System Architecture

Granular parakeratosis is a rare red, scaly skin disorder that mostly affects body folds, particularly the armpits. Granular parakeratosis is most commonly associated with middle-aged women, but it has also been documented in newborns, children and males of all ethnicities. It may progress to a malignant stage if not treated at an early time. Thus, an automated and intelligent identification technique is required while processing the skin lesion images. An automated model implemented here is using the *U-Net* architecture with binary cross-entropy function. The system architecture is shown in Fig. 2.

The dataset obtained from the publicly available site is input to the proposed model initially. The images of size $224 \times 224 \times 3$ go through the encoder and decoder with the application of binary cross-entropy loss function. Finally, we get the segmented images of size $224 \times 224 \times 3$ as shown in figure obtaining the highest accuracy, sensitivity and minimal loss.

The purpose of this study is to compare several segmentation strategies on granular parakeratosis lesions and then to compare the results produced from the implementation of the suggested *U-Net* binary cross-entropy method with numerous existing state-of-the-art techniques.

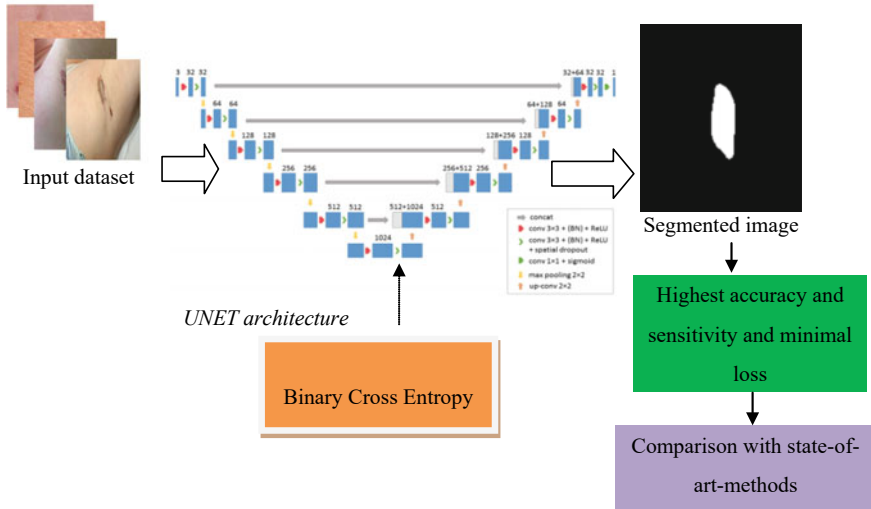


Fig. 2 System architecture

4 Segmentation Techniques

As technology progresses, image segmentation methodologies are gaining importance in the field of digital processing. Researchers and scientists have created several image segmentation approaches throughout the years [15–17]. In this paper, the segmentation procedure is carried out using Unet and the binary cross-entropy approach. Also, the RGB images are first scaled to 224×224 pixels before being converted to NumPy arrays.

4.1 Fcn

Convolutional 2D layers, dropout and batch normalization regularization techniques are stacked in the FCN model. This regularization is used to avoid overfitting. To account for nonlinearity, a ReLU activation layer is included. The input images' height and width are specified as (224, 224, 3). The number of channels for coloured images (RGB) is fixed at three [18, 19]. But the drawback of this architecture is it ignores the small lesions which are of greater importance in identification of the disease.

4.2 SegNet

The SegNet architecture includes an encoder and a decoder network and a pixel-wise classification layer. The encoder network includes 13 convolutional layers that match with the first 13 convolutional layers of the VGG16 network [20]. Using the encoder's memorized max-pooling indices, the decoder network up samples the input feature. A sparse feature map is created as a result of this approach [21, 22]. The disadvantage is it provides less accuracy for smaller datasets but the model should be designed in such a way that it provides better accuracy for small as well as larger datasets.

4.3 DeepLabv3 +

For semantic segmentation, DeepLabv3 + is a state-of-the-art deep learning architecture designed by Google. It improves on DeepLabv3 by adding a basic effective decoder module to improve the segmentation results [23]. The purpose of architecture is to give semantic labels to each pixel in the input image [24]. The limitation of this method is it mixes the target pixel and leads to a blurry segmentation. The pitfalls of all the above approaches can be overcome by the proposed U-Net with binary cross-entropy technique.

4.4 U-Net

Olaf Ronneberger et al. [25, 26] created the UNET for bio-medical image segmentation. There are two paths in the architectural structure. The contraction path, also known as the encoder, captures the context of an image. The encoder is made up of layers that are both convolutional and maxpool. The decoder or expanding path allows for transposed convolutions to be used for localization.

The binary cross-entropy loss function is implemented here, and the model is built using an Adam optimizer. The loss function for binary cross-entropy is provided by an Eq. (1):

$$H_p(q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \quad (1)$$

where y represents the label and $p(y)$ indicates the predicted probability of the point for all N points.

That is, for each point $y = 1$, it concatenates $\log(p(y))$ to the loss. Conversely, it adds $\log(1-p(y))$ for the point $y = 0$.

The model is trained on a Windows 10 system with an Intel Core i5-2.4 GHz processor and 8 GB RAM configuration using Python.

5 Results and Discussion

The results of parakeratosis lesion segmentation using the *U-Net* with the binary cross-entropy method are shown in Fig. 3.

Figure 4 shows the images obtained by performing the segmentation using FCN, SegNet and DeepLabv3 +

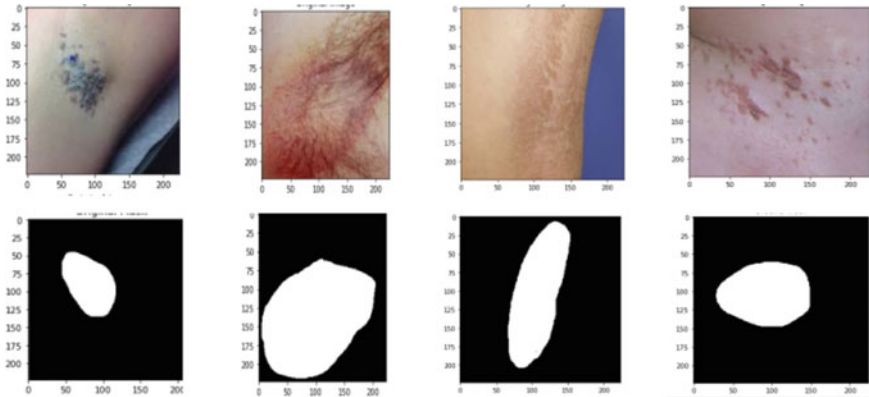


Fig. 3 Granular parakeratosis segmentation using *U-Net*

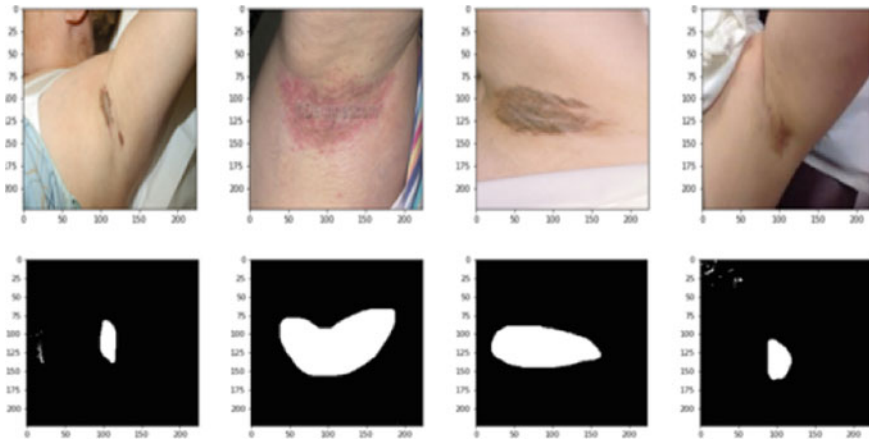


Fig. 4 Granular parakeratosis segmentation using FCN, SegNet and DeepLabv3 +

5.1 Dataset

A dataset is made up of several pieces of data that are used to train the model to uncover predictable patterns throughout the entire dataset. Datasets are essential for the advancement of numerous computational domains, providing results with scope, robustness and confidence [27, 28]. “With the advancement of machine learning, artificial intelligence and deep learning, datasets have become popular.” This paper retrieves the dataset from freely available resources like DermnetNZ. DermNet is a dataset available worldwide consisting of RGB images of granular parakeratosis. Here, the skin lesion images, as well as the mask image, are converted to BMP format. The images have been categorized into 1080 training set, 147 test set and 278 validation set, respectively. All the images are initially converted to $224 \times 224 \times 3$ pixels and loaded into the model, thereby obtaining the segmented images.

5.2 Evaluation

The methods are evaluated using accuracy, sensitivity and specificity metrics. This is computed as

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FP}}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

Abbreviations: TP—true positive, TN—true negative, FP—false positive, FN—false negative

The performance of the model is evaluated using the metrics mentioned in this section and compared with several existing methods like FCN, SegNet and DeepLabv3 + . It is observed that the proposed method resulted in an accuracy of 96.71%, sensitivity of 100%, and specificity of 91%, resulting in outstanding performance compared to FCN, SegNet and DeepLabv3 + as shown in Fig. 5, and the corresponding values are given in Table 1.

By applying the binary cross-entropy loss function to many segmentation algorithms, including FCN, SegNet and DeepLabv3 + , and comparing them to the suggested model, the relevance of the binary cross-entropy loss function can be demonstrated. The loss values obtained for the proposed model and the existing algorithms are shown in Table 2. The graph displayed in Fig. 6 is created using the values specified in this Table 2

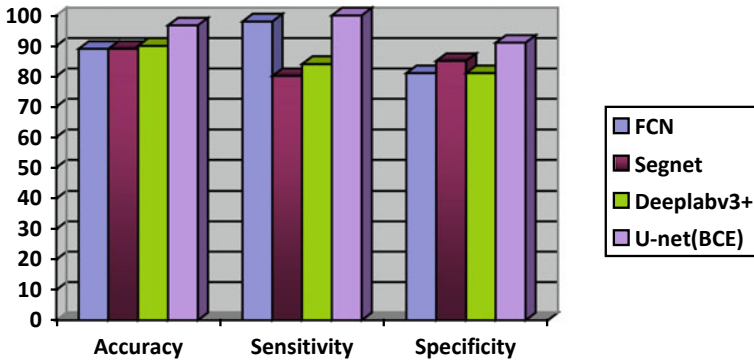


Fig. 5 Performance comparison of FCN, SegNet, DeepLabv3 + and U-Net with BCE

Table 1 Accuracy, SENSITIVITY, AND SPECIFICITY VALUES for FCN, SegNet, DeepLabv3 + and U-Net with BCE

	FCN	SegNet	DeepLabv3 +	U-Net(BCE)
Accuracy	89	89	90	96.71
Sensitivity	98	80.05	84	100
Specificity	81	85	81	91

Table 2 Values obtained for binary cross-entropy for FCN, SegNet, DeepLabv3 + and U-Net with BCE

	FCN	SegNet	DeepLabv3 +	U-Net (BCE)
Binary cross-entropy loss	0.29	0.89	0.81	0.21

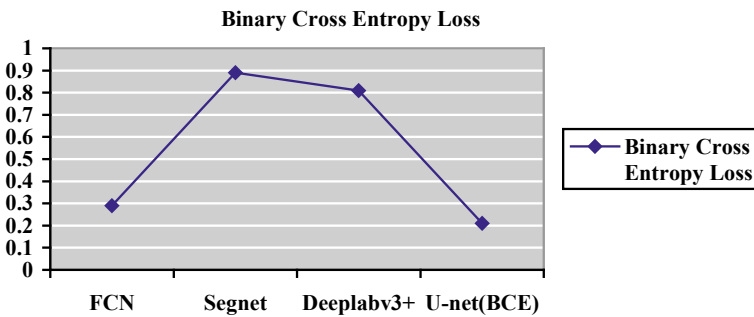
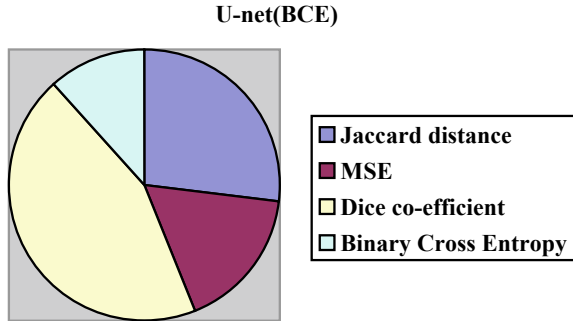


Fig. 6 Significance of binary cross-entropy on the proposed method and existing methods

Table 3 Values obtained for several loss functions

	Jaccard distance	MSE	Dice coefficient	Binary Cross-entropy
<i>U-Net</i> (BCE)	0.49	0.31	0.81	0.21

Fig. 7 Several loss functions implemented on the proposed method



In the field of machine learning, several loss functions are available, such as the Jaccard distance, mean squared error (MSE), dice coefficient and binary cross-entropy loss function. Out of these, the binary cross-entropy loss function is proved to provide minimal loss value. The proposed method is implemented using all of these loss functions, and the corresponding values obtained are shown in Table 3, and these values are used to draw the graph shown in Fig. 7.

5.3 Comparing to the State-Of-The-Art

Finally, we compare BCE-based *U-Net* to the state-of-the-art techniques, including ensembled deep network architecture and probabilistic deep learning models, in literature. Table 4 gives an overview of an accuracy, sensitivity and specificity values of these approaches and the proposed method. It is worth noting that a direct comparison of the two approaches is not entirely fair because they differ in many ways, not just the kernel. None the less, the results of the proposed model are excellent to the best of our knowledge since it achieved the highest accuracy and sensitivity on lesions of granular parakeratosis. The comparison results are the outcome of algorithm implementations.

6 Conclusion

This paper gives a brief discussion of the BCE-based *U-Net* model’s implementation as well as comparison of segmentation approaches such as FCN, SegNet,

Table 4 Comparison of BCE U-Net with state-of-the-art techniques

References	Accuracy (%)	Sensitivity (%)	Specificity (%)
[7]	93	90	93
[8]	94.32	88.76	–
[9]	88.78	91.88	89.42
U-net with BCE	96.71	100	91

DeepLabv3 + with the proposed model. The goal of this research is to give a comparison of such well-known existing image segmentation methods with the BCE-based U-Net technique in order to determine which method is best for medical picture segmentation. When compared to the ensemble and probabilistic-based state-of-the-art approach, the accuracy and sensitivity of the results found in BCE-U-Net are substantially greater. BCE U-Net obtained an accuracy of 96.71%, sensitivity of 100% and specificity of 91%.

References

1. Kumarasinghe SP, Chandran V, Raby E, Wood B (2020) Granular parakeratosis is a reaction pattern in hyperkeratotic flexural erythema. *Australas J Dermatol* 61(2):159–160. <https://doi.org/10.1111/ajd.13216>
2. Uzma Jamil SK (2015) Valuable pre-processing & segmentation techniques used in automated skin lesion detection systems. *Int Conf Model Simul* 290–295. <https://doi.org/10.1109/UKSim.2015.24>
3. Afandi A, Isa IS, Sulaiman SN, Marzuki NNM, Karim NKA (2020) Comparison of different image segmentation techniques on MRI image. *Smart Innov Syst Technol* 165(11):1–9. https://doi.org/10.1007/978-981-15-0077-0_1
4. Chen S, Wang Z, Shi J, Liu B and Yu N (2018) A multi-task framework with feature passing module for skin lesion classification and segmentation. In: 2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018), vol 2018-April, no Isbi, pp 1126–1129. <https://doi.org/10.1109/ISBI.2018.8363769>
5. Bi L, Kim J, Ahn E, Feng D (2017) Automatic skin lesion analysis using large-scale dermoscopy images and deep residual networks, pp. 6–9 [Online]. Available. <http://arxiv.org/abs/1703.04197>
6. Yang X, Li H, Wang L, Yeo SY, Su Y, Zeng Z (2018) Skin lesion analysis by multi-target deep neural networks. In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), vol 2018-July, pp 1263–1266. <https://doi.org/10.1109/EMBC.2018.8512488>
7. Matsunaga K, Hamada A, Minagawa A, Koga H (2017) Image classification of melanoma, nevus and seborrheic keratosis by deep neural network ensemble, pp 2–5 [Online]. Available. <http://arxiv.org/abs/1703.03108>
8. Mustafa ID, Hassan MA (2016) A comparison between different segmentation techniques used in medical imaging. *Am J Biomed Eng* 6(2):59–69. <https://doi.org/10.5923/j.ajbe.20160602.03>
9. Adegun AA, Viriri S, Yousaf MH (2021) A probabilistic-based deep learning model for skin lesion segmentation. *Appl Sci* 11(7). <https://doi.org/10.3390/app11073025>

10. Khan MA et al (2018) An implementation of normal distribution based segmentation and entropy controlled features selection for skin lesion detection and classification. *BMC Cancer* 18(1):1–21. <https://doi.org/10.1186/s12885-018-4465-8>
11. Sarma R, Gupta YK (2021) A comparative study of new and existing segmentation techniques. In: *IOP conference series: materials science and engineering*, vol 1022, no 1. <https://doi.org/10.1088/1757-899X/1022/1/012027>
12. Ünver HM, Ayan E (2019) Skin lesion segmentation in dermoscopic images with combination of yolo and grabcut algorithm. *Diagnostics* 9(3). <https://doi.org/10.3390/diagnostics9030072>
13. Liu L, Tsui YY, Mandal M (2021) Skin lesion segmentation using deep learning with auxiliary task. *J. Imaging* 7(4). <https://doi.org/10.3390/jimaging7040067>
14. Bi L, Kim J, Ahn E, Kumar A, Fulham M, Feng D (2017) Dermoscopic image segmentation via multistage fully convolutional networks. *IEEE Trans Biomed Eng* 64(9):2065–2074. <https://doi.org/10.1109/TBME.2017.2712771>
15. Beaulah Jeyavathana R, Balasubramanian R, Pandian AA (2016) A survey: analysis on pre-processing and segmentation techniques for medical images. *Int J Res Sci Innov III*(June):2321–2705
16. Mahmooud M, Alamin TI, Esmail MY (2020) EasyChair preprint implementation and comparison of different segmentation techniques for MRI and CT images
17. Shridhar M, Sethi AS, Ahmadi M (1986) Image Segmentation: a comparative study. *Can Elect Eng J* 11(4):172–183. <https://doi.org/10.1109/CEEJ.1986.6591942>
18. Huang L, Gong Zhao Y, Jun Yang T (2019) Skin lesion segmentation using object scale-oriented fully convolutional neural networks. *Signal Image Video Process* 13(3):431–438. <https://doi.org/10.1007/s11760-018-01410-3>
19. Understanding and implementing a fully convolutional network (FCN) | by Himanshu Rawlani | Towards Data Science. <https://towardsdatascience.com/implementing-a-fully-convolutional-network-fcn-in-tensorflow-2-3c46fb61de3b> (Accessed Oct 08 2021)
20. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.* pp 1–14
21. Badrinarayanan V, Kendall A, Cipolla R (2017) SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(12):2481–2495. <https://doi.org/10.1109/TPAMI.2016.2644615>
22. Ninh QC, Tran TT, Tran TT, Anh Xuan Tran T, Pham VT (2019) Skin lesion segmentation based on modification of SegNet neural networks. In: *2019 6th NAFOSTED conference on information and computer science (NICS)*, pp 575–578. <https://doi.org/10.1109/NICS48868.2019.9023862>
23. DeepLabV3+ | papers with code. <https://paperswithcode.com/model/deeplabv3-1?variant=deeplabv3-r101-dc5-1> (Accessed Oct 08 2021)
24. Goyal M, Oakley A, Bansal P, Dancey D, Yap MH (2020) Skin lesion segmentation in dermoscopic images with ensemble deep learning methods. *IEEE Access* 8:4171–4181. <https://doi.org/10.1109/ACCESS.2019.2960504>
25. Weng W, Zhu X (2021) INet: convolutional networks for biomedical image segmentation. *IEEE Access* 9:16591–16603. <https://doi.org/10.1109/ACCESS.2021.3053408>
26. Zafar K et al (2020) Skin lesion segmentation from dermoscopic images using convolutional neural network. *Sensors (Switzerland)* 20(6):1–14. <https://doi.org/10.3390/s20061601>
27. Pernambuco BSG, Steffens CR, Pereira JR, Werhli AV, Azzolin RZ, Da Silva Diaz Estrada E (2009) Online sound based arc-welding defect detection using artificial neural networks. In: *2019 Latin American robotics symposium (LARS), 2019 Brazilian symposium on robotics (SBR) and 2019 workshop on robotics in education (WRE)*, pp 263–268. <https://doi.org/10.1109/LARS-SBR-WRE48964.2019.00053>
28. Dekker R (2006) The importance of having data-sets. In: *IATUL Annu Conf Proc.*, vol. 16, pp. 89–92, 2006, [Online]. Available. <internal-pdf://dekker-2006-datasets-0862637824/dekker-2006-datasets.pdf>

Real-Time Health Monitoring of Relays and Circuit Breakers



S. Harshitha, B. S. Arpitha, H. Shwetha, N. Sinchana, B. Smitha,
and M. J. Nagaraj

1 Introduction

It is important to monitor the relays and circuit breakers which are working to reduce the power outage and the expensive charges for the damages that are caused after the failure [1]. To maintain the reliability and the stability, it is necessary to protect the equipment from abnormal conditions. Relay and Circuit breakers have a service life when operated under rated conditions. When abnormal conditions occur, the life of the equipment reduces significantly [2]. To maintain the health of the relays and circuit breakers, first the parameters need to be identified that cause the problem to prevent the faults. When the parameters are exceeded more than rated, then the error message and precaution message are sent using the GSM module. This GSM module helps to get the message on the registered on the mobile number to take up the precautionary measures. While doing the hardware circuit, the GSM module can be replaced with the Wi-Fi module along with the smart cloud technology where we get the message on the application on the smart phone. The data logs can also be found for future reference.

S. Harshitha (✉) · B. S. Arpitha · H. Shwetha · N. Sinchana · B. Smitha · M. J. Nagaraj
Department of EEE, NMIT, Bangalore 560064, India
e-mail: harshithasuresh2299@gmail.com

B. Smitha
e-mail: smitha.b@nmit.ac.in

M. J. Nagaraj
e-mail: nagaraj.mj@nmit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_40

2 Background and Industrial Scenario

Status monitoring is a big part of guessing the maintenance. Data collected from status monitoring over time provides important information about the current status and history of the equipment. This information about the equipment can be used to predict how the product will perform over time and how it may undermine it, allowing maintenance to be adjusted according to these predictions [3]. For analyzing the failures of the equipment, intelligent decision-making system is used. In present scenario, industries are using SCADA and PLC to get the status of the circuit breakers and relays.

3 Related Work

In condition-based maintenance, for the relay protection unit, it is difficult to find the performance degradation process of a protection unit and it decreases [4]. The master station for monitoring collects self-test information from the protection unit and forecasts the health status in real time using Microprocessor [5].

To check the remaining useful life of the electromagnetic relays, the parameters such as voltage, temperature, duty cycle, and frequency are identified and monitored. X-rays are used to monitor the failure modes, and the results show the low-resolution data [6].

The mechanical and electrical view of CB operation and cost optimization techniques can be implemented. With the fast progress of technology, monitoring online is practically used in substations and gives the information about circuit breaker parts [7].

The analysis of current trends and future trends for the real-time health monitoring helps to get the proper status of the equipment. Different analyses such dissolved gas analysis, vibration analysis, and trip coil current analysis are explained which can be implemented [8].

4 Simulation

Proteus is used to simulate design and drawing of electronic circuits. Proteus Virtual System Modeling (VSM) combines mixed mode SPICE circuit simulation, animated components, and microprocessor models to facilitate co-simulation of complete microcontroller-based designs [9]. Designing a circuit on proteus consumes less time compared to practical construction of circuit. There is no possibility of damaging any electronic component in proteus. The software contains PCB designing as well. The two main parts of Proteus are to design or draw circuits and designing the PCB layout. ISIS is used for designing and simulating circuits. ARES is used for designing

a Printed Board Circuit. ISIS has a wide range of components in the library. ARES offers PCB designing with surface mount packages [10].

5 Establishment of Simulation System

5.1 Simulation

The circuit is designed in Proteus, and flow of simulation is programmed using an Arduino IDE, which is an open-source software for writing and compiling the code into Arduino Module. Figure 1 shows the block diagram of the simulation circuit.

The main code written on the IDE platform will generate a Hex file which is then transferred and uploaded to the controller on the board.

In the design, a microcontroller such as Arduino is used as the main part of control section. Current and temperature sensors are interfaced to microcontroller which collects the temperature and current data of the equipment, thereby passing the information to microcontroller GPIO pin.

A switch/push button is used to turn ON/OFF the relay, and every ON/OFF action on relay is noted and is given as input to the controller which gives the information of relay count. GSM module is interfaced to microcontroller that receives timely updates of action performed on relay. LCD display is used as output displaying all the collected data.

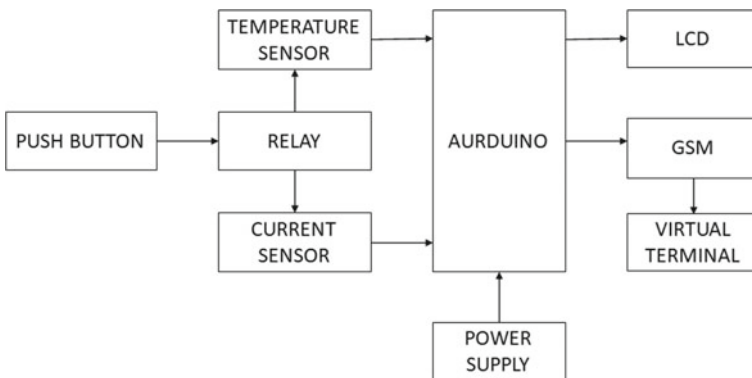


Fig. 1 Block diagram

5.2 Analysis of Simulation Results

Initially, relay is under OFF condition, once power is turned ON a message “Starting, Please Wait” is displayed on LCD. A delay of 1 s(1000 ms) is set to power the circuit. Figure 2 shows the circuit connections of the simulation. A switch keeps a track of relay ON/OFF condition. As soon a switch is pressed, it is considered as one count, and the same count value is stored in controller. The count gets incremented after every press on the switch displaying the same value on LCD.

A permissible limit value is set to the controller, if the switch is pressed more than the limit value an error message of is displayed on LCD as shown in Fig. 3. This is a warning message indicating relay has reached its maximum usage; hence, the relay needs to be replaced. If the user ignores this warning message, it will lead to failure of the equipment.

An LM35 sensor and a current sensor acquires temperature and current value of load under normal healthy condition. The temperature data which is obtained is an analog value; therefore, a conversion is done to display the value in standard format.

In simulation, lamp load is used to keep a track on current consumption. Fault message is displayed in case of excess current consumption as shown in Fig. 3. The message is displayed on LCD and a virtual terminal of GSM module. This condition is achieved when resistance of lamp load is decreased. This message gives a warning

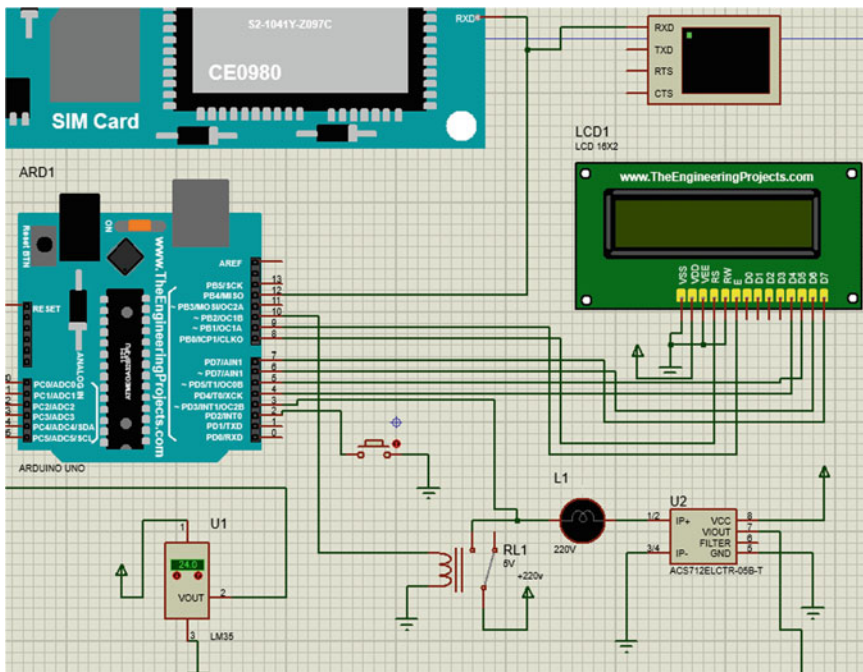


Fig. 2 Circuit implementation

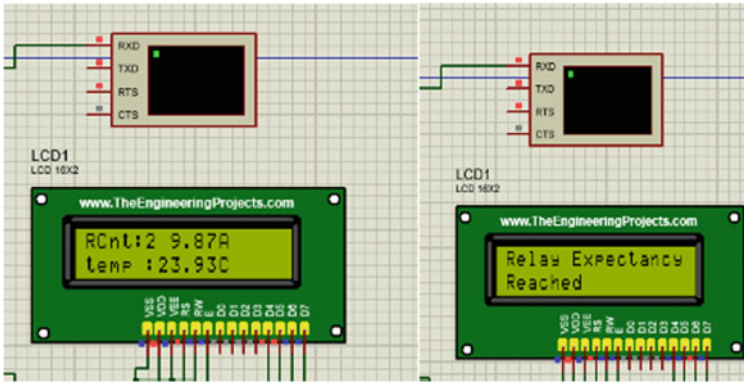


Fig. 3 Results obtained

that the circuit is under short-circuited condition and hence the fault needs to be cleared for safer operation of equipment.

6 Hardware Development

6.1 Circuit Analysis

Hardware system is implemented using an ESP232. ESP32 is Wi-Fi built-in micro-controller suitable for IoT projects with monitoring and controlling general purpose input and output devices. After the controller is powered up, all the necessary packages are imported, and serial port is enabled for debugging and controller will first connect to hotspot with SSID and password-provided Wi-Fi fields.

A regulator is used to step down the voltage to a desired level used for powering ESP32 board as shown in Fig. 4. Wi-Fi is used as a connecting media for ESP32 board. A switch is used to simulate the MCB other parameter such as temperature, voltage, and current is monitored in case of excess current consumption a fault message is displayed, and if set limit count of the MCB, is exceeded then a message is displayed on LCD to warn that life expectancy has reached the limit and need to be replaced.

Once ESP32 is connected to the hotspot, the timer is initiated for a set timer interval. The data is also stored in cloud using a Blynk application, and Blynk is an application that works over Internet used for storing data. Data in cloud can be retrieved or downloaded for further analysis.



Fig. 4 Block diagram of hardware

6.2 Analysis of Hardware Results

The electrical life cycle of the relays is normally expected up to 100,000 cycles. It varies with the type of relay. The count of the relay increases as it opens and closes. Here, the relay life cycle count is set up to 5. Once 5 counts are reached, it will show the message “Relay Life Expectancy Reached” on the LCD as shown in Fig. 5. The same can be seen on the status bar on the Blynk app.

A short circuit is created by shorting a sensor since we cannot short two wires directly in electronic circuits. Shorting two wires directly causes the damage to the components. The resistance variation in the circuit is taken into consideration for detecting the fault. Whenever a short circuit happens, a message “Short circuit detected” is displayed both on LCD and Blynk app.

Relays operate under specific temperature. Whenever there is an increase in temperature, the contacts may get damaged. Here, we have set the temperature to 35 °C limit. The temperature sensor measures the temperature. Increase in the

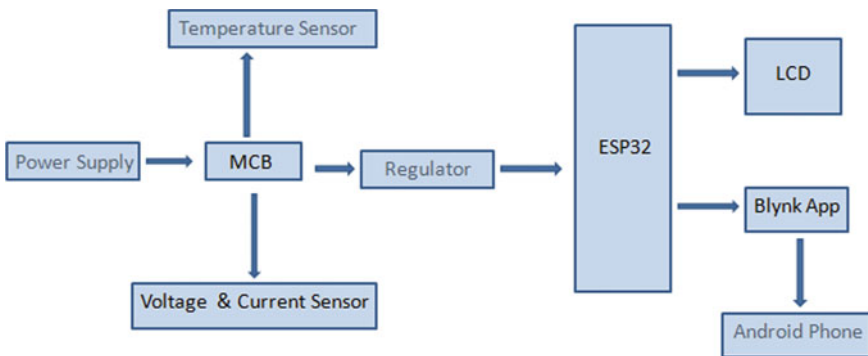


Fig. 5 LCD display for relay expectancy

temperature beyond this limit will be considered as a fault, and the error message similar to Fig. 5 will be printed on LCD and even on the Blynk app.

7 Conclusion and Future Scope

In this paper, some of the monitoring parameters such as temperature, current, voltage, and humidity along with relay count so that the life of the relay can be known. When the short circuit is occurred, it is detected. All the detections of temperature, voltage, humidity, current, relay count, and short circuit fault are displayed over LCD. And connecting this monitoring system to the GSM module through which receiving the respective message that is displayed on the LCD over the registered mobile number is achieved.

Monitoring trip coil current and reducing corrosion in the contacts can be implemented in the hardware. And life of the coil can also be predicted if the operating temperature of the coil is known. Hardware model for this simulation with added trip coil measurement and predicting life of the coil can be implemented so that it helps for the use of industrial purpose and household purposes where we can monitor the life and operation of the protection devices. A GSM module is used in the software, instead of this we can use a Wi-Fi module to store data with cloud technology. A Blynk application can be used instead of GSM module, where data that is stored in Blynk application can be used to develop a machine learning model.

References

1. Kharat B, Sarwade D, Bidgar D, Kadu B (2017) Internet of Thing (I.O.T) base controlling & monitoring of circuit breaker. *Int Res J Eng Technol (IRJET)* 04(05)
2. Dalke G (2005) Application of numeric protective relay circuit breaker duty monitoring. *IEEE*
3. Melli SA, Nadian A, Amini B, Asadi N (2011) Design of online circuit breaker condition monitoring hardware. In: *IEEE 2nd international conference on control, instrumentation and automation*, 27–29 Dec 2011
4. Huang S, Chen S, Qiu Y, Ye Y, Pan W (2011) Online condition monitoring methodology for relay protection based on self-test information. *IEEE* 16–20 Oct 2011
5. Kirschbaum L, Dinmohammadi F, Flynn D, Robu V, Pecht M (2018) Failure analysis informing embedded health monitoring of electromagnetic relays. *IEEE* 23–25 Nov 2018
6. Feizifar B, Usta O (2017) Condition monitoring of circuit breakers: current status and future trends. In: *2017 IEEE international conference on environment and electrical engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*
7. Byerly JM, Schneider C, Schloss R, West I (2017) Real-time circuit breaker health diagnostics. In: *IEEE 2017 70th annual conference for protective relay engineers (CPRE)* 3–6 Apr 2017
8. Xiao F, Zhang Z, Yin X, Ji J, Chen G (2015) Study of maintenance strategy of relay protection system based on condition monitoring. *IEEE*, 1–4 Sept 2015
9. Proteus-design {available online}. https://en.wikipedia.org/wiki/Proteus_Design_Suite/ (Accessed on 29 June 2021)
10. About Proteus software {available online}. <https://www.theengineeringknowledge.com/introduction-to-proteus/> (Accessed on 29 June 2021)

AI-Based Live-Wire News Categorization



S. Jagdeesh Patil, Aashna Sinha, M. M. Anusha Jadav, and Nidhi

1 Introduction

The above objectives can be achieved by creating a model using machine learning algorithms, which classifies the live data into different categories like Sports news, Entertainment, Political, Health, Business, Science and Technology. Multinomial Naive Bayes model is used for the classification of news. Multinomial Naive Bayes is a probabilistic learning method that is used for the analysis of categorical text data. The training is done using the UCI-NEWS aggregator Dataset, which consists of around 4L records. And testing is performed using live news from news API, which contains 100 records. The model gave an accuracy of 92%. The front-end user interface is a web application using Flask in Python, HTML, and CSS.

Developing and deploying news categorization system [1] based on interested content of news-by-news broadcaster are difficult task. To solve this challenging task, an attempt has been made to conceive and develop news categorization system. Major aspects of the news categorization system are presented in this paper.

2 System Architecture

In designing news categorization system [2], the following inputs are taken into account.

- Multinomial Naive Bayes algorithm: This algorithm is a probabilistic learning approach that is commonly used in natural language processing. The Bayes theorem underpins the algorithm.

S. Jagdeesh Patil (✉) · A. Sinha · M. M. Anusha Jadav · Nidhi
Department of ISE, Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: jagadish.patil@nmit.ac.in

- API using Flask: Flask is a small web application framework. It is intended to be simple and quick to get started, with the capacity to scale up to sophisticated applications.

The basic components of news categorization system are depicted in Fig. 1. As seen in here, the news categorization system [4] consists of the following functional building blocks: (a) news categorization—business module, (b) application module, and (c) management console module. A brief explanation of these modules is given below (Figs. 2 and 3).

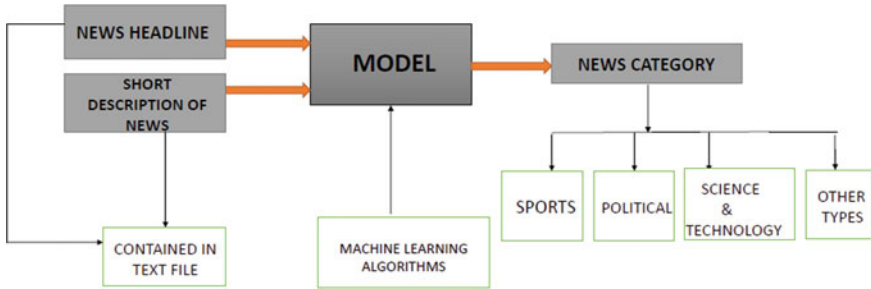


Fig. 1 News categorization system design

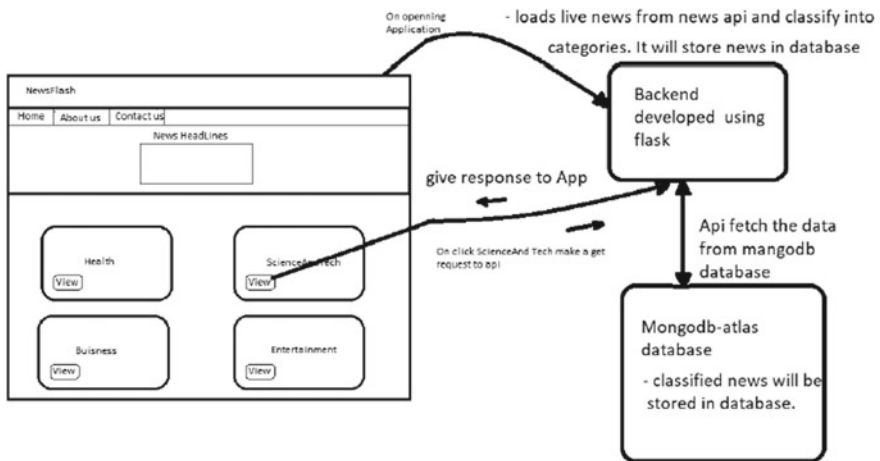


Fig. 2 News categorization system workflow

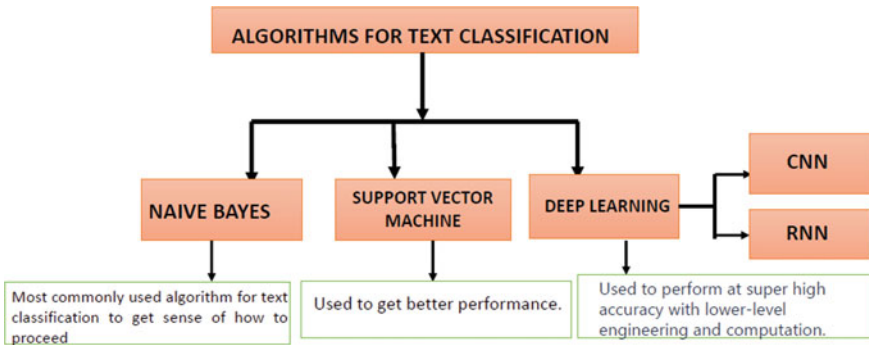


Fig. 3 Algorithms used for news text classification

2.1 Management Console Module

The management console is the module to configure the news categorization—business module with news category parameters. The news categorization model takes these inputs and process further. Business model can be configured combination of following categories of news of their interest and weightages:

- Sports news
- Entertainment
- Political
- Health
- Business
- Science and Technology

2.2 Business Module

The data module consists news categorization model definitions and collects live news data from live-wire using API on live-wire content. The news content is categorized into various categories based on management console inputs and stores data into various baskets in the back-end server.

2.3 Application Module

The application module provides news categorization [5] to user of this system. It is an interface to provide categorization generated in the business module to end-user in regular intervals via web module.

3 Software Implementation

The news categorization approach [6] is based on:

3.1 *NewsAPI*

It is an HTTP, rest API meant for searching and fetching live articles. It provides live articles and historic articles and created an API key to get data.

3.2 *Text Preprocessing and Label Encoding*

The data is preprocessed by converting data to lower case, removing the links, numbers, and punctuation. Data preprocessing is a very important step in our application, which reduces the dimensionality of vectors. We encounter a multi-class problem. The class name is categorical [3], so they are human-readable. Label encoding is a technique that converts categorical data to numerical values. In our case, it converts categories' names into numbers. It assigns labels with a value from zero to classes-1 where classes are a number of distinct classes.

3.3 *Count Vectorizer and TF-IDF Vectorizer*

Count vectorizer converts a text into vector based on the count of each word in the entire text. This creates a matrix, where the columns are unique words and the row is a document. Inside the count vectorizer, the column is not stored as a string. But they are given with a particular index. An object of CountVectorizer() is created and converted value into numerical using fittransform(). TF-IDF vectorizer will calculate relevance of a word to a document, in a set of documents, which is calculated by multiplying the TF of a word in a document and IDF of the word across the document. If the word is common in many documents, then the number will reach 0. Otherwise, it will be 1.

3.4 *Multinomial Model*

Multinomial model is a probabilistic learning method, which is used in natural language preprocessing and for problems with multiple classes. It predicts the class of a text. It calculates the probability of each class belongs to that particular class. It

is based on the Bayes theorem. Inbuilt, it uses Laplace smoothing technique which solves the zero-probability problem.

3.5 *Flask API and MongoDB Atlas*

It is a web application framework which gives libraries to develop web application in Python. It helps to create.

APIs, which are fast. We can handle the API using HTTP libraries. The MongoDB is a cloud database service for any application. It uses Aws, Google cloud, and Azure to deploy fully managed MongoDB, which guarantees availability, scalability, and compliance.

3.6 *Pseudocode*

Algorithm 1: Algorithm of model

Input: data

Output: model

Step 1: Load the dataset

Step 2: Apply the text normalization to the data, which remove capital letter, number and so oil

Step 3: Perform label encoder to the categorical classes

Step 4: Apply Count vectorizer and dump the result to pickle file

Step 5: Apply TF-IDF vectorizer and dump the result to pickle file

Step 6: Train the model and dump the model to pickle file

Algorithm 2: Algorithm to get the live headlines

Input: the live data

Output: display top headings

Step 1: Get the live news from newapi

Step 2: Convert data to json

step 3: Convert the data to Dataframe and to records

Step 4: Create collection object and insert records into collection

Step 5: Get the data from the collection of news Database

Step 6: Append the data to list and Zip it. return to backend

Algorithm 3: Algorithm for Classify live data and store in db

Input: live data from news api

Output: classify the live data a store it

Step 1: Get the live news from newapi

Step 2: Convert data to json

Step 3: Convert the data to Dataframe and to records

Step 4: Create collection object and insert records into collection

Step 5: Get the data from the collection of news Database

Step 6: Apply the text normalization to the data, which remove capital letter,number and punctuation

Step 7: Load count vector, TF-IDF vector and model. And predict the classes for given data

Step 8: Create a data frame for each category

Step 9: Check which category rows in data belong to and append them to that data frame

Step 10: Create collection of each category and insert the records

Algorithm 4: Algorithm to get particular category data

Input: The name of category

Output: Articles under that category

Step 1: Check which category and fetch the data from collection

Step 2: Convert the data to json

Step 3: Create list of news,dsc,img and url

Step 4: Iterate through the data and append to list

Step 5: Zip all the list and Return data to the frontend

4 Testing and Field Trails

It used UCI news aggregator to get the data, then for the preprocessing part, data is divided into 80:20 ratio as shown in Fig. 4. Dataset contains 4 (four) lakh rows of data, and the accuracy is obtained. The model gave an accuracy of 92 percent. Evaluation is done on algorithm's performance in identifying the test data with varied sizes and diverse training data in terms of content.

It is experimented with algorithms, support vector machine, multilayer perceptron. Unlike SVM and MLP, Naive Bayes considers all characteristics to be mutually independent, ignoring co-relationships that might be crucial to the dataset. Multinomial Naive Bayes shows better accuracy than the former algorithms (Figs. 5 and 6).

5 Conclusion and Future Scope

- The model that is built by applying multinomial Naive Bayes algorithm categorizes the data that we collect from different sources of news using NewsAPI into

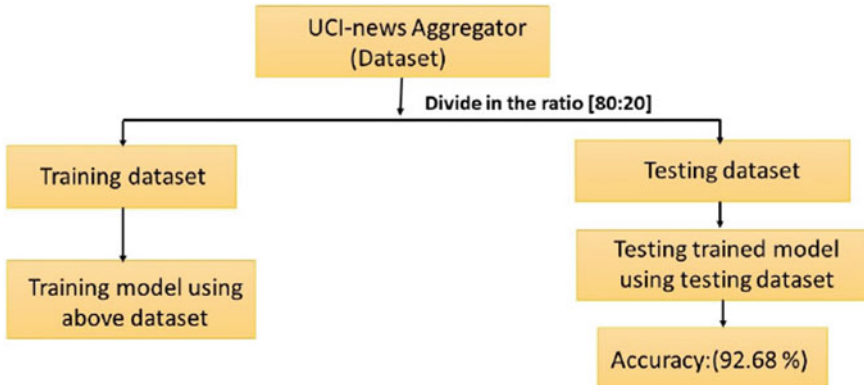


Fig. 4 News categorization system dataset division for training and testing and accuracy

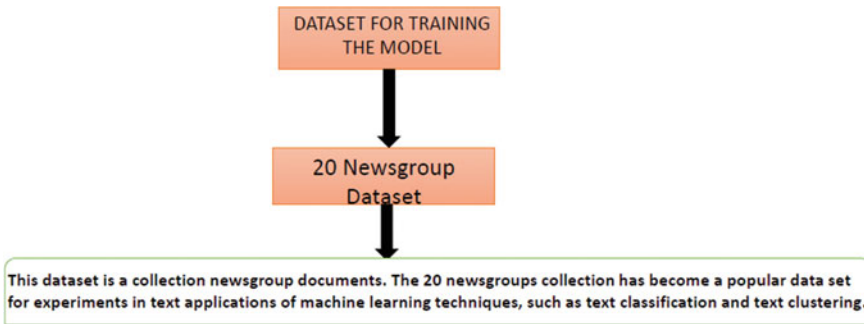


Fig. 5 Dataset used for train the model

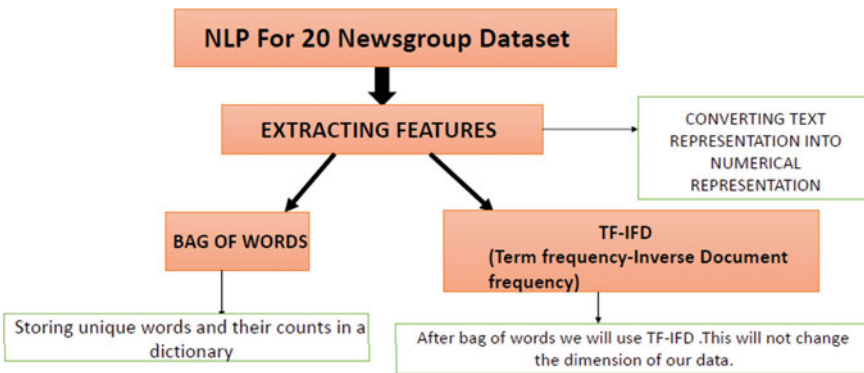


Fig. 6 Data flow

different categories which include Health, Science and Technology, Business, Political, and Entertainment.

- The news articles belonging to particular category can be accessed using the options provided in the website built which also gives the users the functionality to give feedback for further improvement of the website for the future use.

As a future scope:

- Add additional subcategories of the categories already included and provide news articles under them.
- Add news editor login to edit the news before being accessed by the print media and for broadcast in order to provide authorized news by the chief editor of the print media or broadcast, television, etc.

References

1. Patro A, Patel M, Shukla R, Save DJ (2020) Real time news classification using machine learning. *Int J Adv Sci Technol*. Dept. of Information Technology, Fr. Conceicao Rodrigues college of Engineering, Mumbai, India
2. Tsai C-W (2017) Real-time news classifier search engine architecture final report
3. Sivakami M, Thangaraj M (2018) Text classification techniques: a literature review. *Interdisc J Inf Knowl Manage*
4. Tang X, Xu A (2016) Multi-class classification using kernel density estimation on k-nearest neighbours. *Electron Lett*
5. Jivan NE, Yousefi KS, Fazeli M (2010) New approach for automated categorizing and finding similarities in online Persian news. In: *Technological convergence and social networks in information management, international symposium on information management in a changing world, 2010*
6. Suleymanov U, Rustamov S (2018) Automated news categorization using machine learning methods, vol 459. IOP Publishing, p 012006. [Online]. Available, <https://doi.org/10.1088/1757-899x/459/1/012006>

Implementation of STFT for Auditory Compensation on FPGA



S. L. Pinjare , B. R. Rajeev , Kajal Awasthi , and M. B. Vikas 

1 Introduction

Hearing aids are used to compensate for the hearing loss. Hearing loss may occur in outer ear or middle ear (conductive loss) or inner ear (sensorineural hearing loss). Audibility loss and a reduction in the dynamic hearing range are common symptoms of hearing loss. Most common treatment for hearing loss is use of hearing aids. The most important part in hearing aid is the auditory compensation block. It is responsible for compensating for the hearing loss of the patient. In hearing aids, a microphone picks up sounds, the amplifier makes it louder and the receiver after receiving from amplifier sends it to the ear. Frequency structuring and dynamic range correction are two functions that the auditory compensation provides. Former compensates the frequency-dependent loss; it provides more flexibility to meet the requirement of hearing-impaired people to increase the gain level of the audio signal. The latter adjusts the received signal's range in the residual dynamic range. The amount of auditory compensation required is determined by the audiogram which is obtained by performing an audiometry test on the subject. The audiogram depicts the kind, severity and pattern of hearing loss. Based on the patient's audiogram, the amplification required for a specific frequency band is established [1].

Researchers have so far implemented auditory compensation by using filter banks [2]. The general methodology adopted for auditory compensation has been using multi-channel filter banks [3–5]. This requires more filter coefficients. The number of banks is usually limited to 8 in order to strike a balance between the needed frequency response resolution, processing complexity, and signal delay caused by processing. The majority of hearing aids use digital technology which allows for more precise

S. L. Pinjare · B. R. Rajeev (✉) · K. Awasthi · M. B. Vikas
Department of Electronics and Communication Engineering, NITTE Meenakshi Institute of
Technology, Bengaluru, India
e-mail: rajeevz1999@gmail.com

programming and feedback control. The possibility of aliasing is present while digitizing analog signals. The rate of sampling must always be greater than twice of the highest frequency of the sampled waveform. Application of an anti-aliasing filter was also suggested [6]. Overall hearing aid performance is improved by adjusting gain function. The filter bank approach enables a variety of capabilities, such as voice coding and noise reduction, auditory compensation and sub-band coding. Noise reduction, echo cancelation, auditory compensation, and voice enhancement are all performed repeatedly in the DSP algorithm sub-band coding. The majority of designs use a fixed sub-band to minimize the filter bank. Hearing-impaired people are unable to proceed with certain cases in order to improve their auditory skills. Hence, it can be customized for each patient, but the main disadvantage is that there are fewer sub-bands. When FIR filters with extremely small transition bands are required, frequency response masking is a regularly used algorithm [7]. The filter bank approach has also higher power dissipation due to computational complexity. To reduce filter bank hearing aid power dissipation, a low computational complexity interpolated FIR filter bank was proposed, as well as a strategy to reduce the amount of power consumed due to multiplications, which are the filter bank's main operations [8–10]. The filter bank's power dissipation is further decreased with effective word length reduction of the coefficient of filter channels without compromising with the performance. The usage of the 1/3 octave filter bank in acoustic applications closely matches the frequency characteristics of the human ear. The high computational complexity and power consumption limit its applications. A novel approach using short-time Fourier transformation (STFT) is used to provide auditory compensation [11–13]. The Fourier transform of a tiny portion of the audio signal is performed. Using the quasi-stationary property of speech signals, the method of STFT for auditory compensation is being implemented which can overcome many of the drawbacks of the filter bank approach. STFT is a two-dimensional transformation that is calculated by separating the input signal into segments and calculating DFT for each segment using a sliding time-limited window, the dominant audio frequency is determined and the required correction is applied to the incoming signal and delivered to the subject through an earphone. There is no need of using a filter bank, thus reducing computational complexity. This eliminates the problem of signal reconstruction that exists with the filter bank approach, thus leading to very low power dissipation. The implementation of more number of bands is also straightforward, higher rate of sampling can be used, and instead of 16 point FFT, we can use 32 point FFT. This will increase the number of bands and can provide better correction. However, the power dissipation also will increase.

This work focuses on implementation of STFT for auditory compensation, a potential algorithm which could replace the current trend of filter bank approach with significantly lesser amount of logic which eventually leads to better performance, lower power consumption with a smaller footprint.

2 Design Approach and Methodology

2.1 Discrete Fourier Transform and Fast Fourier Transform

Fourier transform is a mathematical method to convert a function in the time domain to frequency domain for non-periodic functions. It is a mathematical tool which decomposes the signal into sum of sinusoidal components or complex exponential components.

Representation of Fourier transform is given by:

$$x(t) \xleftrightarrow{ft} X(iw) \quad (1)$$

where $x(t)$ is a signal when a Fourier transform is applied, $X(iw)$ is obtained with an unit rad/sec, where $X(iw)$ is a complex number. The Fourier transform from time domain to frequency domain is:

$$X(iw) = \int_{-\infty}^{\infty} x(t).e^{-i\omega t} dt \quad (2)$$

The inverse Fourier transform from frequency domain to time domain is:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(iw).e^{i\omega t} dw \quad (3)$$

For sampled signal, we can use discrete Fourier transform given as

$$\widehat{f}_k = \sum_{j=0}^{n-1} f_j e^{-i2\pi jk/n} \quad (4)$$

where n is length of input or number of input samples, f_j is the input sequence, k and j are variables which varies from 0 to $n-1$. The inverse discrete Fourier transform is written as:

$$f_k = \frac{1}{n} \sum_{j=0}^{n-1} \widehat{f}_j e^{i2\pi jk/n} \quad (5)$$

The fast Fourier transform, or FFT, is an algorithm that has been developed to compute the DFT in an extremely economical fashion. It is much faster because of the fact that it utilizes the results of previous computations to reduce the number of operations, employs divide and conquer approach. In particular, it exploits the periodicity and symmetry of trigonometric functions to compute the transform with approximately $N \log N$ operations. DFT can also be expressed as

$$F_k = \sum_{n=0}^{N-1} f_n W^{nk} \tag{6}$$

where W is a complex valued function defined as $W = e^{-i(\frac{2\pi}{N})}$. Now, we divide the sample in half and express equation in terms of the first and last $N/2$ points:

$$F_k = \sum_{n=0}^{(\frac{N}{2})-1} f_n e^{-i(\frac{2\pi}{N})kn} + \sum_{n=N/2}^{N-1} f_n e^{-i(\frac{2\pi}{N})kn} \tag{7}$$

where $k = 0, 1, 2, \dots, N-1$. A new variable, $m = n - N/2$, is introduced so that the range of the second summation is consistent with the first,

$$F_k = \sum_{n=0}^{(\frac{N}{2})-1} f_n e^{-i(\frac{2\pi}{N})kn} + \sum_{m=0}^{(\frac{N}{2})-1} f_{m+\frac{N}{2}} e^{-i(\frac{2\pi}{N})k(m+\frac{N}{2})} \tag{8}$$

Or

$$F_k = \sum_{n=0}^{(\frac{N}{2})-1} (f_n + e^{-ink} f_{n+\frac{N}{2}}) e^{-\frac{i2\pi kn}{N}} \tag{9}$$

Nothing that $e^{-ink} = (-1)^k$, we get for even values, i.e.,

$$F_{2k} = \sum_{n=0}^{(\frac{N}{2})-1} (f_n + f_{n+\frac{N}{2}}) e^{-\frac{i2\pi(2k)n}{N}} \tag{10}$$

$$= \sum_{n=0}^{(\frac{N}{2})-1} (f_n + f_{n+\frac{N}{2}}) e^{-\frac{i2\pi kn}{(\frac{N}{2})}} \tag{11}$$

And for odd values,

$$F_{2k+1} = \sum_{n=0}^{(\frac{N}{2})-1} (f_n - f_{n+\frac{N}{2}}) e^{-\frac{i2\pi(2k+1)n}{N}} \tag{12}$$

$$= \sum_{n=0}^{(\frac{N}{2})-1} (f_n - f_{n+\frac{N}{2}}) e^{-\frac{i2\pi n}{N}} e^{-\frac{i2\pi kn}{(\frac{N}{2})}}, \text{ For } k = 0, 1, 2, \dots, (N/2) - 1 \tag{13}$$

where f_n is an input sequence, N is the length, F_{2k} is the even output values and F_{2k+1} is the odd output values. On summing the both even and odd values, we get a desired output F_k . Fig. 1 depicts radix-2 8-point FFT butterfly diagram where combination

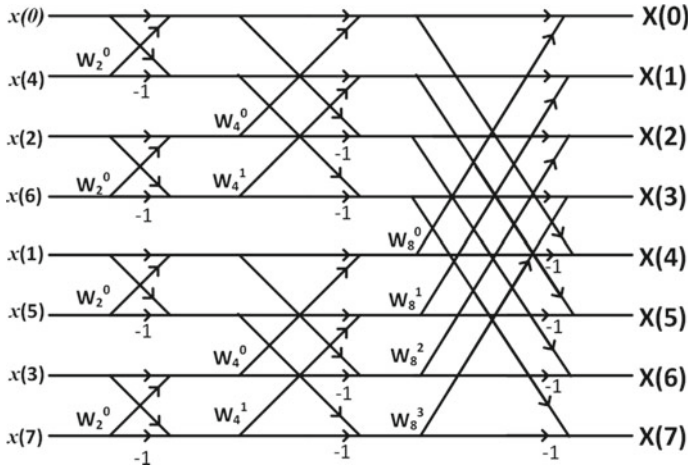


Fig. 1 Radix-2 8-point FFT butterfly diagram

of smaller DFTs combine to form larger DFT. The computation is broken down into three stages, the first stage with four 2-point DFTs, then two 4-point FFT and finally an 8-point DFT. For an N -point FFT, there are N butterflies per stage and $\log_2(N)$ stages. Each butterfly implies 1 complex multiplication and 2 complex additions. Hence, there are $(N/2) \cdot \log_2(N)$ multiplications and $N \cdot \log_2(N)$ additions.

2.2 STFT and the Algorithm for Auditory Compensation

Signal is considered stationary when its statistical properties do not change with respect to time. Motion of visual objects is lucid if frames are presented successively within an interval of 42 ms [14]. This stimulates the question: Whether the same paradigm can be applied to speech signals since they are quasi-stationary too? To analyze local frequency spectrum of an audio signal, particular number of samples (frames) can be collected periodically and converted to frequency domain using Fourier transform. This can be achieved with short-time Fourier transform. The window’s hop length in STFT can be used to separate the samples temporally and the window length can be used to define the number of samples in a frame. Essentially, STFT is DFT with input being controlled by a window function. Hence, for discretized audio signal $x[n]$, discretized time-limited window of length T $w[n]$, hop length D and i th frame x_i , we have:

$$x_i = x[n + iD] \cdot w[n], \quad \text{where } n = 0, 1, 2, 3, \dots, T - 1 \quad (14)$$

Fourier transform can be taken on these frames to analyze localized frequency behavior:

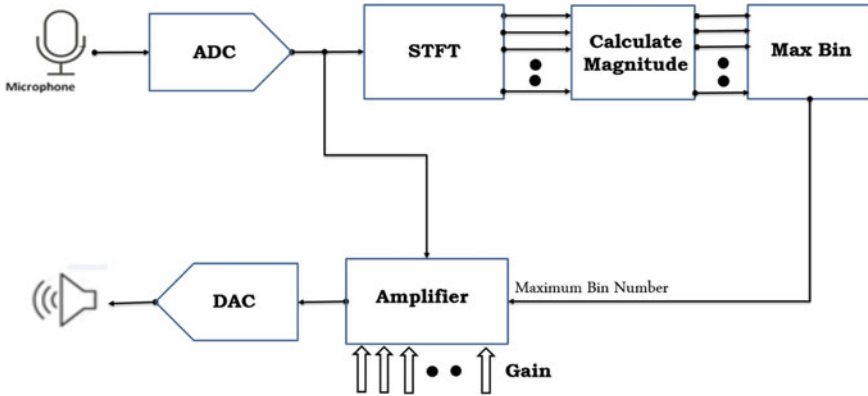


Fig. 2 Block diagram of STFT algorithm for auditory compensation

$$X(K) = \sum_{n=0}^{T-1} x_i(n)e^{-j\frac{2\pi}{T}kt}, \quad \text{where } k = 0, 1, 2, 3, \dots, T - 1 \quad (15)$$

The length of DFT is equal to the frame length, which in turn is equal to the window length. Since the Fourier transform is applied on discrete signals, T (length of DFT) discrete bins are obtained. When DFT of length T is applied on a signal sampled at f_s Hz, frequency of k th bin is given by:

$$f_k = \frac{k f_s}{T} \quad (16)$$

To sum up, Eqs. (14) and (15) collectively work as STFT and can be used to analyze local frequency spectrum of the input signal. Figure 2 represents the block diagram of STFT algorithm for auditory compensation. The input audio signal is sensed by the microphone and digitized using an ADC.

In the audiometry test, the subject’s audibility is tested for octave frequencies 250 Hz–8 kHz. Hence, the input audio must be sampled at 16 kHz at least to be able to recover the highest frequency. For the STFT, a window function must be provided as input apart from the input audio signal. Length of DFT is equal to length of the window since the frame length is limited by the window. Magnitude of the DFT’s outputs is taken and is fed to MAX bin block, where the index of the highest magnitude component is determined (dominant frequency for that frame). This index is then fed as input to the amplifier which is essentially a look-up table (LUT). The index obtained from MAX bin block is used to address the gain values stored in LUT, the gain values are loaded to the LUT beforehand based on the audiometry test done on the subject according to the respective bins. The gain value obtained from the LUT is multiplied with the input audio signal and is then given as input to DAC which eventually will drive the speaker. Since the time taken to process the frame (determine the dominant frequency component and its respective gain value)

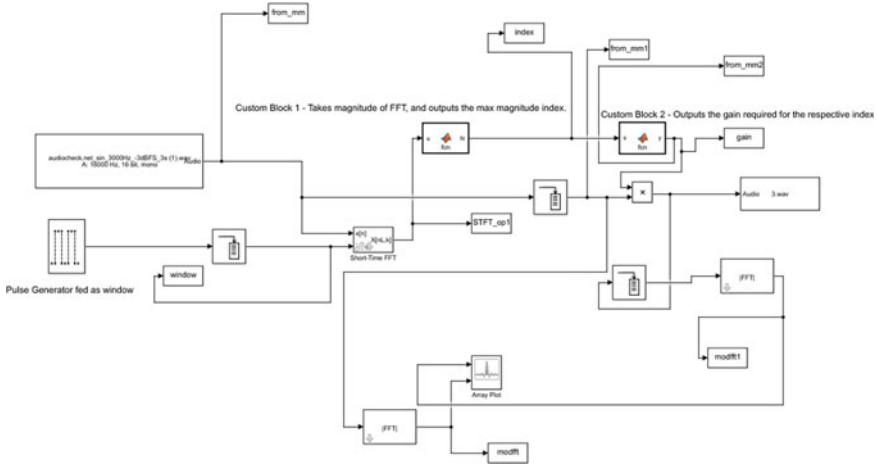


Fig. 3 STFT algorithm on Simulink for auditory compensation

is lesser than the quasi-stationary period of the speech signal, this model should be able to produce specified amount of amplification for the respective frequencies.

The STFT algorithm for auditory compensation wave was implemented on Simulink. Figure 3 shows the block diagram of the implemented system on Simulink.

An audiofile (in any of the formats, i.e.,.mp3/.wav/.flac/.wma) is used in this work as audio signal. The input signal is sampled at 16 kHz. The output of “From Multimedia File” block is discrete, hence, to perform STFT, along with the discrete input a discrete window function is required for which a discrete pulse generator is used. Parameters defined in this block define STFT properties such as the window length, DFT length (FFT is used instead owing to its reduced computational complexity) and hop length. Sample time of both, the pulse generator block and “from multimedia file” block should match since both these values are multiplied in STFT block before the FFT is performed. Figure 4 represents the internal block diagram of STFT.

Functional units are only multiplier and FFT unit. Buffers are added to give input element-by-element (sample-wise), and the other buffer is added to match the FFT length. Blocks “input,” “window” and “fft_in” records data passing through and sends to MATLAB workspace. Buffer connected to output of multiplier buffers (collects) 16 products and then sends it to FFT unit.

The STFT parameter “hop length” is defined by specifying number of zeroes within a time period T of the window. It is given by:

$$\text{hoplength} = (1 - \text{Duty cycle}).T \tag{17}$$

The samples for which corresponding window value is ‘1’ is buffered and sent to the FFT unit. The FFT length and the window length must be same, and in this project, they are both set to 16. The 16 complex outputs obtained from FFT is fed to a custom Simulink block which determines the magnitude and then the index of

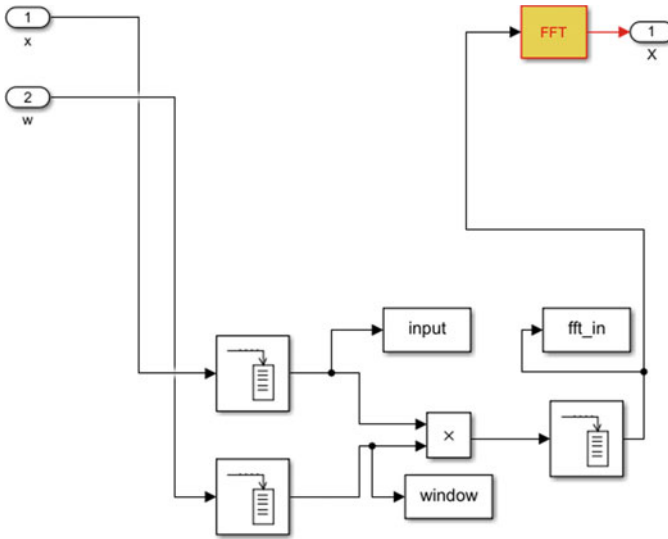


Fig. 4 STFT internal block diagram

the component which has highest magnitude. This index corresponds to bin number of the dominant frequency component. This is then fed to another custom Simulink block which outputs the gain value based on the input fed (index). The gain values are determined beforehand for the subject by audiometry test and stored according to the bin numbers. The discretized digital output from “From multimedia file” block is multiplied with the obtained gain value for amplification, and this product is sent to “To multimedia file” block which writes another audio file. This represents the corrected audio signal as per audiogram. The generated audio file can be played either on MATLAB or on any other media player to observe amplification.

The STFT parameters, window length and hop length, are determined by trial. Various audio files sampled at 16 kHz with 16-bit sample precision were given as input to the system, and after amplification, it was written to another audio file and the generated audio file was tested for quality by playing it on a media player. Aging does not affect the low-frequency thresholds so 16-point FFT is used in this work. Hence, one 1 kHz band is placed in the first bin and the frequency resolution would be 1 kHz. Although 32-point FFT could provide better results since it has frequency resolution of 500 Hz, the resource utilization would be significantly higher [15]. Since FFT length is set to 16, window length must also be 16. Hop length determines the number of samples that are not going to get processed; the more it is, lesser computation would be done, but it might also degrade the audio quality. Maximum hop length must be determined such that the audio quality is retained as that of source. This is done by incrementing the hop length by ‘1’, and then, the simulation is run to check the audio quality. The hop length is incremented till the audio quality is retained, with this methodology, a maximum hop length of 16 was achieved. In the

Discrete Pulse Generator block, difference between period and pulse width serves as the hop length. The parameter phase delay could also be used instead to set the hop length. Pulse width specifies the window length. This implies, alternate frames consisting of 16 samples are being processed, which in turn implies that only 50% of the total samples are processed, thereby reducing the switching in circuit by 50% and reducing dynamic power dissipation.

3 HDL Implementation

The whole system as indicated in Fig. 5 is implemented in RTL using Verilog HDL. Verilog code for 16-point FFT is generated using MATLAB. Verilog code for all other modules were designed manually. Other modules required are clock dividers, SIPO unit, Max bin block, gain look up table, gain multiplier. The system's main clock which runs at 256 kHz is fed to the FFT block and a clock divider circuit. The FFT module is computationally intensive and requires seven clock cycles to provide the output which is why it is driven with a faster clock. In the system, two clock dividers, clock divider 16 (clk_div_16) and clock divider 32 (clk_div_32) are required.

The multiplexer must select input from the ADC for a duration of one frame. Since each frame consists 16 samples and the sampling rate is 16 kHz, one frame happens to be 1 ms long. As alternate frames are being processed, multiplexer must output '0' for 1 ms as well, for which select line must be '0' for 1 ms. The system clock runs at 50% duty cycle, with the help of clock dividers the required window can be generated. Hence, a window of 1 ms and a hop length of 1 ms is required. The clock period for such a window must be 2 ms with 50% duty cycle. clk_div_32 (8 kHz clock) is used as select signal input to multiplexer.

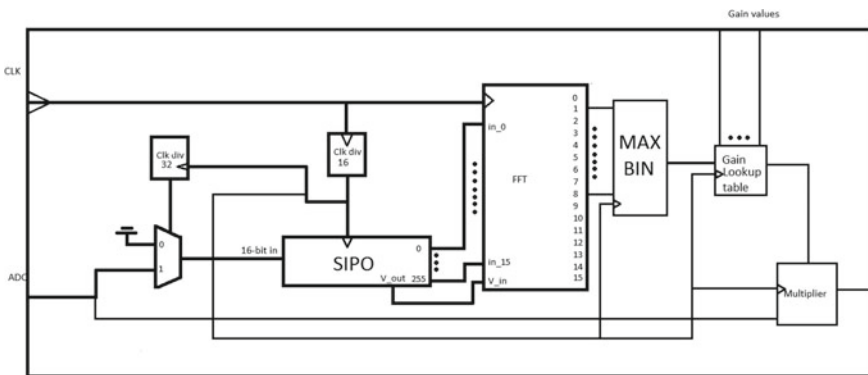


Fig. 5 RTL block diagram of the system

Serial-In Parallel Out (SIPO) unit is driven by `clk_div_16`(16 kHz clock). Serial in Parallel Out unit collects 16 input samples and then outputs them parallelly. It has an inbuilt counter which counts the number of inputs received. Once the number reaches 16, it outputs all the received inputs at once and sets the valid bit high so as to indicate that the output is ready.

The input from ADC, each of 16-bit, after passing through the multiplexer, gets accumulated in SIPO. Sixteen such samples from ADC get accumulated in SIPO after which they are fed to FFT block for processing. Since the second $N/2$ values of FFT are repetition of first $N/2$ values, only first $N/2$ outputs from this block are carried to the MAX bin block. As 16 point FFT is performed, the first 8 outputs of FFT block are real components and the next 8 are corresponding imaginary components. The MAX bin block uses first eight outputs of FFT, determines the maximum among them and finds its index (bin number) and outputs it. The module must be initially reset to initialize the states of registers. The MAX BIN block calculates the dominant frequency component and outputs its index, which is then fed to the gain lookup table wherein gain values are pre-stored; hence, the gain value according to the index stored in lookup table is given as output. The amplification required at frequencies corresponding to the bin values is determined by the audiogram and stored in the gain look up table. The gain multiplier is a signed 16-bit multiplier. The sampled input in real time is multiplied by the gain value in gain multiplier block to produce the required amplification. It is assumed that the audio signal is quasi-stationery.

4 Implementaion on XILIXN VIVADO

Implementation of the proposed system was done with respect to Xilinx Artix-7 (xc7a100tcsq324-1) using Xilinx Vivado HLx Design suite 2019.2. Figure 6 depicts the elaborated design schematic for the system. It is very similar to the RTL block diagram shown in Fig. 5.

Table 1 depicts timing summary of implemented design. It can be seen that setup, hold and pulse-width slacks are positive and that the design has met timing

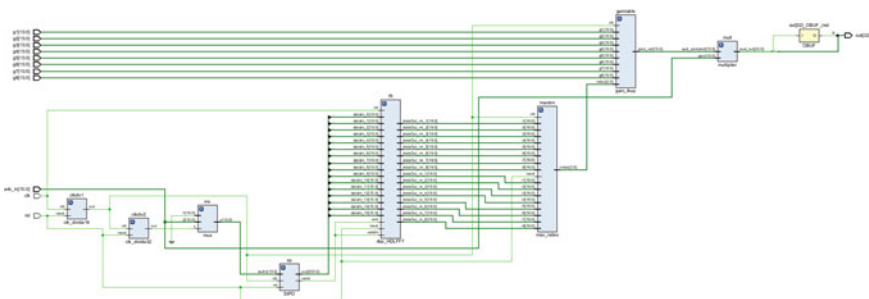


Fig. 6 Elaborated schematic on Vivado

Table 1 Implemented design timing summary

Design timing summary		
Setup	Hold	Pulse width
Worst Negative Slack (WNS): 3730.144 ns	Worst Hold Slack (WHS): 0.051 ns	Worst pulse width slack (WPWS): 1952.145 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total pulse width negative slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of failing endpoints: 0
Total Number of Endpoints: 15,697	Total Number of Endpoints: 15,697	Total number of endpoints: 5156
<i>All user specified timing constraints are met</i>		

Table 2 Resource utilization, implementation

Hierarchy									
Name	Slice LUTs (63400)	Slice Registers (126800)	F7 Muxes (31700)	Slice (15850)	LUT as Logic (63400)	LUT as Memory (19000)	DSPs (240)	Bonded IOB (210)	BUFGCTRL (32)
top	3231	4771	16	1359	3040	191	46	179	2
clkdiv1 (clk_divide16)	4	4	0	4	4	0	0	0	0
clkdiv2 (clk_divide32)	16	5	0	7	16	0	0	0	0
fft (dsp_HDLFFT)	2385	4350	0	1142	2194	191	45	0	0
gainable (gain_ikup)	32	128	16	51	32	0	0	0	0
maxbin (max_index)	502	23	0	150	502	0	0	0	0
mult (multiplier)	0	0	0	0	0	0	1	0	0
sp (SIPO)	295	261	0	156	295	0	0	0	0

constraints. It can be inferred that after logic, power and area optimisations, the elaborated schematic which was merely a graphical representation of the RTL has transformed into device specific schematic after synthesis and PAR. Table 2 represents resource utilization for the implemented design.

It can be seen that FFT block contains significantly more logic than any other blocks. Large number of IOBs on top module is attributed for the gain values being fed using verification IP cores VIO, ICON and ILA. While creating this IP core, the nets to observed and probed must be mentioned after which the design is re-implemented. For this re-implemented design, a bitstream is generated and then the FPGA is programmed after which input signals (or nets) are triggered from Vivado and sent to the FPGA board, the board processes and sends back the output to Vivado. In this manner, all desired inputs, outputs and intermediary nets can be probed. To program the device (FPGA), the implemented design is converted to a bitstream using Xilinx bitstream generation tool. The FPGA is programmed and verified for the functionality.

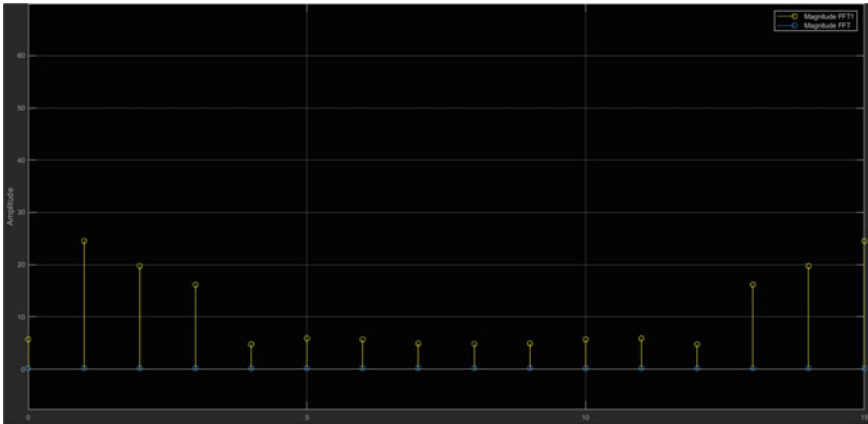


Fig. 7 Comparison of outputs from magnitude FFT blocks of input and output

5 Results and Analysis

5.1 Verification of the Algorithm on Simulink

The model built on Simulink was tested for numerous audio files sampled at 16 K Hz with 16-bit precision. The amplification was observed by keeping the volume levels of the media player at same level for input and output audio files. The amplification was also observed by comparing the outputs of magnitude FFT blocks for input and output frames (16 samples). Figure 7 depicts the comparison of outputs between the magnitude FFT blocks of the input and output. Magnitude FFT represented in blue is for the input signal whereas magnitude FFT1 represented in yellow is of the output. It can be seen from the waveform that the magnitude of output is significantly higher than that of the input.

To see frequency-based amplification, control over each frequency bin also had to be tested, for which monotone signals were fed and gain values were configured only to amplify that particular bin and their magnitude FFT graphs were compared. When 16-point FFT is applied on a signal sampled at 16 kHz, each bin has resolution of 1 kHz. Figure 8 represents magnitude FFT comparison for a monotone signal of 3 kHz. Only the third address of the LUT was filled, rest all were initialized to zero. This way, it was ensured that only the 3rd bin was amplified.

5.2 Modeling with Delay on Simulink

The delay that the implemented system imposed is obtained from post-implementation simulation. The same (delay of about 1.15 ms) has been modeled in

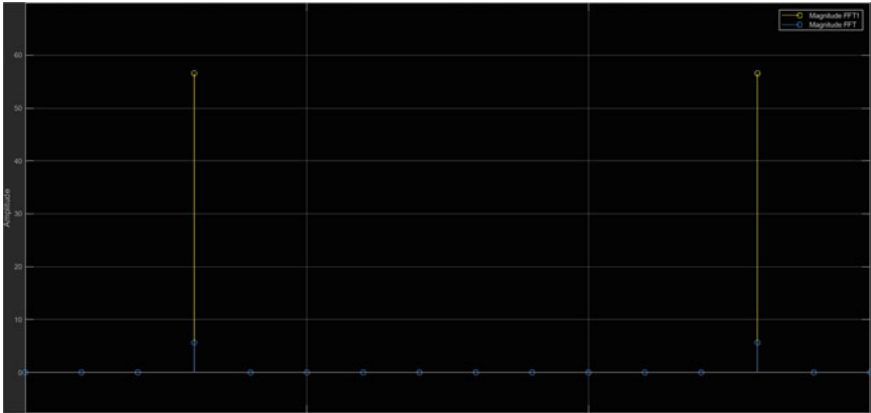


Fig. 8 Amplification of monotone signal

Simulink and another audio file is written with these constraints. Figure 9 represents the system being modeled on Simulink with the delays. The output of the second custom block is made to go through 16-unit delay elements. Each of these unit delay blocks provide 1 sample time delay, and 16 such blocks are used to model for the parasitic effects (in terms of delay) introduced by the FPGA.

Figure 10 represents audiogram of a patient. The custom block 2 is populated with gain values in accordance with the audiogram and is then simulated and the Fig. 11 represents the FFT magnitude for the same. Since gain values are required vary logarithmically, they are significantly higher than the input samples.

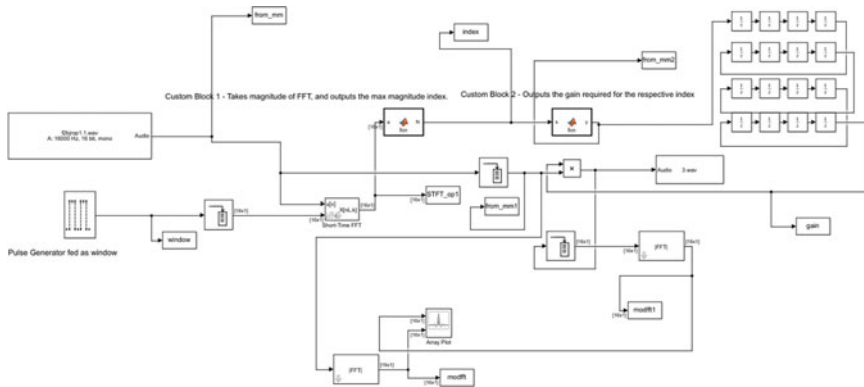


Fig. 9 Modeling on simulink with delay

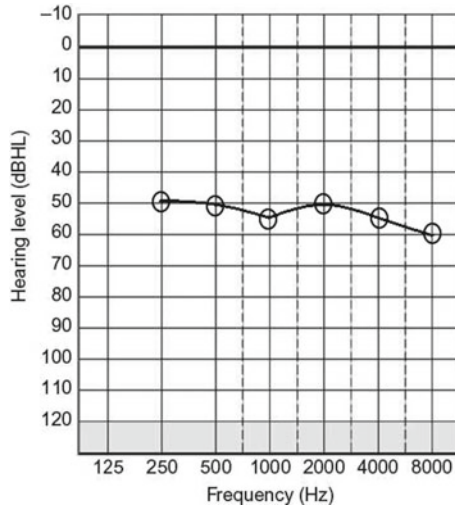


Fig. 10 Audiogram

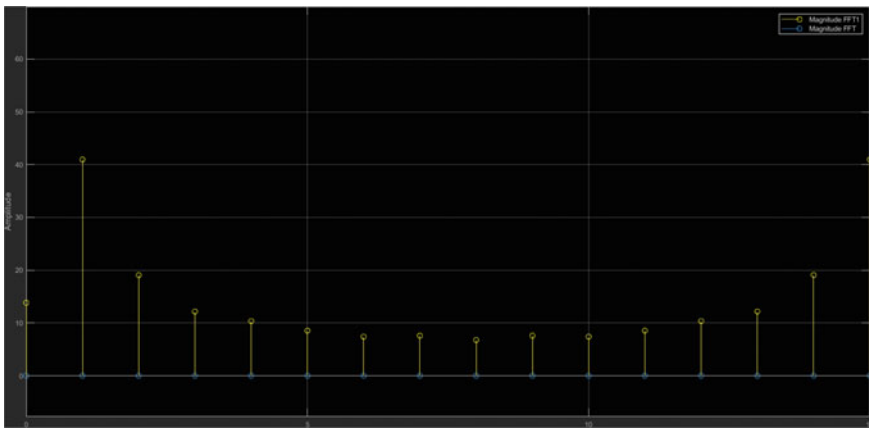


Fig. 11 Magnitude FFT when configured with respect to the above audiogram

6 Conclusion

Hearing loss is diagnosed when hearing testing finds that a person is unable to hear 25 decibels in at least one ear. Treatment for hearing loss depends on the cause and severity of hearing loss. The hearing aid is the most commonly used solution. The STFT approach for auditory compensation was verified on Simulink and implemented on Digilent Nexys-4 DDR Artix-7 FPGA using Xilinx VIVADO. Though filter bank approach provides high quality and high personalization (in terms of amplification), its resource utilization is significantly higher when compared to that

of STFTs. By merely determining the dominant frequency component for that instant and appropriately applying gain replaces the reconstruction process followed in filter bank approach, by doing so, not just the synthesis filter is forsworn the additional noise filter added with the synthesis filter can be dropped as well. Frequency resolution can be changed by varying the FFT length and window length instead of adding filters. STFT introduces a new paradigm for the design of digital hearing aids and can replace the filter bank approach.

References

1. Alexandro MSA, Eduardo LR, Bampi S (1999) Dynamically reconfigurable architecture for image processor applications. In: DAC '99: proceedings of the 36th annual ACM/IEEE design Automation Conference, USA, pp 623–628. <https://doi.org/10.1109/DAC.1999.782018>
2. Gonzalez RC, Woods RE (2014) Digital image processing. Pearson Education Limited
3. Fowers SG, Lee D-J, Ventura DA, Archibald JK (2012) The nature-inspired BASIS feature descriptor for UAV imagery and its hardware implementation. *IEEE Trans Circuits Syst Video Technol* 23(5):756–768. <https://doi.org/10.1109/TCSVT.2012.2223631>
4. Fularz M, Kraft M, Schmidt A, Kasiński A (2015) A high-performance FPGA-based image feature detector and matcher based on the FAST and BRIEF algorithms. *Int J Adv Rob Syst* 12(10):1–15. <https://doi.org/10.5772/61434>
5. Kuo YT, Lin TJ, Li YT, Liu CW (2016) Design and implementation of low-power ANSI S1.11 filter bank for digital hearing aids. *IEEE Trans Circuits Syst: Regular Papers* 57(7):1684–1696. <https://doi.org/10.1109/TCSI.2009.2033539>
6. Levyitt H (1987) Digital hearing aids: a tutorial review. *J Rehabil Res Dev* 24(4):7–20
7. Liu CW, Chang KC, Chuang MH, Lin CH (2013) 10 ms 18 band quasi-ANSI S1.11, 1/3 octave filter bank for digital hearing aids. *IEEE Trans Circuits Syst* 60(3):638–649. <https://doi.org/10.1109/VLSI-DAT.2012.6212620>
8. Chong KS, Gwee BH, Chang JS (2006) A 16-channel low-power non-uniform spaced filter bank core for digital hearing aids. *IEEE Trans Circuits Syst* 5(9):853–857. <https://doi.org/10.1109/TCSII.2006.881821>
9. Wei Y, Liu D (2013) A reconfigurable digital filterbank for hearing-aid systems with a variety of sound wave decomposition plans. *IEEE Trans Biomed Eng* 60(6):1628–1635. <https://doi.org/10.1109/TBME.2013.2240681>
10. Chong KS, Gwee BH, Chang JS (2006) A 16-Channel low-power nonuniform spaced filter bank core for digital hearing aids. In: 2006 IEEE Biomedical Circuits and Systems Conference, pp 186–189. <https://doi.org/10.1109/TCSII.2006.881821>
11. Girisha GK, Pinjare SL (2020) Implementation of novel algorithm for auditory compensation in hearing aids using STFT algorithm. *Acta Technica Corviniensis—Bull Eng* 13:4654–4660
12. Griffin D, Lim J (1984) Signal estimation from modified short-time Fourier transform. *IEEE Trans Acoust Speech Signal Process* 32(2):236–243. <https://doi.org/10.1109/TASSP.1984.1164317>
13. Alsteris LD, Paliwal KK (2007) Iterative reconstruction of speech from short-time Fourier transform phase and magnitude spectra. *Comput Speech Lang* 21(1):174–186. <https://doi.org/10.1016/j.csl.2006.03.001>
14. Nakajima Y, Matsuda M, Ueda K, Remijn GB (2018) Temporal resolution needed for auditory communication: measurement with mosaic speech. *Frontiers Human Neurosci* 12:149. <https://doi.org/10.3389/fnhum.2018.00149>
15. Kurakata K, Mizunami T, Sato H, Inukai Y (2008) Effect of Ageing on Hearing Thresholds in the Low Frequency Region. *J Low Freq Noise Vib Active Control* 27(3):175–184

Smart Headgear for Unsafe Operational Environment



P. Dhanush, S. Jagdeesh Patil, R. U. Girish, G. Chethan, and S. K. Chethan

1 Introduction

Two-wheeler accidents [2] are increasing day by day leading to death of numerous lives. The probability of these deaths can be decreased significantly by using Smart Headgear (Helmets). About 1.2 million individuals are losing their lives in street accidents. The demise rate is not diminishing despite the fact that the clinic is giving crisis emergency vehicle services. So, to conquer these issues, we have two rules to be met by wearer of savvy head protector. One is that the rider must wear the helmet which is checked by FSR sensor, and second is the rider should not have consumed alcohol before riding which is checked by the alcohol MQ3 sensor.

At the point when the rider has devoured liquor, the MQ3 sensor will detect the rider's inhale to identify the measure of liquor content and make an impression on enlisted contact. Third, when the rider meets with a mishap, the Gyro sensor will identify it and module will send the client's area to crisis administrations and his enrolled contacts through a SMS. The cap can distinguish a mishap, utilizing the Gyro sensor. Liquor MQ3 sensor detects the breath of the rider to recognize if the current level is within authorized limit or not. FSR sensor identifies whether the rider is wearing the Helmet or not.

The purpose of Smart Headgear/Helmet [3] is to provide safety for the vehicle rider. This Headgear has advanced features like alcohol detection [1], accident identification, location tracking, use as a hands-free device, and fall detection. This makes it not only a Smart Headgear but also a feature of a smart bike; ignition switch of the vehicle cannot turn ON, without wearing the Helmet, which makes it compulsory to wear the helmet.

P. Dhanush · S. J. Patil (✉) · R. U. Girish · G. Chethan · S. K. Chethan
Department of ISE, Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: jagadish.patil@nmit.ac.in

An RF Module is used as wireless link for communication between transmitter and receiver. If the rider is drunk, the ignition gets automatically locked and sends a message to the registered number (guardians or Government Authority) with his current location. In case of an accident, it will send a message through GSM along with location with the help of GPS Module. The most important use of this paper is to detect fall detection; if the rider falls down from the bike, the Helmet sends a message.

Developing and deploying robust user-friendly Headgear (Helmet) which rider or miner wear like normal Helmet and at the same time serve to.

- Monitor the health of the driver (drowsiness, sleep, fall detection by accident, or self-fall).
- Locate the GPS location of his movement.
- Inform the authorities and siblings for medical help and security.

2 System Architecture

The workflow of Smart Headgear is shown in Fig. 1. Smart Headgear is using commercial grade embedded hardware with sensor like Accelerometer, Gyroscope, GPS, FSR, MQ3, etc.

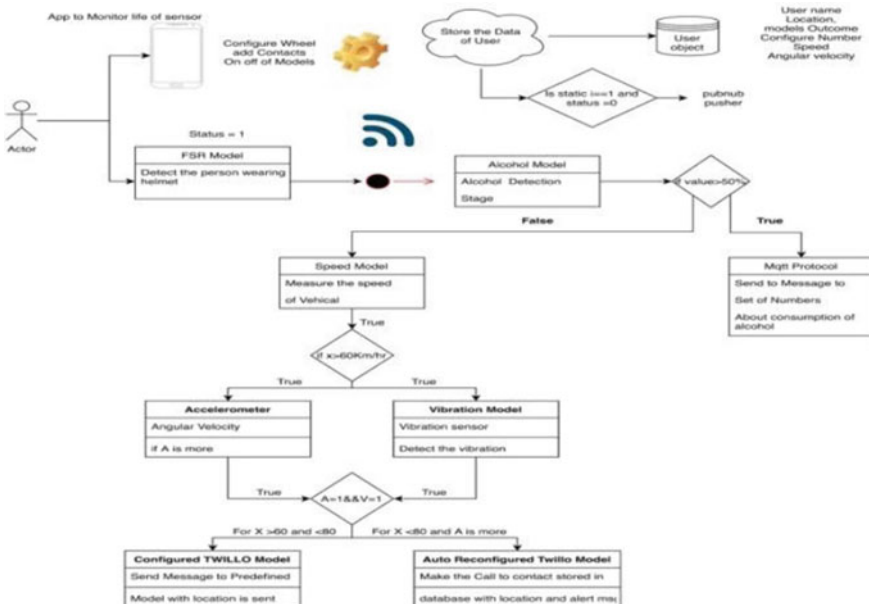


Fig. 1 Workflow of smart headgear

- The sensors collect the monitoring parameters like level of intoxication, sleep drowsiness of driver, speed of vehicle, fall detection, location of Headgear, user details, etc.
- The sensors collect the date and time and then packetized and communicate these data on regular intervals to storage location (cloud database).
- Mobile app is developed to register user for the service and avail the services.
- Users can also use web app for register process and for sake of service-related queries.
- The AWS EC2 is used for analysis of the health of sensor by using outlier model.

In designing Smart Headgear, the following are taken into account:

The project consists of various sensors like alcohol sensor (MQ3), Gyro sensor, vibration sensor, Accelerometer, NEO-6 M GPS Module, and FSR sensor. The main controller that we have used is NodeMCU which is small in size and very less expensive and produces high performance with accuracy. The software that we have used for building an intelligent smart application using the flutter called “SHMET” for monitoring the life of the sensors and for many more features. Django is another powerful framework that we have used for building web applications. Rest API is built as a backend and deployed on the Heroku Platform. The database that we have used for this project is MongoDB cluster for storing all data. The basic components of Smart Headgear consist of the following components.

2.1 NodeMCU (Node Microcontroller Unit)

A NodeMCU is one of the microcontrollers that we have used in this project to make a user plugin model which is an open-source board for specifically developing IOT-based applications. NodeMCU is small in size which is very effective when it comes to building micro-projects in terms of both efficiencies and performance. The NodeMCU has hardware that runs on the ESP8266 Wi-Fi. NodeMCU supports Arduino IDE which can be easily programmed using Arduino IDE. NodeMCU has a high processing power which co-operated at 80–160 MHz a customizable clock recurrence. NodeMCU likewise has inherent Wi-Fi and deep rest working highlights which make it ideal for IOT projects. In this paper, NodeMCU connects all sensors to collect all sensor values for checking whether the user has consumed alcohol, the vibration sensor for the fall detection, and sends all data to the cloud database.

2.2 Alcohol Sensor (MQ3)

This module is used for checking whether the user consumed alcohol or not. We have tested this alcohol sensor with different alcohol containers like a sanitizer, water, etc. After all, if the user consumed more alcohol, we are going to send SMS to parents

or guardians that the user has consumed more alcohol. This alcohol sensor has high sensitivity and fast response time. This makes sure that the user is in a safer position. Features of alcohol sensor are as follows:

- Wider detecting scope that helps for user.
- Highly stable and also have long life.
- Produces fast response and also has high sensitivity.

2.3 Module Gyro Sensor

Gyro sensors are precise rate sensors. The precise speed can be characterized as the adjustment of rotational point per unit of time. Gyro sensors can detect the rotational movement of the Helmet and also the progressions in direction. Lately, Gyro sensors have tracked down that these sensors can be utilized in camera shake location frameworks, movement detecting, and vehicle solidness control frameworks which are on the other hand called against slip, and so on Gyro sensors have filled quickly in the spaces like vehicle driver security and emotionally supportive networks and furthermore in a robot movement control. The Application Module provides recommendation service to end user and user feedback interface. It is an interface to provide recommendations generated in the business module to end user in regular intervals via Web Module API and when user searching the catalog information provided by the service provider. The user feedback is collected here after watching the content and provided to business module for generating the recommendations for that user. They are used for measuring angular velocity sensing and angle.

2.4 Vibration Sensor

Vibration sensors use the effects called piezoelectric mechanism which measures the changes within pressure, temperature, and force by changing to an electrical charge. It has an ability to sense weak vibration signals. Whenever an accident occurs and if the user is using this smart helmet, firstly it senses the pressure and immediately sends the data to NodeMCU, and then, NodeMCU sends SMS to his family members along with his location details without any delay.

2.5 FSR Sensor (Force Sensing Resistor)

This sensor is mainly used in our project for the detection that whether the user is wearing the helmet or not. This sensor will sense human touch for the fetching data. This sensor will be activated before the user starts his bike. If the user wears helmet and satisfies the condition, then it will send the signal to the user that he can start his

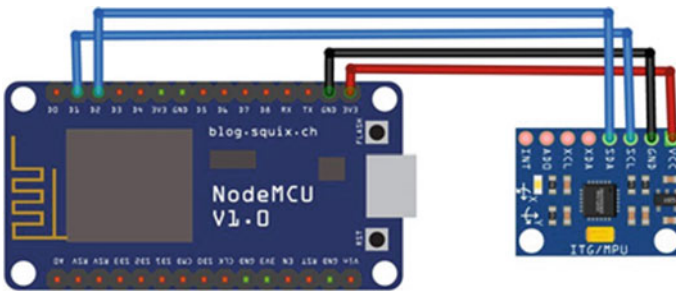
bike. FSR sensors are used extensively in medical systems and in robotic field and industry applications.

3 Software Implementation

As the project has been implemented using RPI3 and sensor like FSR MQ3, Gyro, and GPS Modules, the steps followed are as below:

3.1 Setup of NodeMCU

NodeMCU contains “hub” and “MCU” (miniature regulator unit).



3.2 Creating Cloud Database

As the values for NodeMCU and sensor must be stored for future analysis and reference, we need a cloud-based database. As the parameters are dynamic, NoSQL Database has been used (Mongo Atlas), which is made available for the team members to access the database.

- The Tables are Users, Alcohol values, and Sensor Status.
- The tables are mapped each other.

3.3 Setting up of Rest API (Backend)

The Rest API is used to communicate sensors by app and web app. The Rest API was created by using Flask, the API was tested using Postman, the API is deployed

on Heroku, and the URL takes a parameter with predefined arguments and renders the json. The data from this URL is inserted or updated into Mongo dB.

URLs:

<https://shmet.herokuapp.com/>

<https://shmet.herokuapp.com/api/user/12345/status/off>

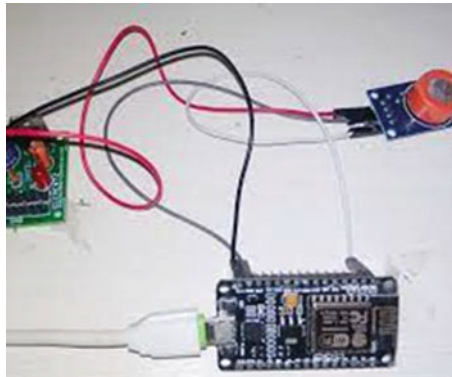
<https://shmet.herokuapp.com/api/user/12345/getalcohol>

<https://shmet.herokuapp.com/api/valid/dhanushnayak.ram@gmail.com/Dhanushp>

3.4 *Circuit Connection of MQ3*

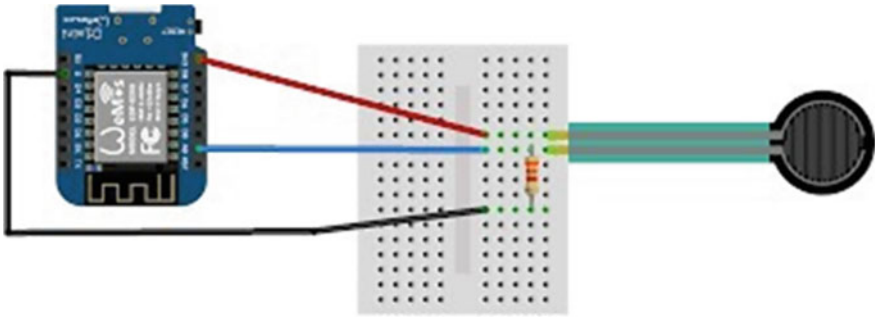
This sensor is used to test the alcohol content from breath. As RPI does not have analog converter, Analog to Digital IC is used which is called as MCP3008. Pin Diagram of MQ3 Module:

VCC supplies power for the module. You can connect it to 5 V output, GND is the Ground Pin and needs to be connected to GND pin, D0 provides a digital representation of the presence of alcohol, and A0 provides analog output voltage in proportional to the concentration of alcohol. The A0 is connected to channel 0 of MCP3008. The circuit connection is as shown in below figure.



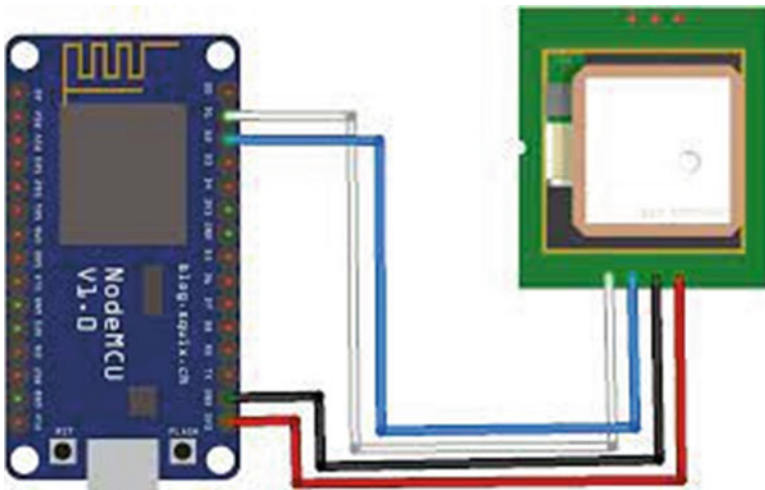
3.5 *Circuit Connection of FSR*

The FSR is used to check if rider is wearing Helmet. FSR allows us to detect physical pressure, squeezing, and weight. A force sensitive resistor (FSR) is a material which changes its resistance when a power or pressing factor is applied. At the end of the day, a force sensitive resistor is a sensor that permits you to identify actual pressing factor, crushing, and weight. The circuit connection is as shown in figure.



3.6 Circuit Connection of GPS

GPS is used to track or to fetch the location of Helmet as well as Bike Riders, which helps whenever person consumes alcohol or met with accident to forward the location to the guardians. The connections are shown below.



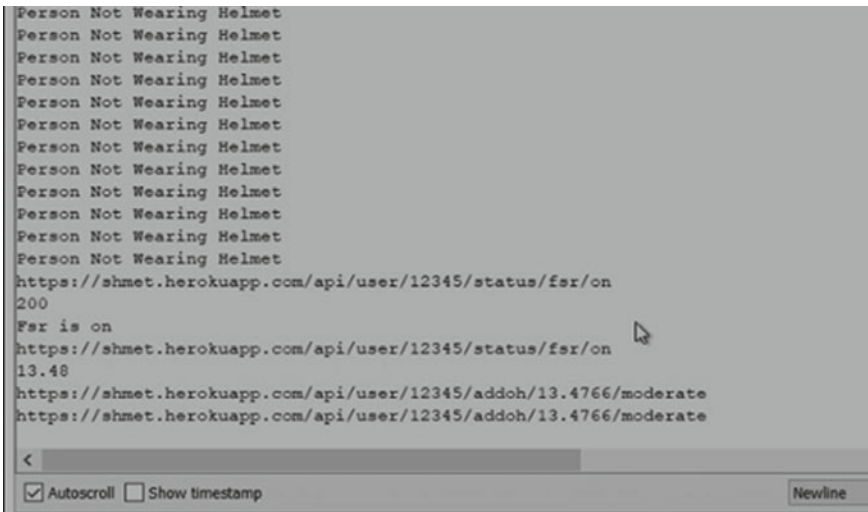
4 Testing and Field Trails

4.1 Testing of Alcohol Sensor

Presently to test the working of sensor, we used alcohol-based solutions on the sensor, the fumes of the liquor into the sensor and change the pot move clockwise with Status LED is ON the pot move back counterclockwise until the LED goes OFF when the fumes of alcohol reduce. The sensor can be adjusted to authorized level and prepared for use.

4.2 Testing of FSR Sensor

We were able to check the working by just applying some pressure on the sensors using our hands and put some weight using objects. Then, we were able to notice the change in value of sensor in the terminal while it was running on the IOT model.



```
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
Person Not Wearing Helmet
https://shmet.herokuapp.com/api/user/12345/status/fsr/on
200
Fsr is on
https://shmet.herokuapp.com/api/user/12345/status/fsr/on
13.48
https://shmet.herokuapp.com/api/user/12345/addoh/13.4766/moderate
https://shmet.herokuapp.com/api/user/12345/addoh/13.4766/moderate
<
 Autoscroll  Show timestamp Newline
```

4.3 Testing of Gyro Sensor

We were able to see the working of Gyro sensor after connecting to a bread board and power supply and just changing the orientation of the sensor and rotating the

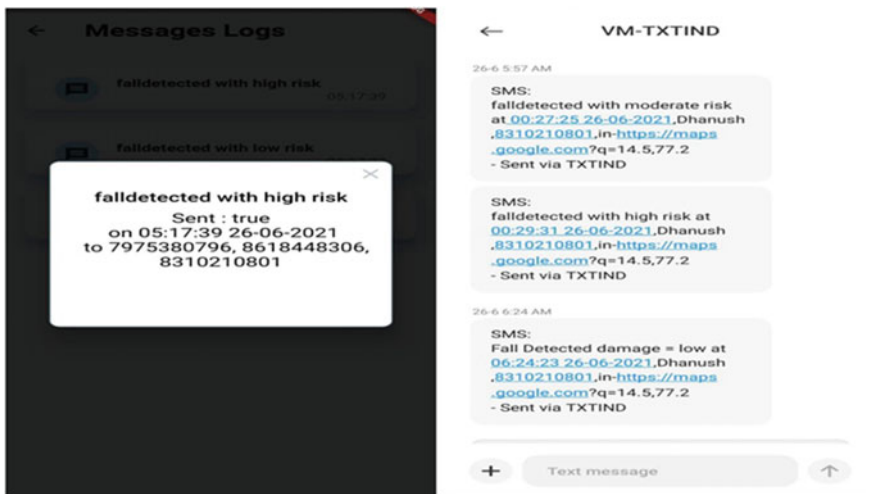
position of sensor. We could see the change in value at the terminal we changed the from not fall to fall as indicating the axial position of sensor has been disrupted.

```
10
Not Fall
10
Not Fall
10
Not Fall
10
Not Fall
9
Not Fall
1
TRIGGER 1 ACTIVATED
Not Fall
19
TRIGGER 2 ACTIVATED
49
49
TRIGGER 3 ACTIVATED
```

Autoscroll Show timestamp Newline

4.4 Testing of GPS

After implementing GPS sensor within the model and configuration of GPS using an API, user was able to locate the current position within the app itself and we were also able to send the co-ordinates to the SOS emergency alert as well.



5 Conclusion and Future Scope

- The smart protective headgear guarantees the security of the vehicle rider to wear head protector and furthermore guarantees that the rider has not devoured liquor more than prescribed limit.
- In case by chance, if any of security rules are abused, the proposed framework will forestall the vehicle rider from beginning the riding.
- The smart protective headgear additionally helps in treatment of the outcome of mishaps by sending a SMS to the area of the biker to the police headquarters and guardians.
- It helps to monitor the casualties to be noticed, on the off chance that he/she met with a mishap.
- Smart Headgear is worked with a ton of exceptional frameworks and highlights, so they can in any case wear Smart Headgear/Helmet like they wear the standard ones. Thus, this paper helps to make a smart helmet audit with pleasant highlights yet at the same time awesome in each perspective.

References

1. Vijayan et al. (2014) Alcohol detection using smart helmet system
2. Gorges et al. (2019) Impact detection using a machine learning approach and experimental road roughness classification. *Mech Syst Signal Process* 117:738–756
3. Tapadar et al. (2018) Accident and alcohol detection in Bluetooth enabled smart helmets for motorbikes. In: 2018 IEEE 8th annual computing and communication workshop and conference (CCWC), pp 584–590

IMPROVE the Solar Panel Proficiency by Using of Free Energy from Street Light



A. Saravanan and N. Sivaramakrishnan

1 Introduction

Energy utilization is most essential thing followed by all developed countries. Day by day, the electricity energy demand has been increasing in many countries. Revolution of electrical cars and e-vehicles in automobile field of reducing the global warming, less utilization of fuels and cost-effective makes that advancement of renewable energy system and development of new techniques for utilization of renewable energy and free energy sources available from solar, wind and tidal energies, etc. India is the country which gets the maximum amount of sunlight throughout the year. As the energy produced by the non-renewable resources in our country is in the verge of extinction, the future world needs the energy. This leads to the way for trapping the energy through the renewable energy. And other renewable energy other sun is the periodic one. So we cannot rely on them to get the stable power supply throughout the year so the solar energy can be used to get the stable power supply without the fuel cost and maintenance cost.

Though it has high initial cost, it produces reliable power without polluting the environment. In recent trends, researches have been widely done to minimize its initial cost and for effective power production. The sunlight can be trapped by the photo voltaic cell. The array of such photo voltaic cell placed in the panel is called photo voltaic panel or the solar panel. The maximum amount of power can be trapped when maximum amount of sunlight falls on the solar panel which can be done by providing the solar panel with solar tracker and the mirror [1–3]. Evening time solar power is disappeared and cannot utilize the power but the alternative wave to find other sources of light energy for utilization, the street light energy as a one of the source. Here, the mechanical design has been changed for mounting of street light

A. Saravanan · N. Sivaramakrishnan (✉)
Department of EEE, SMK Fomra Institute of Technology, Kelambakkam Chennai, India
e-mail: sivaramakrishnaneie@gmail.com

and utilized energy source for power production; the produced power of both solar and street energy has to be conditioned using DC–DC boost converter to increase the voltage output [4]. MPPT algorithm is used to increase the efficiency of the solar panel.

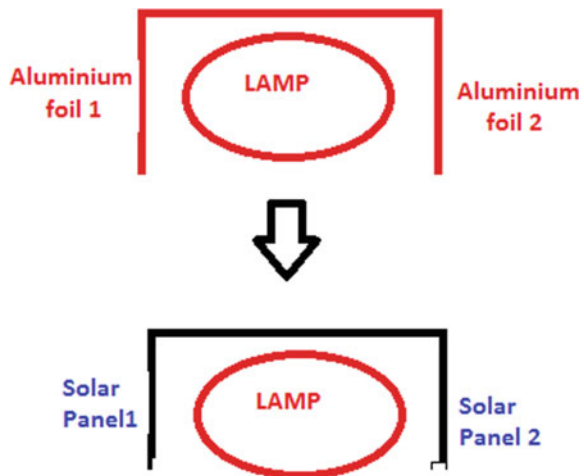
2 Mechanical Design and Arrangement of Solar Panel

The arrangement of solar panel has been made for utilization of solar and street light power during day and night time, respectively. The placement of solar panel is shown in Fig. 1. This arrangement contains the various segments like mirror, solar power tracker, cooling equipment and GSM-based monitoring to monitor the performance and improve the efficiency [4]. The street light mechanical design and arrangement have been changed. The four aluminium layers are used to focus the light during night time but the two aluminium covers are enough to focus light without affecting the light luminance. The mechanical design has been changed to replace solar panel for two aluminium covers. Here, we place the solar panel with solar power tracker over the street light by removing the two layer of the aluminium covering as only the four sides of the aluminium covering is used to focus the light.

Solar tracker

Light Detecting Resistor (LDR) is cost-effective device widely used for light detecting and tracking applications. LDR is placed at positions of the top and the bottom layer of the solar panel which has been used as the tracking device of solar panel to track the maximum position of the sunlight. A relay has been used to control the two limit switches. One limit switch is used to turn the solar panel to the original position at the early morning, and other limit switch is used to stop the solar panel at

Fig. 1 Mechanical structure arrangement of street light replaced by two solar panel arrangements



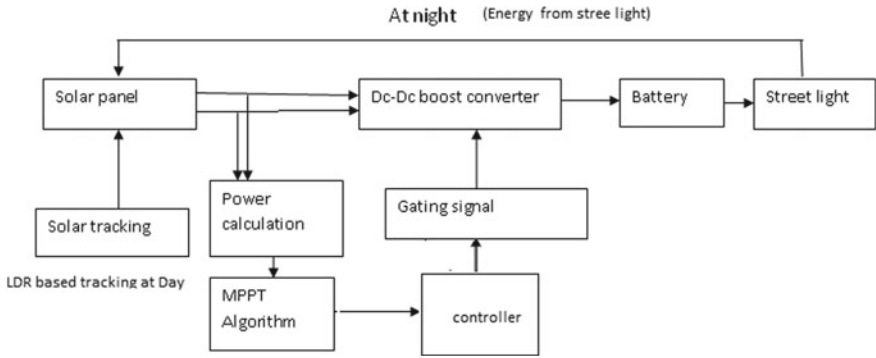


Fig. 2 Overall block diagram of the system

the east direction during the early morning before sunrise. Here, we use the stepper motor for the rotation of the solar panel. And the controlling of the servo motor for the solar tracking is done by using controller. The MPPT algorithm is used to track the maximum output power which increases the efficiency. Here, the MPPT algorithm is implemented in FPGA microcontroller. The DC–DC boost converter is used to increase the output voltage which is then given to the inverter circuit to get the Ac power supply. The basic block diagram for the above method is given in Fig. 2.

3 Proposed System-Overview

Here, we have various elements like solar panel with solar tracker, MPPT algorithm to get high frequency gating signal for IGBT in the boost converter, battery to store energy and the street light as the load. During the day time, the solar panel with the solar tracker is used to track the sunlight and turn the solar panel towards the sunlight so as to get the high intensity of the light from the sun so as to get the high efficiency [5].

At night time, it turns the solar panel towards the street light to trap the maximum light from the street light as the free energy so that 80% of the light that the street light uses is returned back. Here, we use the boost converter with the MPPT algorithm [6, 7]. The P and O method which is the easiest and cheapest method of implementing MPPT is used. It requires only two sensors for measuring the voltage and current, and by perturbing, we get the point at which the voltage and the current are maximum.

And this is used to produce the gating signal for boost converter to get high efficiency. The battery is used to store the energy, and the street light is used for illumination purpose during the night time.

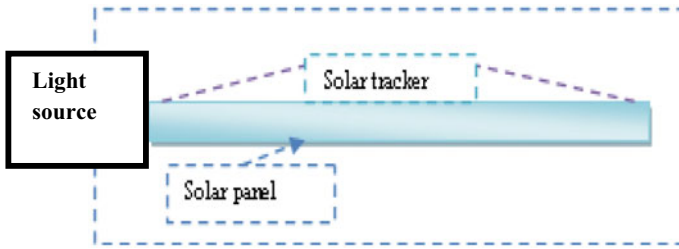


Fig. 3 Solar panel with solar tracker

Solar panel with solar tracker

Solar tracking is the automatic tracking system which is used to track the maximum position of the sunlight and turns the solar panel towards the sun at the day time and turns towards the street light during the night time. This solar tracking can be done by two ways either by using the timing controls which can be used to place the solar panel at a position. The main disadvantage of the timing control is that something during the rainy season even at the morning it darkens.

At that time, it cannot turn the solar panel towards the street light. So we go for the method which uses three LDRs; one is used to sense that the sky darkens or not for automatic street light control and also for turning the solar panel towards the street light. And other two LDRs are used for comparing the intensity of the light and turn the solar panel to the point where we get the maximum sunlight.

This can be done by simple method by controlling the stepper motor by the following method. Here, it compares the light at both LDRs and turns the solar panel towards the position of the LDR which get maximum intensity of light and stop when both the illuminations to the LDR are same (Fig. 3).

Here, the flow chart for the solar panel with solar tracking is shown in Fig. 4.

After receiving the sunlight whole day, the solar panel is turned to westward direction at the end of the day, i.e. the sun set direction. And at the night time, it turns the solar panel towards the street light [8].

Next day, the sun rises from the eastward direction, so that there must be some logic to turn the solar panel towards the eastward direction. This is done by the following logic which is given in Table 1.

The Set/Reset system works in such a way that in the evening when the sun sets, the tracker will be in such a position that it will activate the limit switch which can be used for reverting process.

The result for this Set/Reset system is summarized as follows:

So, from the result it is noted that the tracker works in the perfect manner (Table 2).

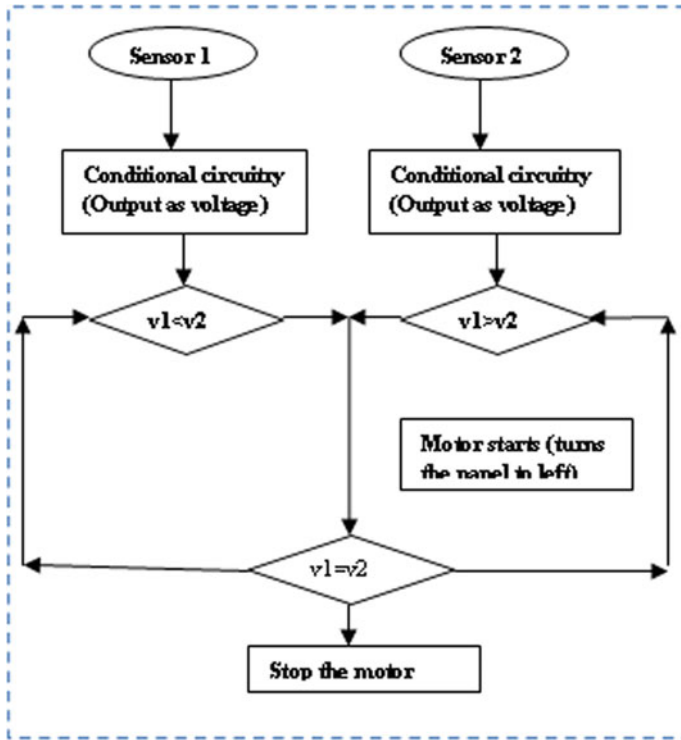


Fig. 4 The flow chart for solar panel with solar tracker

Table 1 Motor control based on LDR

Status of LDRs	Status of motor
LDR 1 is illuminated more than LDR 2	Motor will run in anticlockwise direction
LDR 2 is illuminated more than LDR 1	Motor will run in clockwise direction
Both LDRs illuminated to same light	Motor will be stopped
Both LDRs are exposed to darkness	Motor will be stopped

4 Maximum Power Point Tracking Algorithm (MPPT)

MPPT is the simple method implemented to trap the maximum power from the solar panel. Without MPPT algorithm, most of the energy obtained from the solar panel will be wasted.

The main objective of the MPPT algorithm is to find the panel operating voltage that will provide the maximum power output from the solar panel [9–12]. In MPPT

Table 2 Motor control based on limit switch position

Status of limit switches	Status of motor
If motor is running clockwise	
Limit switch RESET is pressed and released	Motor is continuously running in clockwise direction (in this situation, motor will not follow the output of LDRs)
Limit switch SET is pressed and released	The motor will start following the outputs of LDRs
If motor is running anticlockwise	
Limit switch RESET is pressed and released	Motor changes its direction and runs in clockwise direction (In this situation motor will not follow the outputs of LDRs)
Limit switch SET is pressed and released	The motor will start following the output of LDRs

algorithm, the output power of the circuit will be maximum only when the Thevenin impedance of the circuit matches with the load impedance of the circuit. The MPPT algorithm is implemented by means of FGPA microcontroller.

There are different methods of implementing the MPPT algorithm such as:

- (i) Perturb and observe method
- (ii) Incremental conductance method
- (iii) Fractional short circuit current
- (iv) Fractional open circuit voltage
- (v) Neural networks
- (vi) Fuzzy logic.

The choice of selection of algorithm is done by considering cost and complexity in MPPT. Here, we use perturb and observe method (also termed as hill climbing method) as it is cheap and easy to implement.

Perturb and observe method (P and O method)

It is the cheapest method to implement the MPPT algorithm. Here, we use the single voltage sensor to track the maximum output voltage of the PV panel. [11, 13, 14] So that cost of implementing the Maximum Power Point is less when compared to other method. Here, it does not stop on reaching the Maximum Power Point; it keeps on perturbing on both the direction. Here, modifying the panel operating voltage is done by modifying the converter duty cycle. The algorithm to implement the MPPT algorithm using perturb and observe method is shown Fig. 5.

Here at first, it compares the previous power with the new one when this power is greater than the previous power; then, it increases the operating voltage. If the power is less than the previous power, then it decreases the operating voltage so that the maximum power from the solar panel can be trapped.

From the figure, it can be clearly understood that if the voltage on the right side decreases, the MPPT increases the power, and if the voltage on the left side decreases, MPPT increases the power. This is the main idea of the perturb and observe algorithm.

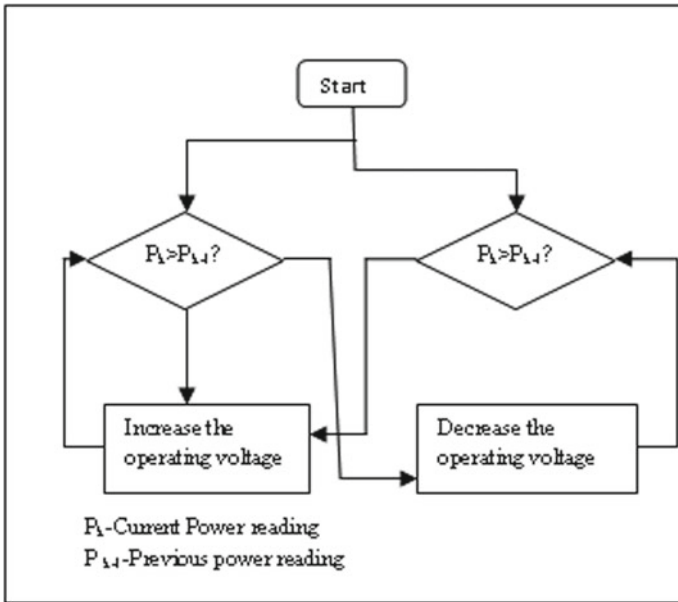


Fig. 5 Flow chart of perturb and observe method

For implementing perturb and observe algorithm, it is necessary to find the voltage and the current of the PV panel.

5 DC-DC Boost Converter

Here, we used the DC-DC boost converter to boost the low voltage DC supply to the high voltage DC supply which is obtained from the output of the solar panel. Here, we use the parallel capacitor in the load side to increase the output voltage of the solar panel. By increasing the output voltage, the efficiency can be increased.

The output voltage of the DC-DC boost converter circuit is $V_0 = V_s / (1 - \alpha)$.

Where V_0 —output voltage of the DC-DC boost converter.

V_s —Source or the input voltage from the solar panel.

α —firing angle of the converter.

Here by varying the firing angle, we can easily get the Maximum Power Point.

Cooling

Cooling the solar panel will improve the efficiency of the solar panel. Here, when the solar panel is heated, the efficiency of the solar panel to produce the output power is reduced. Here, the natural cooling is done by the external wind [15]. If this air is not

Table 3 Battery charging condition status

Condition	Indication message sent to the mobile
When battery is full	Battery full
When battery is half full or medium	Switch off the heavy load or switch off the load that is not necessary
When battery is empty	Battery is empty so use only the light load
When performance is reduced	Check the solar panel for dirt or dust, so clean it
If battery is full, and performance is good	Solar panel works in perfect manner

sufficient to cool the solar panel, then the external setup is done by providing with the cooling fan which is controlled by the microcontroller. This microcontroller will turn on the cooling fan only when it is necessary.

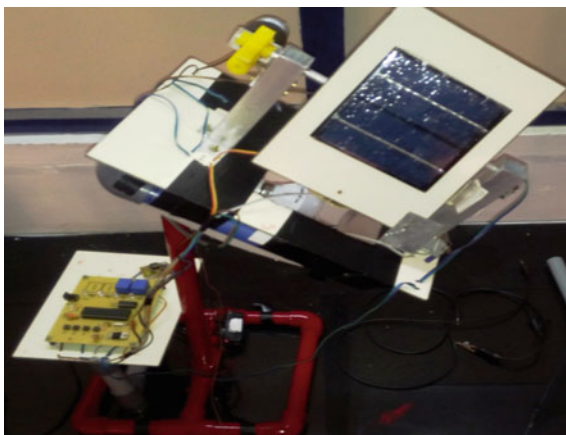
6 Monitoring of the Solar Panel Using Gsm Module

Here, the solar panel can be monitored continuously to improve the performance and to drive the maximum power out of its performance and can be checked continuously by noting the power output at various time either manually or remotely [15–17]. Here, we use the GSM module to check whether the solar panel works perfectly or some external factor reduces its performance such as dust, dirt and fallen leaf from the tree which can be manually cleared when known, which increase the performance. Here, we can use the GSM module to check the status of the battery and switch off the load accordingly. The load can also be switched on and off accordingly. Here, when the battery is full during the day time, the solar can directly supply the load online with battery backup. When the battery is half full or empty, it should instruct the user to reduce the connected load and then simultaneously supplies the load and charges the battery. It also instructs the user to clean the solar panel when the dust or dirt in the solar panel reduces its performance. The table for monitoring the performance and improving it is given in Table 3.

7 Hardware Implementation of Proposed System

Here, when the sun is in the east direction, the solar tracker which is placed above the solar panel tracks the position of the sun and move the solar panel towards it, so the maximum amount of light will be focused on the solar panel which increases the power output. During the night time, it turns the solar panel towards the street light so that the 80% of the power that the street light uses is returned back. Here, the solar panel efficiency reduces when it gets heated so the forced cooling is given to the solar panel when the natural cooling is insufficient. The performance of the solar

Fig. 6 Hardware implementation of our project



panel is monitored by means of remote monitoring of the solar panel using GSM module. Here, we have done this in single street light; if this is implemented in all areas in the city, then we have the surplus amount of power which can be connected to the grid to meet the current power demand problem easily (Fig. 6).

8 Conclusion

The main objective of this paper is to increase the efficiency of the solar panel which can be implemented in practical case. I tried my level best to increase the efficiency of the solar panel with the simplest method. Here, the solar tracking system contains the LDR to detect the position of the solar panel and to turn the solar panel to increase the concentration of the sun light towards the solar panel, and during the night time, it turns towards the street light to increase the efficiency and ultimate efficiency of the solar panel. The MPPT algorithm is used to track the maximum power from the solar panel and the sufficient cooling in order to get the optimum efficiency.

References

1. Piegari L, Rizzo R (2010) Adaptive perturb and observe algorithm for photovoltaic maximum power point tracking. *Renew Power Gener*, IET 4(4):317–328
2. Femia N, Petrone G, Spagnuolo G, Vitelli M (2004) Optimizing sampling rate of P and O MPPT technique. In: *Proceedings IEEE PESC*, pp 1945–1949
3. Esram T, Chapman PL (2007) Comparison of photovoltaic array maximum power point tracking techniques. *IEEE Trans Energy Convers* 22(2):439–449
4. Mokhtar A et al. (2011) Design and development of energy-free solar street LED light system. In: *2011 IEEE PES conference on innovative smart grid technologies-middle east*, IEEE

5. Hossain E, Muhida R, Ali A (2008) Efficiency improvement of solar cell using compound parabolic concentrator and sun tracking system. In: 2008 IEEE Canada electric power conference, IEEE
6. Mamarelis E, Petrone G, Spagnuolo G (2014) A two-steps algorithm improving the P and O steady state MPPT efficiency. *Appl Energy* 113:414–421
7. Arshad R et al. (2014) Improvement in solar panel efficiency using solar concentration by simple mirrors and by cooling. In: 2014 International conference on robotics and emerging allied technologies in engineering (iCREATE), IEEE
8. Winston DP et al. (2020) Performance improvement of solar PV array topologies during various partial shading conditions. *Sol Energy* 196:28–242
9. Pakkiraiah B, Sukumar GD (2016) Research survey on various MPPT performance issues to improve the solar PV system efficiency
10. Hua C-C, Fang Y-H, Wong C-J (2018) Improved solar system with maximum power point tracking. *IET Renew Power Gener* 12(7):806–814
11. Zakzouk NE et al. (2016) Improved performance low-cost incremental conductance PV MPPT technique. *IET Renew Power Gener* 10(4):561–574
12. Yilmaz U, Turksoy O, Teke A (2019) Improved MPPT method to increase accuracy and speed in photovoltaic systems under variable atmospheric conditions. *Int J Electr Power Energy Syst* 113:634–651
13. Abdel-Salam M, El-Mohandes M-T, Goda M (2018) An improved perturb-and-observe based MPPT method for PV systems under varying irradiation levels. *Sol Energy* 171:547–561
14. Kazmi SMR et al. (2009) An improved and very efficient MPPT controller for PV systems subjected to rapidly varying atmospheric conditions and partial shading. In: 2009 Australasian universities power engineering conference, IEEE
15. Prudhvi P, Sai PC (2012) Efficiency improvement of solar PV panels using active cooling. In: 2012 11th International conference on environment and electrical engineering, IEEE
16. Alhammad YA, Al-Azzawi WF (2015) Exploitation the waste energy in hybrid cars to improve the efficiency of solar cell panel as an auxiliary power supply. In: 2015 10th International symposium on mechatronics and its applications (ISMA), IEEE
17. Nazar R (2015) Paper title: improvement of efficiency of Solar panel using different methods. *Int J Electr Electron Eng (IJEEE)* 7
18. Overall efficiency of the grid connected photovoltaic inverters, European Standard EN 50530, 2010
19. Sera D et al. (2006) Improved MPPT algorithms for rapidly changing environmental conditions. In: 2006 12th International power electronics and motion control conference, IEEE
20. Gomathy SSTS, Saravanan S, Thangavel S (2012) Design and implementation of maximum power point tracking (MPPT) algorithm for a standalone PV system. *Int J Sci Eng Res* 3(3):1–7

Hardware Implementation of Machine Vision System for Component Detection



P. Smruthi, K. B. Prajna, Jibin G. John, and Aslam Taj Pasha

1 Introduction

Automation has revolutionized fabricating in which complex operations have been broken down into basic step-by-step instruction which is revised by a machine. This mechanism will require for the efficient gathering, and review has been realized completely in diverse manufacturing forms. These assignments have been more often than not done by the human workers, due the continuous fault in manual inspection which made the machine vision system more attractive and useful [1]. The relevance and importance of machine vision systems in machine tool industrial applications are described. A basic machine vision model and the system design are detailed. The system design includes various sub-systems which are chosen based on the application. It describes the vision system as four-step processes. Starting with acquiring the image, converting the pixels to array of images, reviewing and analyzing the image and finally sending the output to external devices. The sequence and the operation of each element of the system are explained. Finally, application of machine vision system is described [2]. Machine vision (MV) is a kind of technology which enables the computing devices to capture, analysis and inspect still or moving images. It

P. Smruthi

VLSI and Embedded Systems, Nitte Meenakshi Institute of Technology, Bengaluru, India

K. B. Prajna (✉)

Department of Electronics and Communication, Nitte Meenakshi Institute of Technology, Bengaluru, India

e-mail: prajna.kb@nmit.ac.in

J. G. John · A. T. Pasha

Research and Technology Development Group, ACE Designers Ltd, Bengaluru, India

e-mail: jibin_gj@acedesigners.co.in

A. T. Pasha

e-mail: aslamtaj@acedesigners.co.in

includes all applications in which a mixture of computer hardware and software application provides functional mechanism to devices in the execution of their functions in reference with the captured images. MVS used in production line follows basically four steps, as the component comes near the sensor, then it sends signal to controller to trigger the camera to acquire the image of the component and a LED bulb to highlight key features. Then, the input data from the captured image is sent to controller which is deployed with algorithm and software. A novel Hough transform-based technique is used for circle recognition. For edge identification in the voting process, the design uses a scan line-based ball detection algorithm. The simulation results reveal that an image with a VGA resolution of 640×480 is processed in 10 ms, i.e., the design suggests that the processed architecture can meet the required speed [3]. Industrial controller inspects the image and converts into digital output, which indicates the component is accepted or rejected.

2 Literature Survey

A machine vision system is made up of several components, from the camera that captures an image for examination through the processing engine that generates and conveys the results. Incoming picture data is processed by software, which then provides pass/fail outcomes. The collected images are stored in the database which is used in the testing process. As the testing process starts, it will access the image and compare it with the obtained image. The parallel communication protocol is used to communicate between the industrial controller and peripheral interface. Machine vision software can take numerous forms and used for an application-oriented purpose.

Golnabia et al. [1] the relevance and importance of machine vision systems in machine tool industrial applications are described. A basic machine vision model and the system design are detailed. The system design includes various sub-systems which are chosen based on the application. It describes the vision system as four-step processes. Starting with acquiring the image, converting the pixels to array of images, reviewing and analyzing the image and finally sending the output to external devices. The sequence and the operation of each element of the system are explained. Finally, application of machine vision system is described.

Al-Kindi et al. [2] propose using a vision-based monitoring and regulator system in manufacturing lane to increase CNC machining performance. A solution is presented and developed to enable the integration of vision processing with CNC machines by addressing a number of pinpointed concerns that prevent such integration. These processes are being refined into a practical methodology that used on laboratory-scale CNC milling machines. To generalize the findings, two distinct kinds of bench-type CNC machines are used. Each of the two CNC 9 machines has two cameras positioned on the machine spindle to provide accurate picture data in the cutting direction. An indicative parameter is proposed and used to evaluate the tool imprints that outcome. The total results demonstrate the validity of the approach and motivate additional

research and development in order to create real-world intelligent vision-controlled CNC machines on a large scale.

Baginski et al. [4] propose how capable the industrial controller has been controlling the technology of automation. This details about how system put forth the automation encopresis of variety of hardware and software components. These steps make development of independent manufacturing, less time consumption. The problem of integrating hardware has been prevented in open-loop control system. The open control user group has defined the CALL interfaces. These CALL interfaces allow for straightforward data sharing not just during the design phase of an automated solution, but also throughout its execution. CALL also allows access to dispersed components through a fieldbus system like INTERBUS.

Liao [5] deal with an industrial camera having high speed, miniaturization, portable, low power and smart. Color photographs from a real-world scene with controlled light conditions are captured and processed by the camera. A color image processing pipeline has also been built into the camera to ensure high-quality photographs. The automated white balance (AWB) algorithm has been enhanced to improve the strength of the proposed camera to illumination fluctuations. On real-world photographs, the modified method achieves a good white balance processing result.

He Xiangyan et al. [6] the fuzzy PID algorithm-based dimming of high-power LEDs for machine vision lighting is intended to sustain continuous detection of ambient lighting. The proposed system consists of two components: hardware and software. Among the hardware components that employ DSP as the primary control device are the constant current drive circuit for high-power LED, the temperature of LED measurement circuit and the ambient illumination intensity detection circuit. If the measured room lighting differs from the specified room lighting, the fuzzy PID algorithm takes off and automatically changes the PID value depending on the error and error rate in DSP chip. The fuzzy PID algorithm regulates the duty cycle of the PWM pulse produced by the DSP. The experimental result demonstrates that the measured ambient lighting may be kept at an adaptable constant. The use of this technology offers a set of standards for energy-efficient lighting and automated dimming control of manufacturing lane.

He Xiangyan et al. [7] present energy savings, environmental protection, anti-seismic properties and high luminosity that are just a few of the benefits of high-power LEDs. As a result, it is used widely used in various sector, including lighting system. The image quality of examined object is strongly reliant on stability of the light source. As a consequence, integrating an automated dimming technology with a high-power LED enhances work performance and delivers more energy-efficient lighting. The light intensity of the high-power LED automatically adjusts based on the image definition of the measured object from the detection system's computer, which has an image sensor. The following components make up the hardware element of the automatic dimming system: a driver circuit for constant current LED, a processor of 16 MB and an environment setting light sensor.

Deng et al. [8] describe the detection of contour using Gaussian filter for better image segmentation. The contours of a grayscale image are determined via edge

detection. Only pixels points with a different intensity are considered in edge detection. By combining the gPb-owt-ucm method with Gaussian smoothing filter, the performance of the image segmentation has been improved. The change of color space is conducted and explored on the same dataset for improved performance. As a result, an improved accuracy of several datasets is obtained.

Seo et al. [9] describe a novel Hough transform-based technique for circle recognition. For edge identification in the voting process, the design uses a scan line-based ball detection algorithm. The simulation results reveal that an image with a VGA resolution of 640×480 is processed in 10 ms, i.e., the design suggests that the processed architecture can meet the required speed.

3 Methodology

A. Inspect the Presences of Component

A proximity sensor is interfaced to digital input–output port of the controller. The signal from the sensor is accessed by the controller using the input–output driver. As the component comes near the proximity sensor, it senses the presence and a signal is sent to the controller to start the process.

B. Collecting Sample Data

As the sensor sends the signal to controller, it triggers the camera, as a first step to acquire several samples of the target object.

C. Basic Pre-processing of Image

Then analyze the images for how clearly it shows the target object. Image pre-processing is the process on images to obtain a smooth and noise-reduced image. These operations enhance image clarity by decreasing the entropy/noise in the information. The pre-processing aims at improving the image quality and brightening that overwhelms undesired distortions and improves image features related for further processing.

D. Performing Featured-Based Approach

A feature is a piece of information about the certain region of an image which is relevant for solving the computational task related to an application. Basically, featured-based approach is to find certain region of the image has certain properties. These features are the detailed structures in the image such as points, edges, lines or circles. Feature-based approach involves two stages:

- i. Firstly, the features are detected in two or more successive images. This act of feature extraction is to obtain the amount of information to be processed by eliminating the unimportant parts.
- ii. Secondly, these features are matched between the data samples. The feature detection stage involves features to be located precisely and reliably. Feature detectors are used to obtain the feature more accurately. The feature matching stage faces difficulties in matching, such as if the image displacement is known to be smaller than the distance between features. Some of the

methods to solve these difficulties are by using edge detection technique. The edge detection theory is well advanced, compared to any of the two-dimensional feature detection and most edge detection methods, either finding zero-crossings in the Laplacian of a Gaussian of the image.

E. Performing Feature Matching

Feature matching is a powerful technique for detecting a detailed target in a specific area of an image. The feature matching method compares and analyzes point correspondences between the collected image and the obtained image. If any region of the image is larger than the maximum value, that region of the part of an image is targeted and deemed to contain the reference object.

4 Design

Machine vision system refers to the processes and methods used to extract data from an image on an automatic basis, with the end result being another image format. The extracted information consists of a set of data such as the position, orientation and identification of each object, which can be a good-part/bad-part signal. This signal is used for the further manufacturing process. It can be also used for applications such as industrial robot and process guidance, automatic inspection, security monitoring and vehicle guiding. The image depicts the design of a machine vision system. It is built out of a conveyor that moves the component toward the proximity sensor. A signal is sent to the controller to capture the image which is processed in the controller after it is captured. Higher processing performance requirements are given by rule-based training. Multiple stages of processing are typically performed in a sequence that results in the desired outcome (Fig. 1).

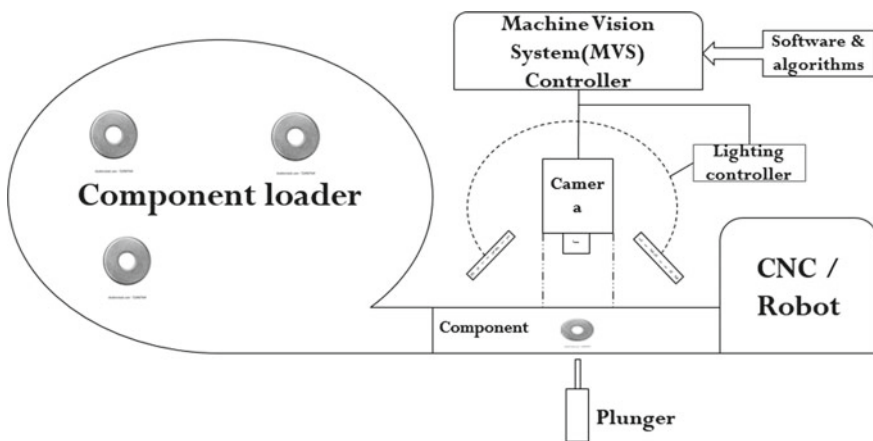


Fig. 1 Design of machine vision system

A typical sequence might begin with tools that modify the image, followed by data extraction from those objects, followed by communication of that data or comparing it to target values to produce and convey “pass/fail.”

4.1 Operation of a Machine Vision System

Figure depicts the key mechanisms of a visual system. Numerous operations, such as image acquisition, pre-processing, segmentation and feature extraction, are carried out in a continuous and real-time way. When the proximity sensor detects an object, it sends a signal to the controller, which causes the camera to record an image. In a vision system, the optical-acquisition sub-system converts optical image information into an array of numerical data that can be processed by a processor. The light from a source illuminates a segment of an image to the image sensor, resulting in an optical picture. Image arrays are used to convert an optical image to an electrical signal that can then be transformed to a digital image. In general, cameras with either line scan or area scan elements offer significant advantages. For light detection, the camera system may use a charge coupled device (CCD) sensor or a CMOS sensor.

All pre-processing, segmentation, feature extraction and other operations are accomplished using digital images. At this point, the image classification and interpretation are complete, and the actuation operation can be performed in order to interact with the division. Thus, the actuation sub-system communicates with the controller in order to regulate or change any given prerequisite in order to acquire a better image acquisition. A visual system performs the following functions: picture capture, image processing and item recognition within an object collection. The scene is illuminated by light from a source, and image sensors generate a photosensitive image. Image processing methods include image sensing, displaying image data and digitization. Image processing is the process of changing and creating the pixel values of a digital image in order to provide a more suitable form for subsequent processes. Segmentation is the process of dividing an image into expressive sections that correspond to different parts of the image (Fig. 2).

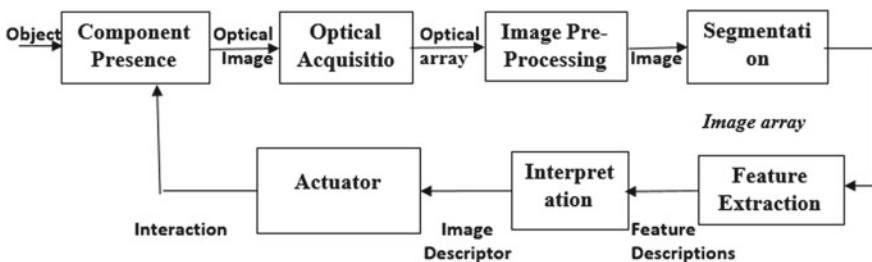


Fig. 2 Block diagram of machine vision system

Segmentation is the process of dividing an image into expressive sections that correspond to different parts of the image. The goal of feature extraction is to classify the hierarchical qualities, or picture features, contained inside a component. Pattern categorization refers to the process of identifying an unidentified object inside an image as belonging to one of several possible object categories.

4.2 Hardware Architecture

The Machine Vision system consists of several elements such as camera, lens, component presence sensor, solenoid cylinder. The basic elements of the system are shown in Fig. 3.

1. **Camera:** Nearest camera resolution available is 0.3 MP (640 × 480 pixels).
2. **Industrial Controller Specifications**
 - i. Power supply—20–50 V DC, 4.63 A
 - ii. Processor—Intel i3
 - iii. RAM—8 GB
 - iv. RS232—2 ports
 - v. I/O ports—DIO 8 IP/OP
 - vi. 4 USB
 - vii. Fanless
 - viii. IP64
3. **Workpiece Conveyor Belt:** The banded piston is placed on the belt. The continuous movement of the belt is stopped as the sensor signal is detected.

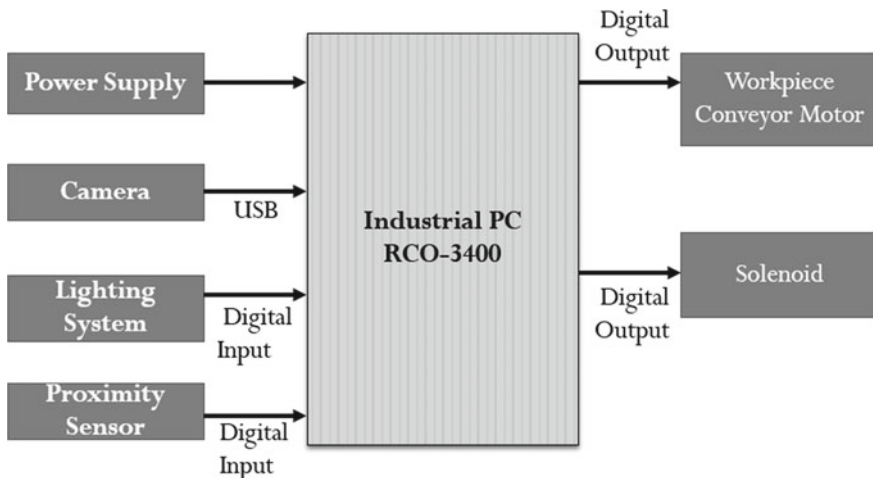


Fig. 3 Elements of machine vision system

4. **Proximity Sensor:** It is the object detection sensor. NPN-type sensor is used to connect to the controller output. As the component is passed near the sensor, it detects it and then sends the signal to the controller. Using image processing, the circle is detected. The algorithm used is Hough circle detection.
5. **Controller DIO:** The controller DIO is of NPN-type system. It is the sinking system where 0 V is given to the controller.
6. **Solenoid Plunger:** This is the pneumatic cylinder which is used in reject process.

The banded piston component is used for the feature identification. As the component is placed on the conveyor belt, it is on. The component is moving on the belt. The proximity sensor is used to identify the component once the component is detected. The conveyor motor is stopped. The image processing is performed, and the decision is taken where it is a good or bad component (Fig. 4).

Figure 5 shows the flowchart of hardware implementation of MVS.

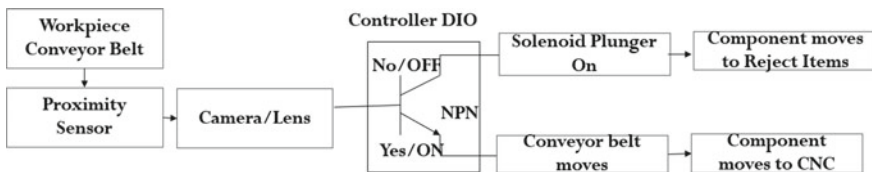


Fig. 4 Hardware implementation of machine vision system

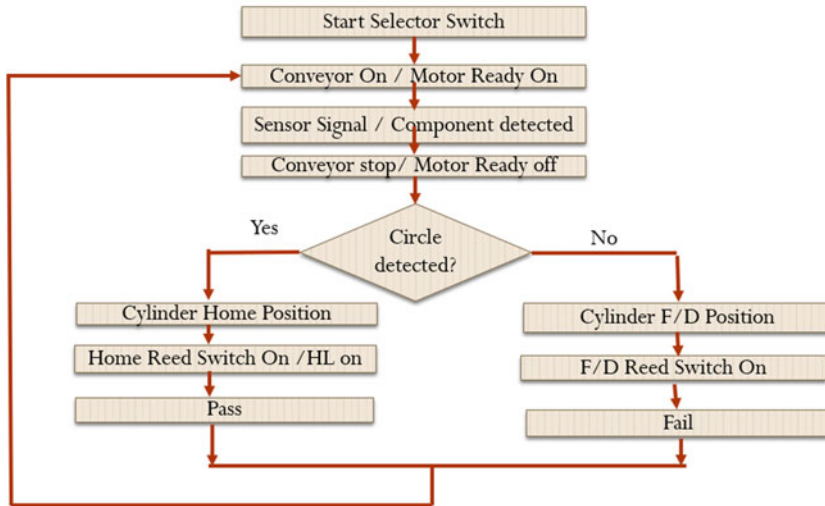
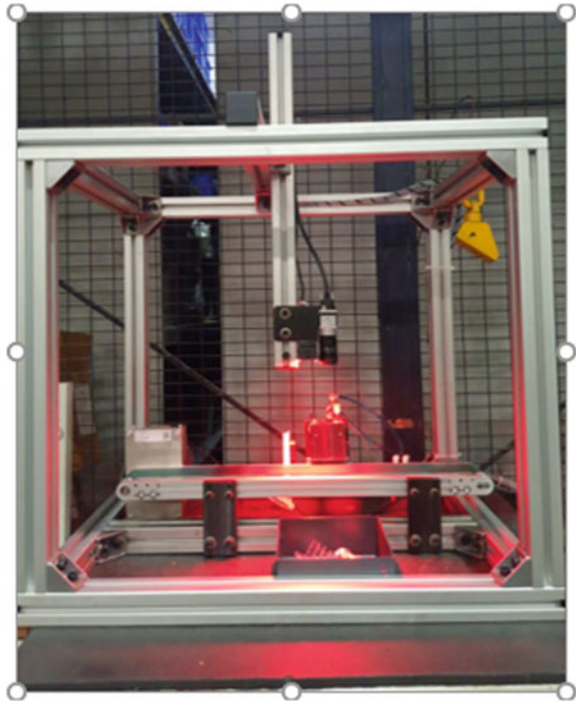


Fig. 5 Flowchart of hardware implementation

Fig. 6 Machine vision system fetcher



4.3 Machine Vision System Fetcher

The placement of each component in the fetcher (Fig. 6) is according to the process. This fetcher is a prototype which consists of work piece conveyor where the components are loaded for the process [10]. The sensor is placed based on the sensing distance and also the sensing range. As the component passes the sensor, the light ray which the sensor continuously transmits breaks and a reflection signal is sent back. The mechanical fetcher is designed using aluminum sheet metal. The size of sheet is calibrated using software and designed using the fetcher. The entire architecture is designed to inspect, analyze and identify the moving component. The process is carried out in a continuous manner.

4.4 Electrical Cabinet Box

The electrical cabinet box for the machine vision system is as shown in the figure. Electrical cabinet (Fig. 7) consists of element such as controller, SMPS, SSR relay, MCB. Supply to the cabinet is given by 3-phase poles, from which single pole is

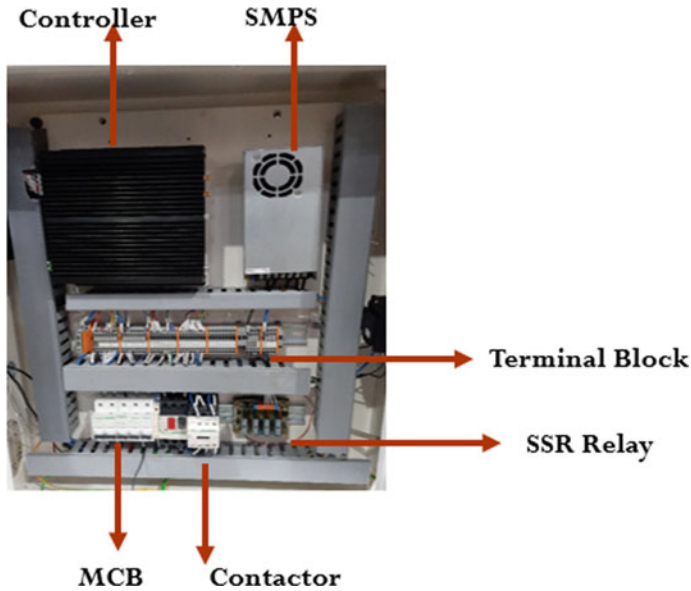


Fig. 7 Electrical cabinet box

taken to the entire cabinet. The single-pole supply is given to the SMPS directly for rest; all elements in it are given through terminal of SMPS.

The controller power is given through 10 A MCB to avoid the shorts. This MCB is connected to the SMPS terminal block. Even the exhaust fan, lighting is connected to the SMPS terminal block through MCB. The terminal block is used as the connection point to all the elements in the cabinet. Contactor is used to control the speed of the motor using digital input–output signal. If any short in contactor, the motor tripped signal is given to provide to stop the conveyor motor. Two solid-state relays are used to control the pneumatic cylinder forward and home position. The supply to the SSR relay is given through the 24 V.

4.5 Electrical Planning Diagram

EPLAN is computer-aided engineering (CAE) design software solution. Figure 8 describes the electrical connectivity of the electrical cabinet. Three-phase supply is taken as a single pole using the MCB. This single-phase supply is connected controller, SMPS, exhaust fan. The length of the wire for supply is 10 mm, whereas the connection to SMPS controller is 6 mm.

The conveyor motor and sensor connection are shown below. The cable length from sensor to DIO port is 0.5 mm. From the motor to contactor, the cable length is 6 mm (Fig. 9).

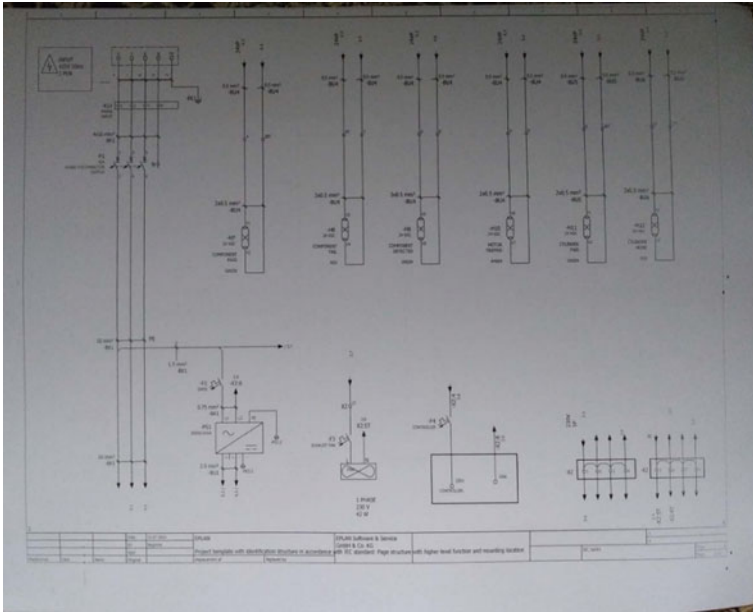


Fig. 8 Electrical diagram for MVS

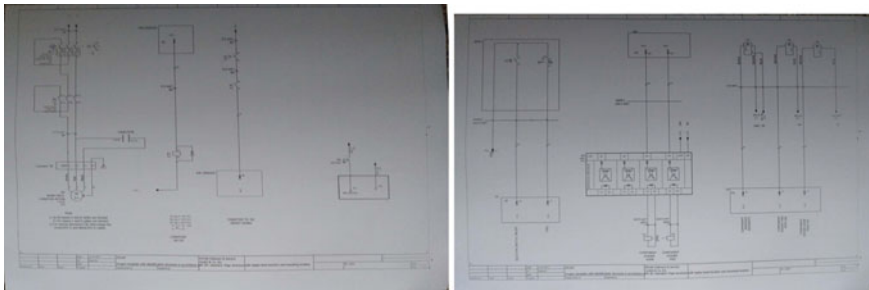


Fig. 9 Conveyor motor and sensor connection of MVS electrical diagram for MVS

The software for machine vision system deals with capture of image to the decision-making which is done using PyCharm IDE. The main goal of PyInstaller is to be compatible with third-party packages to create an executable file. The source code is converted to executable format which is understood by the controller. The objective is feature detection for a dot of 0.5 mm. This is done by using rule-based technique. In this method, the real-time image is captured and processing is done. The processing methods are blurring, thresholding and Hough circle detection. Once the processing is completed, the decision is taken by software as accepted or rejected component.

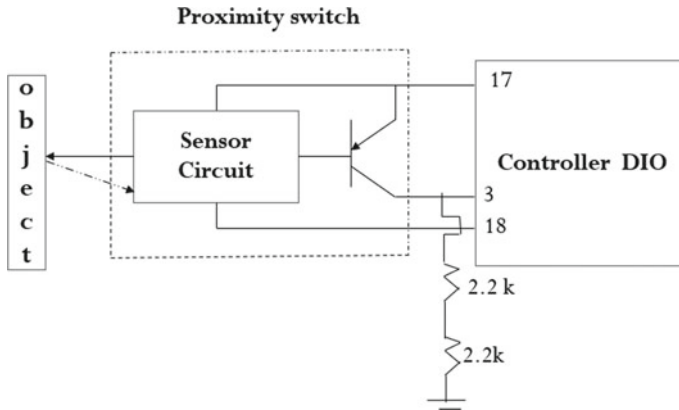


Fig. 10 Proximity sensor component detection

4.6 Hardware Interface

A proximity sensor is a non-contact sensor. Proximity sensors detect objects when the light emitted from the sensor is reflected back at the photoelectric receiver. Through-beam sensors detect targets when the target breaks the beam of light between the sensor's emitter and receiver (Fig. 10).

Pin 17 and Pin 18 of controller DIO are 24 V and ground. Pin 3 is digital input which is connected to the signal of proximity sensor. The sensor circuit used here is the LC oscillation which is used to generate a designed resonant oscillation continuously. A pull-down resistor of 2.2 k is connected to the signal to maintain always 0 V at the signal. Since the PNP-type sensor is connected to make it NPN, it is connected to resistor. The below is the code for accessing the input pins:

```
Mydll = cdll.LoadLibrary (../inpoutx64.dll).
```

```
Driver = Mydll.IsInpOutDriverOpen.
```

```
Output = mydll.Inp32(0xA03).
```

A solenoid plunger is the moving part of a solenoid that transfers linear motion from the solenoid to the component that it is designed to operate. It controls the flow direction of compressed air. A moving part inside the valve blocks or opens the ports of the valve. SSR relay has a Zener diode and optocoupler; this is used to switch the position of the cylinder when the voltage is applied (Fig. 11).

A conveyor belt works by using two motorized pulleys that loop over a long stretch of thick, durable material. When motors in the pulleys operate at the same speed and spin in the same direction, the belt moves between the two. The width of the belt 30 mm, length of the belt 100 mm, the speed of the belt 5 mm/s (Fig. 12) [11–13].

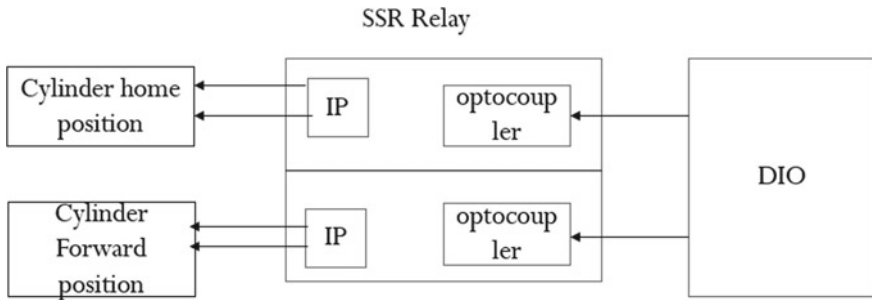


Fig. 11 Solenoid plunger

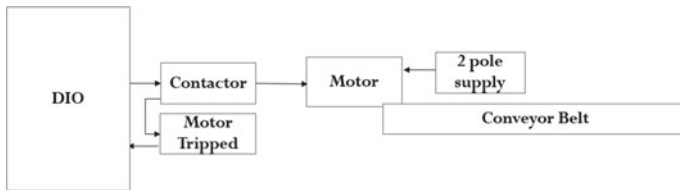


Fig. 12 Conveyor belt

5 Results

Implementation of machine vision system for component detection was developed. This system was able to identify the six circles in component using Hough circle detection technique. The system was taking decision based on the circles detected whether to pass or fail the component. If the component does not have six circles, it was pushed to rejected side. Once the circles are detected, it was moved to accepted box. Visual representation of this was shown using graphical user interface.

Graphical user interface is visual appearance of the code capturing. The end-user can visually see the operation which is happening in the image. The basic graphical user interface is created only to detect a particular feature inside the component (Fig. 13).

The above figure shows the basic graphical user interface for banded piston component detection. This consists of more elements such as image pre-processing, detection algorithm, output display, user input ROI and saving the setting [14–18]. The pre-processing elements in the GUI reduce the noise and the disturbance in the capture image. These technique make the image clear. Detection algorithm is used to get feature properly in the output as also shown in the GUI screen.

The machine vision system component testing was done. The features detected are outer diameter of 30 mm with 6 dots with in proper orientation. The radius of each dot is 0.5 mm. The distance between each dot is of 2 mm. The trail round was carried for component (Fig. 14).

The acceptance and rejection details are given in Table 1.

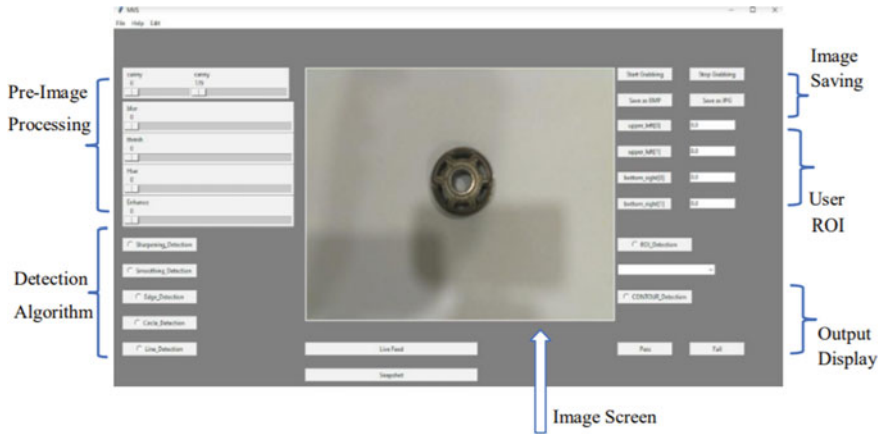


Fig. 13 Graphical user interface

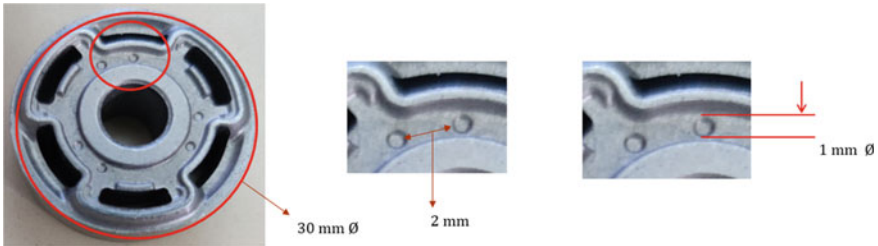




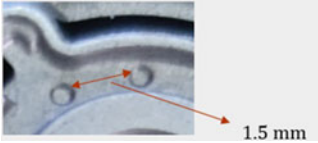

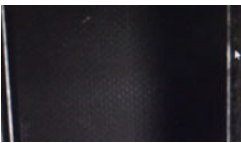



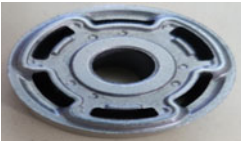

Fig. 14 30 mm diameter–banded piston used in shock absorbers

6 Conclusion

The distribution of defects in components and the probability of manual inspection and automated system are shown. The manual operator can miss one defect in every 5 min; he could potentially accept one component with a life-endangering defect for perhaps every 10,000 pieces he tests. The automated system will help inspection by only 50% increase when compared to the manual inspection (Fig. 15 and Table 2).

The details of number of components identified in one cycle for different conveyor belt speed is shown in Table 2.

Table 1 Acceptation/rejection table

Trail no	Accept/reject	Defect	Status	Remarks
1	Rejected			An extra circle was detected. Other than 6 dots. This dot was of same radius as the other 6 six circles
2	Rejected			The distance between two circle was not 2 mm
3	Rejected			Proper lighting was not available, then the component will not be visible
4	Rejected			The feature with different diameter. The diameter greater 1 mm is not detected
5	Accepted			Proper orientation and placement of features are identified correctly

7 Future Scope

The developed system is to be flexible for variety of components, and graphical user interface is to be upgraded. The system needs to be integrated with CNC machine for the complete process.

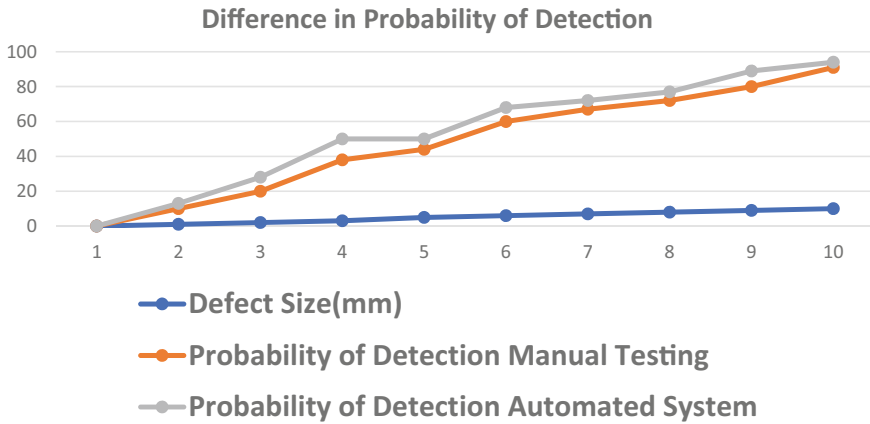


Fig. 15 Difference in probability of detection chart

Table 2 Component detection for different belt speed

Conveyor belt speed	No. of components identified in one cycle
30 rpm	4
80 rpm	7
90 rpm	10

References

1. Golnabi H, Asadpour A (2007) Design and application of industrial Machine vision systems. *Robot Comput Integr Manufactu* 23(6):630–637
2. Al-Kindi G, Zughaer H (2012) An approach to improved CNC machining using vision-based system. *Mater Manuf Process* 27(7):765–774
3. He X et al. (2020) Design of high-power LED automatic dimming system for light source of on-line detection system. In: 2020 IEEE 5th Information technology and mechatronics engineering conference (ITOEC), IEEE
4. Baginski A, Covarrubias G (1997) Open control-the standard for PC-based automation technology. In: Proceedings 1997 IEEE international workshop on factory communication systems. WFC'S'97, IEEE
5. Liao W et al. (2010) An industrial camera for color image processing with mixed lighting conditions. In: 2010 The 2nd international conference on computer and automation engineering (ICCAE), vol 5, IEEE
6. He X et al (2020) An adaptive dimming system of high-power LED based on fuzzy PID control algorithm for machine vision lighting. In: 2020 IEEE 4th Information technology, networking, electronic and automation control conference (ITNEC), vol 1, IEEE
7. He X et al (2020) Design of high-power LED automatic dimming system for light source of on-line detection system. In: 2020 IEEE 5th Information technology and mechatronics engineering conference (ITOEC). IEEE
8. Deng G, Cahill LW (1993) An adaptive Gaussian filter for noise reduction and edge detection. In: 1993 IEEE conference record nuclear science symposium and medical imaging conference, IEEE

9. Seo SW, Kim M (2015) Efficient architecture for circle detection using Hough transform. In: 2015 International conference on information and communication technology convergence (ICTC), IEEE
10. Noble FK (2016) Comparison of openCV's feature detectors and feature matchers. In: 2016 23rd International conference on mechatronics and machine vision in practice (M2VIP), IEEE
11. Lü C, Wang X, Shen Y (2013) A stereo vision measurement system based on openCV. In: 2013 6th International congress on image and signal processing (CISP), vol 2, IEEE
12. Higuchi T et al. (2019) CIPy: a NumPy-compatible library. accelerated with openCL. In: 2019 IEEE International parallel and distributed processing symposium workshops (IPDPSW), IEEE
13. Seman P et al. (2013) New possibilities of industrial programming software. In: 2013 International conference on process control (PC), IEEE
14. Martin S (1990) PC-based data acquisition in an industrial environment. In: IEE Colloquium on PC-based instrumentation, IET
15. Wuth SN, Coetzee R, Levitt SP (2004) Creating a python GUI for a C++ image processing library. In: 2004 IEEE Africon. 7th Africon conference in Africa (IEEE Cat. No. 04CH37590), vol 2, IEEE
16. Hernandez-Ordonez M et al. (2007) Development of an educational simulator and graphical user interface for diabetic patients. In: 2007 4th International conference on electrical and electronics engineering, IEEE
17. Bright G, Potgieter J (1998) PC-based mechatronic robotic plug and play system for part assembly operations. In: IEEE International symposium on industrial electronics. Proceedings. ISIE'98 (Cat. No. 98TH8357), vol 2, IEEE
18. Canny J (1986) A computational approach to edge detection. *IEEE Trans Pattern Anal Mach Intell* 6:679–698
19. Yan S, Yao L, Zhang Y (2019) Design of industrial robot sorting system based on smart camera. In: 2019 International conference on artificial intelligence and advanced manufacturing (AIAM), IEEE
20. Chakole S, Ukani N (2020) Low-cost vision system for pick and place application using camera and ABB industrial robot. In: 2020 11th International conference on computing, communication and networking technologies (ICCCNT), IEEE

DEOMAC—Decentralized and Energy-Efficient Framework for Offloading Mobile Applications to Clouds



A. L. Shanthi and V. Ramesh

1 Introduction

The deployment and usage of smartphone applications and platforms have increased dramatically, around the world for various tasks such as sending emails, watching videos, online banking, browsing the Internet, navigating using online maps and using social media [1]. To perform the above tasks, different applications were utilized. Due to rapid growth of user demands and mobile applications, the Quality of Service (QoS) is restrained by limitations at the mobile side such as limited available connectivity, finite energy, resource limitations, and shared wireless medium. Running complex applications on resource-limited mobile devices which have slow processors, limited storage abilities and low battery power, lowered magnitude which will widening the gap between the availability of limited resources and the demands of these complex programs resulting into lowered performances and slow functionality of mobile devices [2].

However, offloading computational task could consume a huge amount of energy and incur delay between cloud clones and mobile devices which could involve considerable communications. Hence, offloading decision should be carefully made at each mobile device for the execution of computation task either at cloud or mobile, by considering the status of the wireless network as well as delay and energy consumption of various operations. Clearly, computation offloading is recommended only when the mobile device execution time is higher than the cloud execution time. Many factors can impact the offloading process and could influence the offloading decision [3–7].

Earlier research suggested a threshold-based policy for making the offloading decision. Apparently, it is very tedious to set a common threshold as it is often

A. L. Shanthi (✉) · V. Ramesh
Department of CSA, SCSVMV, Kanchipuram, TamilNadu 631561, India
e-mail: shanthimanipsg@gmail.com

application dependent. For a mobile device, where environment conditions change dynamically and continuously, it is very tedious to decide the offloading decision solely based on a static threshold. Moreover, a threshold-based policy always makes the same static offloading decision without considering dynamic profiling parameters of mobile devices [8].

An effective decision needs to be taken to ascertain whether the offloading is beneficial or unprofitable in terms of computation execution time and energy conservation. In earlier research, offloading frameworks considered bandwidth, latency, CPU load, available memory as QoS parameters which is not sufficient for making a dynamic offloading decision. There is a need to improve the QoS factors considering for decision-making process. Hence, it is necessary to develop offloading framework with augmentation decision-making strategies to improve energy efficiency of mobile devices. Very few works are concentrated on the above issues and these quandaries are to do inspire the research. The main goal of this research is to propose a decentralized and energy-efficient offloading framework to accomplish offloading process in a dynamic environment. In this research, the neuro fuzzy is utilized for decision-making model and ACSOA makes the decentralized resource allocation process.

2 Literature Review

Several offloading frameworks and models exist to enhance the execution capability and energy optimization in mobile cloud environment but each of these has their own advantages and limitations. Conventional offloading frameworks used adaptive algorithms that migrates heavy computations to the remote servers. These frameworks employ different levels for offloading applications at runtime, but it includes migration cost of the computational components of the mobile application.

The main goal of this frameworks is to find the optimum solution to offload intensive parts of the applications with proper use of assets. However, every framework has its own benefits and limitations. Mostly frameworks have not considered dynamic execution time which is a major aspect for real-time applications. Some of the frameworks has not focused on resource allocation at cloud environment during runtime.

Hassan et al. [14] introduced the POMAC framework for a transparent and dynamic offloading in mobile. Mobile computation offloading operations based on two issues in offloading executions to the cloud. Their scheme first decided if code should be offloaded and, on a decision, to offload, it was done transparently. They implemented their prototype on the Dalvik VM. Their introductory evaluations showed the proposed schemes worked well on real-time applications and outperformed many existing schemes.

The proposed POMAC implemented an MLP-based decision engine that captured dynamic elements and relationships between the elements captured. POMAC also implemented a transparent method level offloading system. This framework mainly

considered network profiling parameters rather than device and application profiling which is not sufficient for making a dynamic offloading decision that gives scope for further research.

Energy savings in smart mobile devices using a MAUI framework were proposed by Cuervo et al. [15]. The proposed framework's targeted offloading processes in a highly dynamic approach as it continuously profiled the processes. The scheme mocks the complexity of remote executions from smart mobile devices giving an impression that the application is executed on the smart mobile device locally. MAUI partitions are based on code annotations and specify components that can be executed on cloud server remotely. MAUI profiler assesses device characteristics after which it monitors them and characteristics of network during the whole execution time as these parameters can change and be the cause of inaccurate measurements and thus erroneous decisions. Offloading decisions happen at runtime where the framework selects components that are executed remotely based on the decision from the MAUI solver which takes MAUI profiler inputs.

Existing frameworks require special compilations/modifications in source codes or binaries which makes it complex for implementations or adaptations. Effective decisions need to ensure offloading as beneficial or unprofitable in their energy conservations or execution times. Predicting offloadable decisions for smart mobile devices based on analysis of static thresholds is a laborious process as parameters may change dynamically and frequently. Static profiling leads to a constant offloading decision. Frameworks that consider latency delays, bandwidths, available memory spaces, and CPU loads only as QoS parameters, do not take dynamic offloading decisions creating a need to improve on QoS factors in decision-making.

Previous frameworks did not focus on the energy efficiency of smart mobile devices in cloud resource allocations which makes it necessary to consider the creation of optimized decision-making in smart mobile devices. However, the decisions should satisfy the reduction of overheads in offloading. One major issue in offloading has been in the accurate allocation of cloud resources for obtaining desired results at minimal execution costs. Most studies have catered to mobile computation offloading operations in the cloud without specifying virtual machine allocations for reduced execution times and improved performances (Table 1).

Contribution of the paper

- To propose and implement a novel mobile application offloading framework that expand the capabilities of mobile devices which includes minimized energy consumption and delay for executing a computation-intensive task.
- To find the felicitous classifier that should have the properties to accurately make the offloading decision which is highly precise and lightweight run on mobile devices.
- To apply decentralized resource allocation in cloud for amending energy efficiency and reducing the computational involution.

Table 1 A comparative review of offloading framework

Framework	Partitioning	Preparation	Decision	Offloading mechanism	Contribution	Automation
MAUI E. Cuervo	Annotates methods as local/remote	Creates two applications one for SMD and one for cloud programming reflection is used to classify offloadable methods	Dynamic based on inputs from the MAUI profiler and solver	Code It does not allow partial executions in the remote server	Offloading code with energy awareness	Method level
Cuckoo R. Kemp	Partitions based on the activity of Androids, Only services are offloaded	JVM in the cloud for executions	Dynamic: services method invocations are offloaded based on cuckoo decisions of remote resource availability	Code Received method calls are evaluated for offloads using heuristic information	Simplifying SMD application developments for offloading benefits	Method level
Clone Cloud B. G. Chun	Partitions using static program analysis and profiling	SMD software is duplicated on the server	Threads which are dynamic migrated from SMD to cloud clone	Offloaded code runs on a VM	Elasticity in applications adaptation in the execution of MCO operations	Thread level
Thinkair S. Kosta	Tasks are split and distributed to multiple VMs	Cloud VM and parallel processing used for managing VM in smart mobile devices	Dynamic	Supports dynamic on-demand allocation of resources for user satisfaction	Cloud VM manager splits tasks between other VMs with parallel processing	Method level

2.1 DEOMAC Architecture

The architecture of DFOMAC framework is shown in Fig. 1. The main components of the architecture are profiler, decision-making, and resource allocation in cloud server.

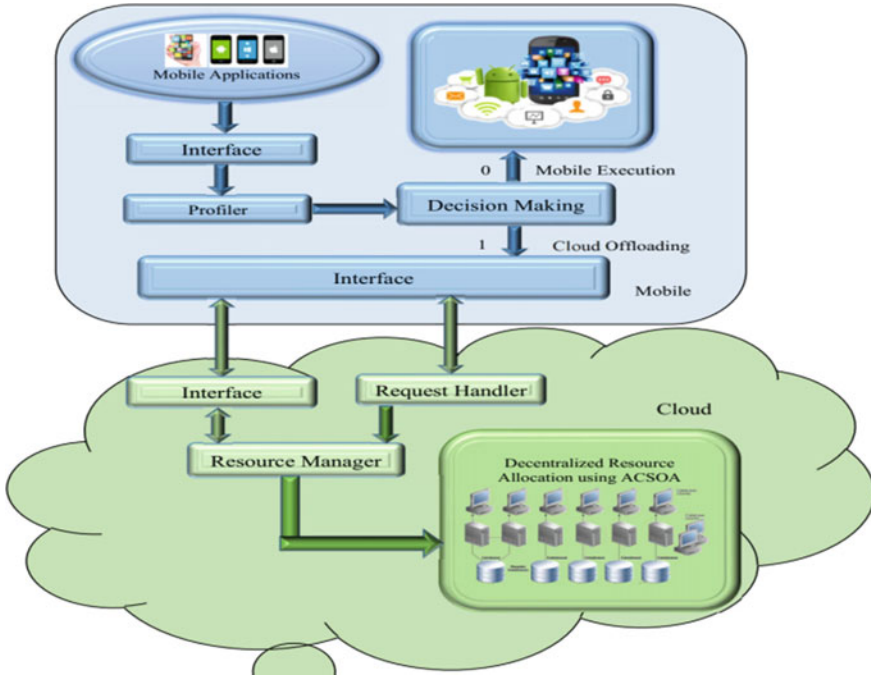


Fig. 1 DEOMAC framework architecture

Profiler

The profiler is the fundamental part of the framework used to gather QoS parameters such as transfer speed, accessible RAM in portable and cloud, information size and battery level and average execution time at mobile and cloud, respectively. Typically, the working method of the profiler is gathering the data and enhancements into the database. Profiler gathers the data that causes the framework to settle on precise choice with respect to offloading by gathering context information about gadget, application and network. Mobile device profiler states the status of versatile such as RAM and battery status of the mobile device. Network profiler screens the data about network condition such as type, signal quality, and transfer speed. Application profiler gathers the highlights about input size and average execution time. Average execution time is determined by taking average of what amount of time required to execute compute intensive task at cloud and mobile. By looking at proc/meminfo records of android, RAM accessibility of versatile and cloud is resolved.

Decision-Making

To make ideal offloading choice, an offloading decision model is developed to anticipate whether to offload or not, which is used by neuro-fuzzy Controller (NFC). The fundamental goal of neuro fuzzy is to choose the execution environment for the compute-intensive task undertaking either at cloud or local mobile device. QoS

parameters such as transfer speed, accessible RAM in versatile and cloud, information size and battery level and average execution time at portable and cloud are utilized as input for the decision-making process. The detailed explanation of the proposed NFC is described in following sections.

Resource Allocation in Cloud server

In cloud, the decentralized resource distribution is accomplished to perform the execution of a compute-intensive task at the minimum consumption time with the assistance of adaptive cuckoo search optimization algorithm. Request Handler in Cloud module will deal with all the solicitations from the mobile contrivances and forward to resource manager (RM) for dispensing virtual machines to execute offloading demand. RM distributes decentralized asset designation of VMs with the usage of ACSOA. Finally, the classification process is also drifted under the NFC method. Here, the classification is exactly denoted as the identical process of the images. In the DFOMAC framework, the API is designed for each and every phase of the process.

Application Programming Interface

Developed android application have three application programming interface to execute the offloading demand in the cloud environment. The primary API is created for sending and receiving output and input from application. The subsequent Interface is used for sending and getting information and yield from cloud. The motivation behind next API is to schedule the undertaking of virtual machines in the cloud environment.

2.2 Adaptive Cuckoo Search Optimization Algorithm (ACSOA)

In this research, the decentralized resource allocation process is done by the adaptive cuckoo search optimization algorithm (ACSOA) to apportion virtual machines for executing offloading demand in a proficient manner in cloud environment. The fundamental point of the proposed algorithm is to allot each task to a virtual machine and redistribute the dispensed machines with another errand to execute a more noteworthy number of requests so as to finish all the request with least time utilization. Offloading demand task comprises parameters such as RAM size expected to finish the the task in terms of MB as (t) and size of the task in terms of Million Instructions (r). Request Handler in Cloud handles the request from the mobile device with the assistance of interface and forward these parameters to resource manager which has ACSOA for mapping it on to optimal Virtual Machines to reach optimal solution It initializes the number of offloading requests, virtual machines and parameters such as image size, average execution time and available memory size as Xitr and generates the initial solution to execute the task.

Task completion time and available memory is computed for virtual machines in cloud environment. Based on that, the fitness function is resolute by minimize the execution time and maximum the available memory for task execution. Update function in ACSOA is the improvement of cuckoo search finds and rank the best arrangement and built new one for worst nest using gradient descent in order to seek optimal virtual machine for task execution. The above process continues till all the request finish its execution. This calculation builds up the essential cuckoo search without losing the quality of high-productivity search of Lévy flights which incorporates fast inquiry strategy with gradient decent (GD) algorithm to improve the intermingling rate while sustaining the astounding attributes of cuckoo search algorithm. Nonetheless, the pursuit cycle might be tedious, because of randomization conduct. With the help of Eqs. (1) and (2), the local search is calculated by using GD approach. However, with the help of Eq. (3) the global search is calculated using Levy flight.

$$x_{n+1} = x_n - \gamma_n \nabla F(x_n), \quad n > 1 \quad (1)$$

$$F(y_i) = \alpha(x_i - y_i)^2 + \beta(y_i - y_{i+1})^2 + \beta(y_i - y_{i-1})^2 \quad (2)$$

Levy flight is a random walk in which the steps are communicated in terms of the step length that are dispersed according to a heavy tailed probability distribution with the direction of steps being isotropic and random.

$$x_i^t = x_i^{t+1} + \alpha \oplus \text{Levy}(\lambda) \quad (3)$$

For the quality of the solution, the update function (9) is reinforced with the GD approach. The DFOMAC framework affects positively on minimum delay and energy saving in the server side along with the mobile devices during the decision-making for offloading applications. Algorithm symbol notations are mentioned in Table 2. The algorithm pseudocode is mentioned below.

3 Implementation and Experimental Evaluation

In this part, we report the execution and test brings about approving the performance of the DEOMAC framework, an image comparison android application was developed and installed on a mobile device. Image comparison method takes majority of the computations and if it is done at the local device, the battery will be depleted rapidly and the reaction time will be bigger. Utilizing this DEOMAC framework, performance of battery in mobile devices improved where comparison method can be offloaded to the cloud depends upon QoS parameters. Four hundred number of images stored in database for 80 objects with five images of each object. A 3/4th of the images is utilized for training and the remaining images for testing. Image

Table 2 Notations

Adaptive cuckoo search optimization algorithm	
1	Input: Number of VM and Task
2	Output: Optimal VM
3	Objective function: $F(X_i) = f(x_1, x_2 \dots x_d)$
4	Initialization:
	Initialize or generate the task (T_i) and resources (R_i)
5	$X_{t_r}^i = (x_{t_r}^1, x_{t_r}^2 \dots x_{t_r}^n)$
6	Initial solution generation $Y_{t_r}^i = (y_{t_r}^1, y_{t_r}^2 \dots y_{t_r}^n)$
7	While $t <$ Maximum iteration do
8	For $i = 1$ to Y_j
9	For $j = 1$ to N //$N \rightarrow$ No of tasks
10	Fitness function $F_i = \max(\gamma_i + 1 - \delta_i)$
11	$\delta_i = E_{ij} + R_i$ //task completion time
12	$\gamma_i = \frac{M_i}{S_i}$ //available memory evaluation
13	End for
14	End for
15	Select the best solution
16	Updation
17	Worst nests are abandoned and built a new one using a gradient descent algorithm
18	Keep the best solution
19	Rank the best solution
20	End while

comparison process is performed for input image and features of the input image were extracted for image comparison process which is utilized by neuro fuzzy. If the offloading decision is at cloud, then the server can compare the input image with images stored in its database. Then, it sends a matched result to the mobile device. Otherwise, the execution can be done at the mobile device.

Our implementation and experiments were refined to evaluate offloading achieves better execution time and reduced energy consumption of mobile devices. The smart mobile device utilized for the research is Samsung Galaxy A7 with Android 8.0 and the AccuBattery application used for evaluating battery consumption. In this research, the private cloud is purchased from the website <https://veloxitec.com> as it provides secure and resizable compute capacity in the cloud. There are about 50 VMs are created through instance with various size and run on the cloud through API. Images with different sizes such as 100 * 100, 200 * 200, 300 * 300, 400 * 400, 500 * 500, 600 * 600 were used for evaluation. The proposed framework evaluation was done with the number of images in the database by adding 400 images in the database and analyze the results of response time, energy consumption for cloud, and mobile execution for various methods. Figures 2, 3, 4 and 5 represents the response



Fig. 2 Analysis of response time using various methods a cloud b mobile

time for cloud and mobile, energy consumption for mobile and cloud, classification accuracy, and offloading prediction for 400 images NFC achieves minimum response time and battery consumption in cloud and mobile than traditional methods. It also achieves maximum prediction and classification accuracy when compared to other methods.

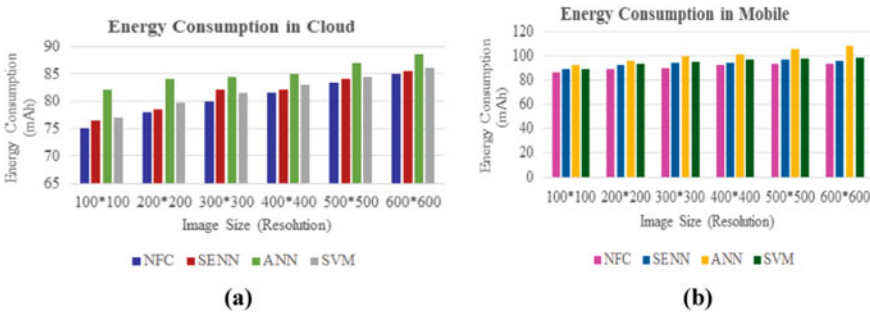


Fig. 3 Analysis of energy consumption using various methods a cloud b mobile

Fig. 4 Classification accuracy

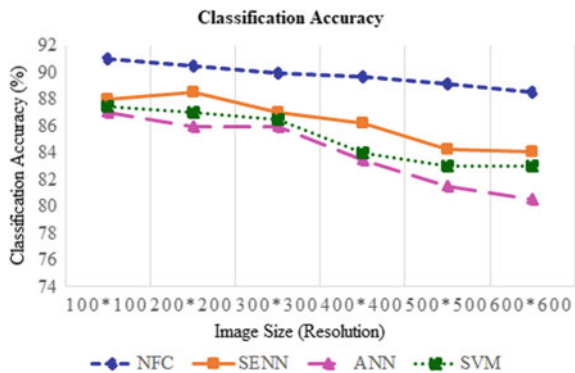


Fig. 5 Prediction accuracy

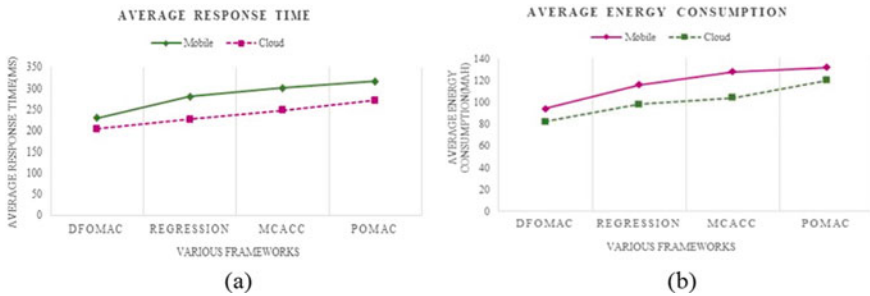
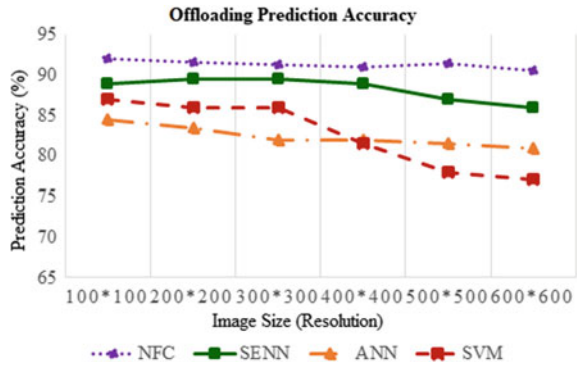


Fig. 6 Analysis of various frameworks a average response time b average energy consumption

For second scenario, the performance evaluation of DFOMAC framework is compared with various existing frameworks for response time and energy consumption at mobile and cloud. Figure 6a and b shows the analysis of average response time and average energy consumption of DFOMAC framework with various existing frameworks.

4 Conclusion

Nevertheless, due to highly dynamic topologies and frequent link connections, emerges a need for a dynamic framework for offloading, which is one of the most critical issues that need to be considered. To address this issue, the proposed work presented dynamic decentralized offloading framework with improved QoS factors considered for predicting the execution environment for the compute-intensive task either at mobile or cloud by NFC. Decentralized resource allocation was carried out in this research with the utilization of ACSOA. The performance analyses of M-POMAC and Extrade with MFCMC framework are analysed using the simulation environment. After that, the DFOMAC framework is analysed in a real-time

environment to meet the exact requirements of the offloading process. The proposed framework is implemented in the real-time environment using Android platform and battery performances of mobile devices are evaluated using AccuBattery application. The proposed framework was evaluated by varying the number of images in the database which is compared with the existing method and shows better energy capability and response time as compared to existing methods. After that, the proposed framework is compared with existing frameworks which shows that execution time, and the energy consumption is better than the traditional frameworks. The main outcomes of this research are providing an energy-efficient offloading framework with improved QoS factors which includes decentralized resource allocation in cloud. In the future, would concentrate on to stretch out this research further to take the situations in which numerous cloud servers are reachable for on mobile device.

References

1. <https://economictimes.indiatimes.com/tech/internet/internet-users-in-india-to-rise-by-40-smartphones-to-double-by-2023-mckinsey>
2. <https://www.ericsson.com/en/mobility-report/reports/june-2020/mobile-subscriptions-outlook>
3. Wang Y, Chen IR, Wang DC (2015) A survey of mobile cloud computing applications: perspectives and challenges. *J Wirel Pers Commun* 1607–1623
4. <https://www.bankmycell.com/blog/how-many-phones-are-in-the-world>
5. <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide>
6. Liu L, Du Y, Fan Q, Zhang W (2019) A survey on computation offloading in the mobile cloud computing environment. *Int J Comput Appl Technol* 59(2)
7. Shruthi BM, Pruthvi PR, Kavana MD (2017) Mobile cloud computation: issues and challenges. *Int J Recent Trends Eng Technol* 3(4) ISSN: 2455-1457
8. Sareen B, Sharma S, Arora M (2014) Mobile cloud computing security as a service using android. *Int J Comput Appl* 99(17):0975–8887
9. Akherfi K, Gerndt M, Harroud H (2018) Mobile cloud computing for computation offloading: issues and challenges. *Applied computing and informatics* 14(1):1–16
10. Kim Y, Lee H-W, Chong S (2019) Mobile computation offloading for application throughput fairness and energy efficiency. *IEEE Trans Wirel Commun* 18(1):3–19
11. Wang Y, Sheng M, Wang X, Wang L, Li J (2015) Mobile-edge computing: partial computation offloading using dynamic voltage scaling. *J Latex Class Files* 14(8)
12. Cardellini V, De Nitto Persone V, Di Valerio V, Facchinei F, Grassi V, Lo Presti F, Piccialli V (2015) A game-theoretic approach to computation offloading in mobile cloud computing. *Math Program* ISSN 0025-5610
13. WU H (2018) Multi-objective decision-making for mobile cloud offloading: a survey. *IEEE Access* 6:3962–3976
14. Hassan MA, Bhattarai K, Wei Q, Chen S (2014) POMAC: Properly offloading mobile applications to clouds
15. Cuervo E, Balasubramanian A, Cho DK, Wolman A, Saroiu S, Chandra R, Bahl P (2010) MAUI: making smartphones last longer with code offload. In: *Proceedings of the 8th international conference on mobile systems, applications, and services*. ACM, pp. 49–62
16. Chun BG, Sunghwan I, Petros M, Mayur N, Ashwin P (2011) CloneCloud: elastic execution between mobile device and cloud. In: *6th Conference on computer systems (EuroSys)*, pp. 301–314

17. Kosta S, Aucinas A, Hui P, Mortier R, Zhang X (2012) Thinkair: dynamic resource allocation and parallel execution in the cloud for mobile code offloading. In: Infocom, 2012 proceedings IEEE, IEEE, pp. 945–953

A Novel Approach for Identification of Healthy and Unhealthy Leaves Using Scale Invariant Feature Transform and Shading Histogram-PCA Techniques



K. S. Shashidhara , H. Girish , M. C. Parameshwara ,
B. Karunakara Rai , and Veerendra Dakulagi 

1 Introduction

This because of varieties in climatic conditions, the cultivation crops and different pieces of the plants, for example, roots, stem, leaf, and seeds are assaulted by the infections [1]. Additionally, the agriculture crop sicknesses spread at a quicker rate contrasted with the other class of yields. This will bring about low harvest yield and cause budgetary misfortune to the ranchers. The plant's and the yield's wellbeing can be kept up by applying the fundamental medication on the plants, yet this in a roundabout way influences the soundness of the shoppers [2]. Further, the persistent increment in populace requests more harvests. Along these lines, vital advances must be taken to create more and solid yields.

The cultivation crops comprise of foods grown from the ground. As indicated by the Indian government, in the year 2015, around 500 million individuals have a place with white collar class and underneath neediness line [3]. For these classifications of the individuals, notwithstanding the food grains, vegetables are of higher need and

K. S. Shashidhara · B. K. Rai
Nitte Meenakshi Institute of Technology, Bengaluru, Karnataka, India
e-mail: sks.nmit@gmail.com

H. Girish
Cambridge Institute of Technology, Bengaluru, Karnataka, India
e-mail: girish.ece@citech.edu.in

M. C. Parameshwara
Vemana Institute of Technology, Bengaluru, Karnataka, India
e-mail: pmcvit@gmail.com

V. Dakulagi (✉)
Guru Nanak Dev Engineering College, Bidar, Karnataka, India
e-mail: veerendra@ieee.org

structures the essential and significant piece of the dinner contrasted with the natural products.

The most every now and again utilized vegetables in the everyday life are given in Table 1 [4, 5]. Experts (experienced farmers) have proposed that, contrasted with the roots and stem portions of the plants, leaves are progressively defenseless for infection assaults. They have recommended that distinguishing the wellbeing state of the plant from the leaves is significant and just as simple contrasted with different pieces of the plant. Subsequently, leaf is considered in this work for distinguishing the wellbeing state of the plants. Specialists have given the quantity of days taken by the infection to assault the plants totally, and this information is summed up in Table 1. Experts have proposed that more the water substance of the stem, more is the plant inclined to ailments. The information given by the specialists on the water substance of the stem is summed up in Table 2. From the table, it tends to be seen that the accompanying three vegetables are having high water content, in particular: tomato, potato, and beans. Potato is a one-time crop for each plant, as the yield is gotten subsequent to uncovering the plant. The life expectancy of beans plant is around a quarter of a year and in this life expectancy it yields four to multiple times. The life expectancy of tomato plant is around a half year and in this life expectancy it yields seven to multiple times [6, 7]. Additionally, tomato has more medical advantages contrasted with potato and beans [7–9]. A portion of the medical advantages of devouring tomato is as per the following: ensures vision and degenerative eye ailment, lessens cardiovascular ailments, decreases the danger of prostate malignant growth and bosom disease, forestalls kidney and nerve bladder stones, diminishes the danger of blood clump, expands fat consuming limit, forestalls stroke, reestablishes biochemical parity in diabetics, improves processing and forestalls obstruction, gives solid and gleaming skin, and supports the hairs [10, 11]. Due to these medical advantages and high return from the tomato plant, we have considered tomato in our work [12].

2 Mathematical Framework

One of the different difficulties looked by the ranchers is to locate a specific plant's leaves that has been ailing or not, out of enormous number of tomato plants developed in the homestead. Accepting that a decent framework exists with the end goal of picture securing, this work robotizes the way toward distinguishing whether a specific plant (leaf) is solid or not. Here, two strategies are proposed for the ID of solid or unfortunate tomato leaves. First strategy depends on the scale invariant feature transform (SIFT), and the subsequent technique depends on the shading histogram and the principal component analysis (PCA). A short audit of the SIFT, the shading histogram and the PCA is as per the following.

2.1 SIFT

The SIFT removes the fixed nearby element focuses from an info picture that are invariant to fundamental picture changes, for example, scale and revolution. The info picture is exposed to Gaussian channels with various cover size utilizing various estimations of σ , to get a lot of pictures with various sizes of blurring. The outrageous focuses are extricated by looking at each purpose of the obscured picture with the focuses in a similar space and the focuses in the local space. The Gaussian scale change administrator utilized by Lowe was the distinction of Gaussian (DoG), and the scale space is Gaussian contrast scale space is $\{G(x; y; \sigma)\}$, Lowe demonstrated that the point extricated by this administrator is invariant in scale change of unique picture $I(x, y)$ (1 cm).

2.2 Color Histogram

Every pixel in a RGB shading picture is a blend of the parts from red plane, green plane, and blue plane. The shading histogram is a graphical portrayal of shading dispersion in a picture. The shading histogram is acquired by plotting the histogram of each plane (red, green, and blue) independently. Utilizing, the histogram data, the shading picture can be evened out in two different ways to upgrade the picture quality, if fundamental. In first technique, the shading picture evening out is performed by applying dark scale histogram leveling on every one of the plane independently, and pressing them back together to get the balanced shading picture.

In the subsequent technique, the RGB picture is changed into YIQ qualities and afterward, the dim scale histogram leveling is applied on the Y channel, leaving the I and Q channels unmodified. The adjusted Y and the unmodified I and Q are connected and changed back to RGB to acquire the balanced shading picture.

2.3 PCA

The PCA removes important highlights from the informational collections and lessens the information from higher dimensional space to the lower dimensional space. It is one of the most much of the time utilized procedure for picture acknowledgment applications [13]. Let be the quantity of preparing pictures.

3 The Proposed Algorithm

To distinguish whether a tomato leaf is solid or not, two of the calculations are proposed in this paper. In both the calculations, the execution is done in two stages. The main stage is the database creation stage, where in the highlights are removed from the info preparing set of tomato leaf pictures. The preparation set comprises of both sound and unfortunate tomato leaves.

The subsequent stage is the recognizable proof stage, where in the highlights are separated from the inquiry picture and contrasted and the highlights put away in the database. On the off chance that there is a match over some specific limit, at that point, the pertinent metadata (sound or unfortunate) is created as the yield. One of the calculation depends on the SIFT and the other calculation depends on the shading histogram and the PCA. Both the calculations are clarified in the accompanying subsections.

3.1 Proposed Algorithm Based on SIFT

The square graph of the proposed calculation dependent on the SIFT is appeared in Fig. 1. From the square outline, it tends to be seen that the arrangement of steps followed in the database creation stage and the recognizable proof stage are same. The succession of steps is as per the following: input picture (preparing picture or the question picture), standardization, and highlight extraction.

In the database creation stage, the info preparing picture is standardized, by changing the RGB picture in to the PGM arrangement and afterward resizing to a predefined size. After standardization, the component is separated utilizing the SIFT and the element is put away in the database alongside the metadata of the relating input preparing picture. The metadata is only whether the tomato leaf is solid or not. At the end of the day, in the event that the information tomato leaf is undesirable, at that point the relating removed highlights are put away under the unfortunate class and on the off chance that the info tomato leaf is sound; at that point the comparing extricated highlights are put away under the solid class. In the ID stage, a question picture whose wellbeing condition is yet to be recognized by the calculation is given as information. Following the comparative method in the database creation stage, the element is extricated. This removed component is contrasted and the highlights put away in the database and the applicable metadata is created as the yield of the calculation.

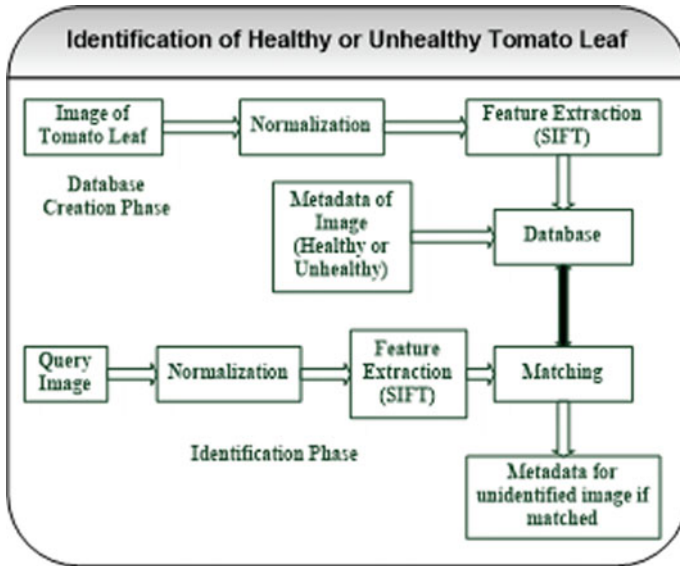


Fig. 1 Block diagram of proposed algorithm based on SIFT

3.2 Experimental Results

The database contains 80 leaf pictures of tomato plants caught utilizing the Sony advanced camera. The highlights of the computerized camera are as per the following: spatial goal of 18.2 uber pixels, advanced zoom of 120X, and optical zoom of 8X. The inexact good ways from the outside piece of the camera focal point and the leaf was 17–20 cm. Each picture was caught by keeping a sheet of dark paper under the leaf and in this way making the foundation dark. This arrangement improves the exhibition of the calculation in recognizing and characterizing the leaf as solid or not. Eighty pictures were caught utilizing the above arrangement conditions. Out of these 80 pictures, 28 pictures are sound and 52 pictures are unfortunate. Further, out of those 52 undesirable pictures, 37 pictures are considered for preparing the framework, and 12 pictures are considered as test pictures. These 37 pictures are named type 1 and type 2. Under sort 1 there are 30 pictures and 7 pictures are of type 2. The leaves under sort 1 considered as ailment assault in early phase, under Type 2 considered as the illness assault last stage. Under sort 1 by apply medications the infection can be fix without any problem. This malady will not impact encompassing plants. Under sort 2 malady is in definite stage, by applying medications it is absurd to expect to fix 100%, and it will influence encompassing plants.

This paper incorporates the proposed system which is executed utilizing SIFT calculation and shading histogram-PCA. The trial results are as per Figs. 2 and 3

The removed key points of leaf pictures utilizing SIFT calculation are to distinguish the specific evaluation.

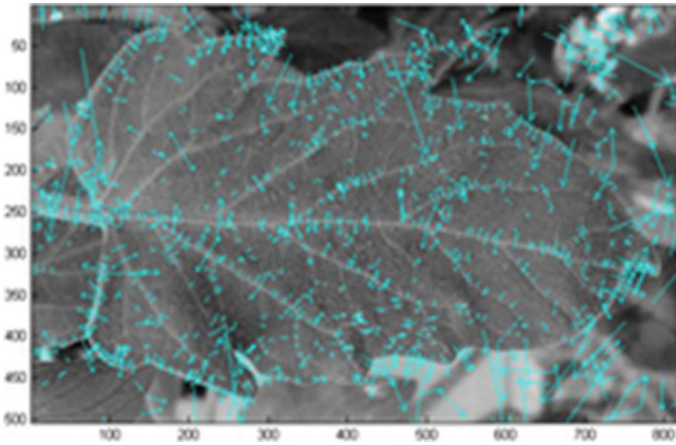


Fig. 2 Key points found in healthy leaf

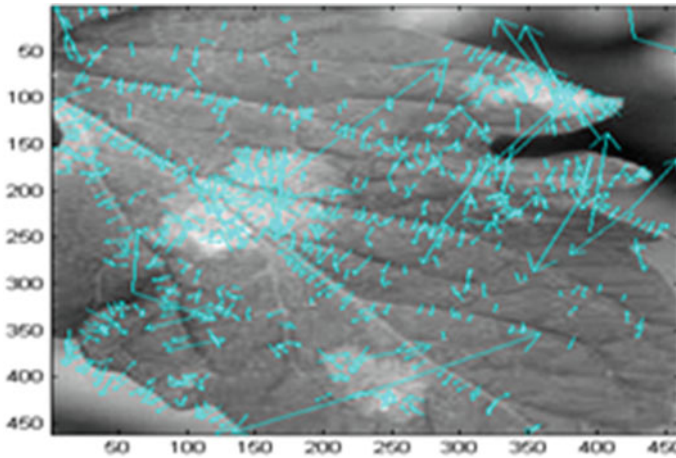


Fig. 3 Key points found in unhealthy leaf

4 Conclusion

The ranchers are confronting colossal misfortunes because of the leaf maladies. Leaf maladies are caused because of the lopsided ecological conditions. Because of this the whole harvest yield has been decreased, cost of the reaped yield will be sold at significant expenses. The purchasers are in a roundabout way influenced by paying gigantic total. Hence, insurance must be taken so as to forestall this ailment by structuring a framework which screens the continuous information of the harvests.

The proposed paper utilizes SIFT and histogram-PCA calculation, which is utilized to distinguish the leaf malady, checking the leaf is sound or not. The SIFT

calculation is utilized to discover the highlights of each dataset and analyze the question picture. In this dataset, the information picture and question picture key points are considered for coordinating. On the off chance that the key points utilized is not coordinated, at that point it is considered as unfortunate.

References

1. Explore Cornell-Home Gardening-Minimizing Diseases in Vegetable Gardens, Gardening.cornell.edu, 2016. Available: <http://www.gardening.cornell.edu/homegardening/scene7af3.html>
2. Medicines from Plants, Webcache.googleusercontent.com, 2016. http://webcache.googleusercontent.com/search?q=cache: http://www.rainforesteducation.com/medicines/PlantMedicines/rfmedicines.htm&gws_rd=cr&ei=Zqw6V5LYI4rRvgSvrq7gCw
3. Agriculture in India, Wikipedia, 2016. Available: https://en.wikipedia.org/wiki/Agriculture_in_India
4. The Most Popular Vegetables, Ranker, 2016. [Online]. Available: <http://www.ranker.com/crowdranked-list/the-most-delicious-vegetables-v1>
5. The most consumed vegetables in the USA, Supplement SOS, 2016. Available: <http://supplementsos.com/nutrition-stats/most-consumed-foods/most-eaten-vegetables-usa>
6. 3 Vegetable Seeds That Have a Large Yield Per Seed. Gardening | Pioneer Thinking, 2015. Available: <http://pioneerthinking.com/gardening/3-vegetable-seeds-that-have-a-large-yield-per-seed>
7. AZ Master Gardener Manual: Potatoes, Ag.arizona.edu, 2016. [Online]. Available: <https://ag.arizona.edu/pubs/garden/mg/vegetable/potatoes.html>.
8. Health benefits of white beans. Healwithfood.org, 2016. Available: <http://www.healwithfood.org/health-benefits/white-beans-navy.php>
9. 10 Healthy Reasons to Dig Into Red Potatoes, Blackgoldfarms.com, 2016. Available: <http://blackgoldfarms.com/community/the-dirt/2015/april/26/10-healthy-reasons-to-dig-into-red-potatoes/>
10. 10 Reasons Why You Should Be Eating More Tomatoes Florida Tomato Committee. Floridatomatoes.org, 2016. Available: <http://www.floridatomatoes.org/news-events/10-reasons-why-you-should-be-eating-more-tomatoes/>
11. 20 Amazing health benefits of tomatoes that should make them a daily staple in your diet. Health Impact News, 2014. Available: <http://healthimpactnews.com/2014/20-amazing-health-benefits-of-tomatoes-that-should-make-them-a-daily-staple-in-your-diet>
12. Mateljan G (2010) 9-oxo-10(E), 12(E)-octadecadienoic acid derived from tomato is a potent PPAR α agonist to decrease triglyceride accumulation in mouse primary hepatocytes. Mol Nutr Food Res 55(4):585–593
13. Implementing the Scale Invariant Feature Transform (SIFT) Method (2016) Available: [http://Implementing the Scale Invariant Feature Transform \(SIFT\) Method](http://Implementing the Scale Invariant Feature Transform (SIFT) Method)

A Comprehensive Review on the Issue of Class Imbalance in Predictive Modelling



Prashanth P. Wagle and M. V. Manoj Kumar

1 Introduction

Many machine learning classifiers are trained with the assumption that the distribution of all the classes in the training set is equal. However, in most real-world applications such as network intrusion detection, credit card fraud detection and health screening [1], there is a dearth of classes which matter the most. For example, in credit card fraud detection, the volume of data pertaining to legitimate transaction is humongous when compared to data pertaining to frauds.

The situation where the ratio of instances in a class is uneven is termed as class imbalance. Class imbalance leads to many challenges in training the classifiers. Class imbalance occurs in data which has only two classes (binary class imbalance) and in data which has multiple classes (multiclass imbalance). The range of methods used to solve the problem is categorized as Data Level, Algorithmic Level and Hybrid Level; the taxonomy of the methods has been given in Fig. 1.

The contribution of this paper is as follows.

- Comprehensive categorization of various families and methods to solve the class imbalance problem.
- Discussion of techniques and evaluation strategies proposed in recent research works.
- To elaborate on the category of methods which are vast, such as cost-sensitive methods and ensemble methods, for solving the class imbalance problem.

P. P. Wagle (✉)

Ethnus Consultancy Services Pvt. Ltd, Bengaluru, India

e-mail: prashanthwagle360@gmail.com

M. V. Manoj Kumar

Nitte Meenakshi Institute of Technology, Yelahanka, Bengaluru 560064, India

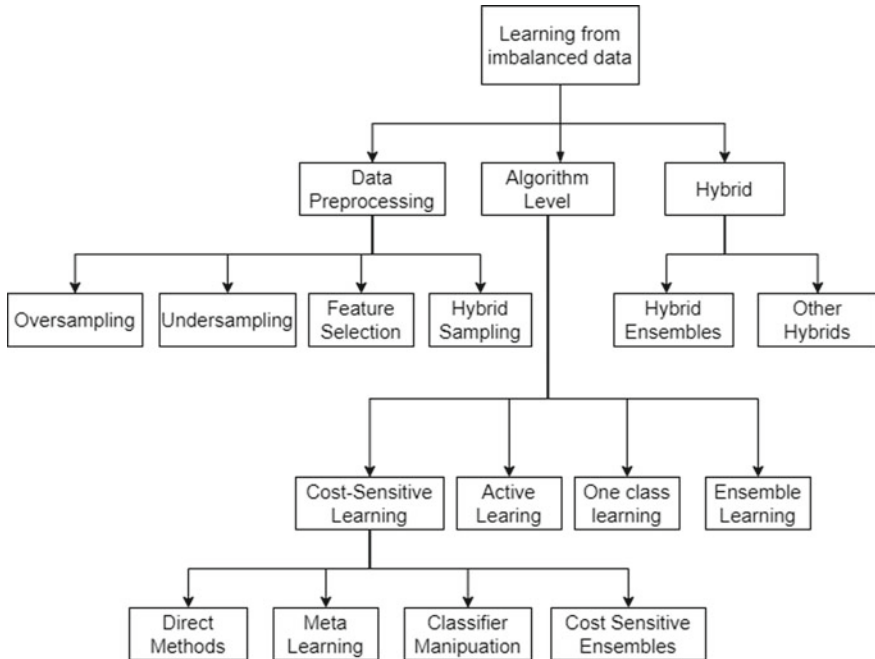


Fig. 1 A graphical overview of techniques to handle class imbalance in binary classification problems

The motivation of this study is to review the existing techniques to handle the imbalance in the distribution of the classes. The novelty of this research lies in presentation of the latest techniques of handling class imbalance like MAHAKIL: diversity-based oversampling approach [2], clustering-based instance selection (CBIS) [3], stacking ensemble learning [4], etc., and evaluation strategies like Matthews correlation coefficient (MCC) [5].

This paper provides a comprehensive overview of the techniques used to handle the effects of skewed class distribution. We have reviewed the standard methods and latest research works which intended to solve the class imbalance problem, also known as the skewed class distribution in machine learning. Section 2 deals with the additional problems that potentially occur along with the problem of class imbalance. Section 3 deals with the techniques published by various resources to deal with the problem. Section 4 provides some examples of domain-specific class imbalance problems and the techniques used to handle them. Section 5 deals with the evaluation metrics to compare the techniques.

We present some research issues for the future and conclude the paper in Sect. 6.

2 Challenges Occurring with Class Imbalance in Predictive Modelling

The performance of classifiers and evaluation metrics due to imbalance classifier depends on various factors, which occur when the training dataset has a skewed class distribution. Several observations have been made in this field [6–11], and they are described below.

2.1 Class Imbalance Ratio

In a binary classification scenario, the minority to majority class ratio can be nearly equal or can be at the ratio of 1:2, 1:5, 1:10,000 and so on. Several research papers discuss this issues, and certain classification algorithms have been reported to be better suited for a smaller skew and certain algorithms for a larger skew. However, the selection of algorithms also depends on other factors (Table 1).

In problems with a larger skew, the algorithms try to achieve higher accuracy by incorrectly classifying all test examples as belonging to the majority class. This scenario may result in non-detection of the monitory class, which in most cases is more important than the majority classes: a problem which is omnipresent in all domains presented in Table 3.

It is also noted that alternative methods of measuring classifiers’ performance such as receiver operating characteristic (ROC) curves are used as opposed to other scalar methods. Many studies have been carried out studying the effect of class imbalance ratio on the performance of the classifiers, where some studies say that an equal ratio may not yield the best performance. Hence, there may be other factors apart from just the ratio.

Table 1 A cost matrix for a binary classification problem where $C(i, i)$ is usually 0. Here, $C(0, 1)$ represents cost of a false negative and $C(1,0)$ is cost of a false positive [52]. The family of techniques of cost-sensitive algorithms use this matrix

	Actual positive	Actual negative
Predicted positive	$C(0,0)$, true positive (TP)	$C(0, 1)$, false positive (FP)
Predicted negative	$C(1,0)$, false negative (FN)	$C(1, 1)$, true negative (TN)

2.2 *Internal Clustering*

Internal clustering is also referred to as disjuncts in the dataset. The imbalance present within the classes is sometimes ignored, which results in erroneous decision boundaries. These imbalances can be imagined as clusters with a cluster. The experiment described in [11] suggests that the problem is not directly caused by class imbalances, but rather, that class imbalances may yield small disjuncts which, in turn, will cause degradation. The authors of [8] argue that in order to improve classifier performance, it is more useful to focus on the issues caused by small disjuncts' than to focus on class imbalance.

2.3 *Lack of Data*

In addition to imbalanced class ratio in the training dataset, another challenge arises where there is not much information about the minority class. More data is usually preferred as it aids in better modelling. Particularly, lack of a proper data would lead to misclassification as it affects the accuracy of the decision boundary in a classification scenario.

2.4 *Class Overlapping*

The degree of overlapping between various classes presents several challenges for the classifier to effectively separate the different classes by forming a decision boundary. Japkowicz [6] presents class complexity as an important factor determining the performance. Especially, the work in [10] develops a systematic study aiming to question whether class imbalances are truly to blame for the loss of performance of learning systems or whether class imbalances are not a problem by themselves. Hence, in this review, we will not be considering class overlapping as a problem caused by class imbalance.

In the next section, we shall present the existing methods to prevent the problem of class imbalance, along with the other issues discussed above.

3 Techniques to Handle Imbalanced Classification

There have been various techniques developed to handle class imbalance in binary classification problems. We have opted for binary classification problems for presenting the methods for ease of understanding. These techniques can be grouped into data preprocessing methods, algorithmic methods and hybrid methods [1].

3.1 Data Preprocessing Techniques

Data preprocessing techniques or methods are concerned with balancing the class distribution of the instances of both classes in the dataset. Balancing the class distribution can be done either by including more instances of the minority class (oversampling) or lesser instances of the majority class (undersampling) [12]. Both oversampling and undersampling can have drawbacks. Undersampling can discard potentially useful data, while oversampling instances may cause overfitting or a combination of both [13]. Hence, we have a third category of methods, which we call Hybrid Sampling, which attempts to overcome the drawbacks previously described [14]. Additionally, there are other family of techniques which fall under data preprocessing methods known as the feature selection methods which are effective for high dimensional data exhibiting class imbalance.

Undersampling Random undersampling [15] is a naive non-heuristic method of randomly removing the instances of the majority class in order to balance the class distribution.

Tomek links (TL) [16] is an undersampling method which can also be used as a data cleaning method. It is a modification of condensed nearest neighbour rule (CNN) [17] which uses rules to determine if a pair of instances E_i and E_j belonging to two different classes form a Tomek link (find math from wiki). TL removes unnecessary class overlap by removing the majority of class links until all minimally distanced closest neighbour pairs are of the same class. The procedure involved in Tomek links is outlined in Fig. 2.

One-Sided Selection (OSS) [19] is an undersampling method which is a combination of applying Tomek links followed by CNN. TL removes noisy and borderline, whereas CNN eliminates the majority class examples that are distant from the decision border.

The Edited Nearest neighbour (ENN) [20] rule removes any example whose class label differs from the class of at least two of its three nearest neighbours. The Neighbourhood Cleaning Rule [21] is an undersampling method, which is a modification of ENN. If an instance belongs to the majority class and its labels differ from the class of at least two of its neighbours, then the instance is removed. However, in the same scenario, if the instance belongs to the minority class and is surrounded by majority classes, then the latter are removed [18].

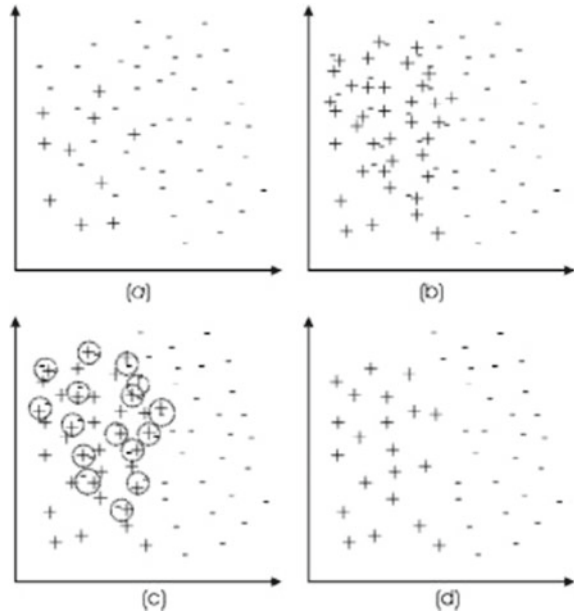
Another family of methods known as the Near-Miss family [22] perform undersampling of instances in the majority class based on their vicinity to other instances in the training set.

The work of Drummond and Holte [23] has shown that undersampling produces much better results when compared to oversampling using cost curves as an evaluation metric.

Several clustering-based undersampling have been implemented in the past, which have yielded good results.

The work in [24] proposes two novel clustering-based strategies which have been applied to ensemble algorithms [25]. In the first strategy, ' k ' clusters were generated

Fig. 2 Balancing a dataset: original dataset (a); oversampled dataset (b); Tomek link identification (c); and borderline and noise example removal (d) [18]



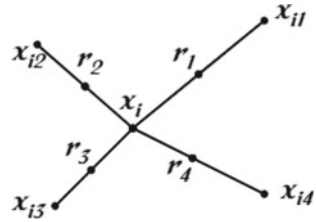
using the k-means clustering algorithm, where ‘ k ’ was equal to the numbers of samples of the minority class. The majority class examples are replaced by the cluster centres, as they are the representatives of the replaced examples. The other strategy involved selecting the nearest neighbour of each cluster centre since it was a real data sample to replace the centroids. Studies were made using datasets with varying levels of imbalance, and the studies showed that the second strategy with ensemble methods was much more preferable for datasets with a large imbalance ratio. The most common method used in the clustering procedure in the algorithms was k-means clustering. However, certain limitations of k-means clustering include determination of the number of clusters before running the algorithm and many others [26]. Hence, [3] recommends a Clustering and Instance Selection method (CBIS) based on the clustering algorithm of affinity propagation [27]. As the name says, CBIS has two components—Clustering and Instance Selection.¹ An interesting fact is that both [3, 28] conclude that clustering algorithm used with Multi-layer Perceptron (MLP) classifier yields the best results, especially for small-scale datasets [28].

In the work [28], techniques based on clustering have been proposed where backpropagation neural networks are used to solve the skewed class distribution.

Oversampling Random oversampling (ROS) is the oversampling equivalent of random undersampling. It is a naive, non-heuristic method of balancing the class distribution by replication of the majority class examples.

¹ Instance selection is used in the preprocessing step of training an ML model, where it retains characteristics of the dataset while minimizing the bulk of the dataset.

Fig. 3 Example of an instance synthesized using SMOTE as described in [29]



ROS has two major shortcomings. Firstly, it may cause overfitting since the minority class instances are replicated [30]. Replications may also cause a specific decision region of the minority class, as observed by [29]. Hence, a new proposal was made in [29] with a method Synthetic Minority Oversampling Technique (SMOTE) which adds synthetic instances for the training procedure. Oversampling of the minority class is done by introducing synthetic examples along the line segments joining the existing minority class examples. Figure 3 shows an example of SMOTE [23]. x_i is a minority class instance which is selected. Four nearest neighbours x_{i1} – x_{i4} are chosen from the training set, and instances r_1 through r_4 are the new synthetic examples generated.

Borderline-SMOTE is a popular extension to SMOTE which involves selecting the borderline instances which are more likely to be misclassified [31]. Adaptive Synthetic (ADASYN) sampling [32] is an oversampling procedure which is an improvement over SMOTE wherein it not only generates synthetic examples adaptively, but also shifts the decision boundary in a bid to emphasize on learning the difficult examples, thereby improving the performance.

Combining Undersampling and Oversampling The work in [31] shows that SMOTE combined with undersampling methods performs better than just vanilla undersampling, based on the former dominating the latter in receiver operating characteristics (ROC) space. However, [33] enlists the disadvantages of SMOTE:- Noise in samples may produce synthetic samples and may lead to blurring of boundaries between the classes and lack of diversity in the generated samples. Thus, Liang et al. [33] proposes an improvement over SMOTE named LR-SMOTE, which claims to address the shortcomings of SMOTE. Evaluation of the experimental results depicts significant reduction in noise and denotes the stability of the algorithm.

Hybrid Sampling To account for the limitations of oversampling and undersampling, ensemble and hybrid methods have been proposed and have shown to address the problems faced by the techniques discussed previously.

A critical issue that is not addressed in all the methods presented until now is diversity.

There are certain relatively recent publications which address the issue of lack of diversity² in the oversampled majority class after the sampling process, especially if the datasets consist of many sub-clusters, Fig. 1. MAHAKIL [2] is one such

² Here, diversity refers to the diversity of the instances of the training data.

technique based on the chromosomal theory of inheritance³ which addresses this diversity. The work in [2] also denotes the problem with citing the examples of SMOTE, random oversampling (ROS), etc., where using SMOTE leads to an issue of high false positives [35] by the classifiers. Barua et al. [36] and Menzies et al. [37] confirmed the same, and [37] suggests that high recall and low false positives of the prediction models suggest the stability of the sampling techniques. Bennin et al. [2] presents MAHAKIL which achieves both high recall and low false positives.

The category of Hybrid Sampling in Fig. 1 is sampling methodologies which combine multiple sampling techniques. Batista et al. [18] presents a few of such techniques, namely CNN + Tomek links, SMOTE + Tomek links and SMOTE + ENN. They show that SMOTE + Tomek links and SMOTE + ENN work well with datasets, with a few positive examples. Conventionally, positive examples are considered to be the minority class throughout the paper. They also demonstrate that ROS, which is computationally less expensive compared to most other techniques, works very well for datasets with lower ratio of class imbalance.

Feature Selection The work in [38] showed that for high dimensional imbalanced datasets, resampling methods are ineffective. They use feature selection methods as an alternative to resampling and test various filter method such as chi-square test (CHI), Information Gain (IG), Pearson Correlation Coefficient (PCC), Feature Assessment by Sliding Thresholds (FAST), Feature Assessment by Sliding Thresholds (FAIR) and Signal-to-Noise Correlation Coefficient (S2N) [38].

The metrics of PRC and ROC curves were used under several classifiers, and through experimental evaluation of these techniques over imbalanced datasets with high dimensionality, they found it to be highly beneficial when compared to resampling methods, regardless of the classifier used. In contrast to the results of [39], the possible explanation they provide is that resampling methods (up to 10 k records) is not effective as applying it for larger datasets. [39, 40] reinforce this explanation, in which they suggest that the best choice for a feature selection technique is dependent on the number of data points of each class of the training set and the number of features desired. In case of small datasets with higher features, feature selection works well, but further investigation is required to compare feature selection and resampling methods on larger dataset.

3.2 Algorithmic Methods

Unlike the previous methods, algorithmic methods work on modifying the learning algorithms to avoid the bias towards the majority class. There are multiple methods of doing the same, and they are categorized under the methods shown in Fig. 1. Cost-sensitive learning has gained attention from the machine learning community

³ The chromosomal theory of inheritance, proposed by Sutton and Boveri, states that chromosomes are the vehicles of genetic heredity [34] where both parents contribute chromosome pair of the offspring.

[41]. Cost-sensitive algorithms consider costs of misclassification [42] to achieve a higher recall value. Misclassification costs are assigned to false predictions, and prediction is made by taking these costs into account. At a high level, cost-sensitive learning can be structured into two categories: meta-learning and implicit method [43]. The former is a method for converting existing cost-insensitive classifiers into cost-sensitive ones, whereas the latter is a way for constructing classifiers that are implicitly cost-sensitive. A third category can be considered, which is modifications of the cost function of existing algorithms to make them cost-sensitive [44].

Cost-sensitive learning methods Algorithms which are inherently developed for cost-sensitive learning are rare. Ling and Sheng [43] enlists two algorithms: Inexpensive Classification with Expensive Tests (ICET) [45] and CSTree [46] which inherently incorporate misclassification costs. Drummond et al. [47] notes that out of all split-criteria, accuracy exhibited the highest cost sensitivity.

The meta-learning methods are more common when compared to implicitly cost-sensitive methods. It can be categorized into sampling and thresholding. Unlike the resampling presented in the data preprocessing methods, which strives to balance the class distribution, cost-sensitive sampling focuses on changing the distribution according to the costs of misclassification [48]. On the other hand, thresholding selects a threshold from the training instances according to the misclassification costs [49]. MetaCost, which is based on bagging, is one such algorithm for rendering the classifiers cost-sensitive [44]. AdaCost is another such algorithm, which is a variation of AdaBoost with the purpose to reduce the cumulative misclassification cost more than AdaBoost [50].

Apart from these methods, classifiers like k-nearest neighbours, Artificial Neural Networks and Support Vector Machines [44] can be internally manipulated so that they become cost-sensitive.

Cost-sensitive learning involves techniques which strive to minimize a loss function associated with a dataset, since most problems do not consider uniform costs for misclassifications [10]. Cost matrix is crucial in determining the costs and is similar to confusion matrices. It may be prepared by an expert, or it can be estimated using the training data [51]. It is a matrix that assigns a cost to each cell in the confusion matrix.

Classification of Cost-Sensitive methods We have subcategorized the cost-sensitive algorithmic technique into four groups, as shown in the figure. Direct approach, where we introduce the cost directly into the classifier and meta-learning approaches, which are predicated on altering either the training data or the classifier's output, namely preprocessing and post-processing, respectively [51]. The standard machine learning algorithms such as Decision Trees and Support Vector Machines (SVM) can be made cost-sensitive using either one of the approaches. There are other ensemble techniques of cost-sensitive algorithms which serve as "wrapper" methods over standard classifiers. They fall into the category of cost-sensitive ensembles. MetaCost [50] is one such technique, which is a meta-learning technique which modifies the training data using ensemble models. AdaCost which is briefed in [49] is a cost-sensitive variant of AdaBoost.

Additionally, classifiers can be manipulated internally so that the classification algorithms can be made cost-sensitive. Particularly, certain findings have shown that certain classification algorithms are less sensitive to class imbalance [8]. Hence, internal manipulation of algorithms is used in cases where there is a need to balance between sensitivity and specificity, as demonstrated in [53].

One-class learning and Active learning One-class learning [54] is a category of learning one of the classes which would be well characterized by the instances in the training data, and there will be little or no instances of the other class. Since there is an overrepresentation of a class in imbalanced data with two classes, active learning can be used. By actively selecting the useful data points, active learning is utilized to build a high-performance classifier while minimizing the size of the training dataset to a minimum. The work in [55] elaborates more on the role of active learning in imbalanced datasets. Bellinger et al. [56] has shown using real-world datasets that when imbalance increases, the performance of binary classifiers decreases and that of the one-class classifiers remain the same and this is pronounced at higher imbalance ratios. An example is Kernel Autoassociators which are a type of autoassociator⁴ in which one takes a kernel feature space as the nonlinear manifold and places emphasis on the reconstruction of input patterns from the kernel feature space [58]. One-class SVM learners are popular in text and document classification [59], but most existing machine learning (ML) algorithms cannot learn from a single class, and hence, one-class learning is less popular [60, 61].

Ensemble learning The category of ensemble learning presents us with techniques built on existing ensemble techniques, namely bagging, boosting and stacking. The work in [25] proposes a taxonomy to categorize the various ensemble techniques intended to negate the effects of a skewed class distribution. Figure 4 depicts the taxonomy provided by [23]. Table 2 gives basic definitions of bagging and boosting [62], which are the techniques from which other ensemble techniques are built upon.

Stacking involves learning several heterogeneous weak learners and combines the base models using a metamodel. A stacking-based ensemble model using the concepts of stacking is discussed in [4].

The category of cost-sensitive ensembles involves modification of existing boosting algorithms, as the existing algorithms focus on improving the accuracy, which is not optimal in imbalanced learning [64].

3.3 Hybrid Methods

The hybrid category in the taxonomy of Fig. 1 involves augmenting data preprocessing and algorithmic techniques to alleviate the imbalance. The hybrid ensembles

⁴ Auto-associative neural networks are a type of neural network used to simulate and explore the associative process [57].

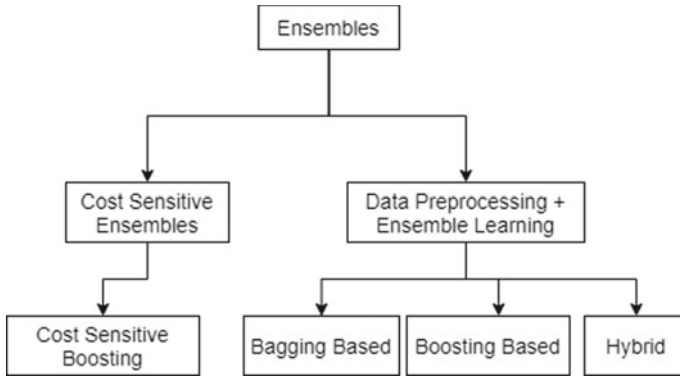


Fig. 4 Proposed taxonomy of ensemble techniques provided by the work in [25]

Table 2 An overview of bagging and boosting [63]

	Bagging	Boosting
Similarities	Uses voting Combines models of the same type	
Differences	The models are built parallelly with separation with each other	Each new model is an iteration over the old one in terms of the performance of the previous models
	Equal weight is given to all models	Weight a model’s contribution by its performance
Examples	Random forests	AdaBoost

consist of ensemble learning techniques augmented by data preprocessing methods for improving the classifier’s performance on the minority classes.

The data preprocessing + ensemble learning family of techniques are a part of the hybrid ensembles subcategory of the hybrid category in the proposed taxonomy displayed in Fig. 1 since they involve a combination of data preprocessing. For example, SMOTEBoost involves SMOTE in synthesizing examples and AdaBoost for better classification by correcting the misclassification of the previous iteration while boosting. Some examples for the algorithms of this category are RUSBoost and SMOTEBoost (boosting-based), SMOTEBagging, QuasiBagging and Under-OverBagging (bagging-based). These algorithms are discussed in detail in [25]. Zhang et al. [65] presents an approach named SIRUS which uses stacking and inverse random undersampling.

The hybrid category proposed in the taxonomy is an interesting technique where it combines bagging and boosting. Liu et al. [66] proposes two novel ensemble methods based on undersampling: EasyEnsemble and BalanceCascade which samples several balanced subsets and trains an ensemble classifier with each subset. Hence, the main shortcoming of undersampling is resolved by combining the results of several classifiers, and thus, the majority class is better utilized. Both the methods were turned

Table 3 A few instances of the domains in which class imbalance is encountered

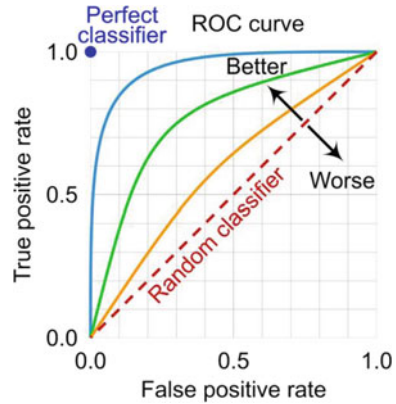
Domain	Category	Sub-category	Technique	Reference
Software defect prediction	Algorithms	Hybrid ensembles	AWEIG + AdaCost Bayesian	[70]
Software defect prediction	Data preprocessing	Undersampling	MAHAKIL: diversity-based undersampling	[2]
Network intrusion detection	Data preprocessing	Oversampling	SMOTE, Cluster-SMOTE	[71]
Network intrusion detection	Hybrid	Other hybrids	Siam-IDS	[72]
Network intrusion detection	Algorithmic	Ensemble	Stacking	[4]
Electric fraud detection	Algorithmic	Other hybrids. One-class, cost-sensitive	Cost-sensitive SVM, one-class SVM, OPF	[73]
Protein fold classification	Data processing	Oversampling	SMOTE	[74]
Weld flaw classification	Data preprocessing, algorithmic	Multiple techniques	Multiple techniques	[75]
Bioinformatics	Data preprocessing	Feature selection	Multiple techniques	[39]
Text and document classification	Algorithmic techniques	One-class learning	One-class SVM	[58]
Mobile malware detection	Algorithmic techniques	Cost-sensitive hybrids	Cost-sensitive C4.5, cost-sensitive SVM	[76]

out to be mostly much better than the fifteen other methods with a higher area under the curve (AUC), F-measure and G-mean. Another novel ensemble method where imbalance data is converted into several balanced datasets and fed into classification algorithms has been discussed in [67].

Sun et al. [68] presents an intelligent undersampling and ensemble-based classification method to resolve the problem of imbalanced classes in noisy situations, which has shown to have better performance with other classifiers.

Other algorithms which combine sampling, feature selection and a combination of one or more classifiers especially for particular domains are categorized as other hybrid techniques. For instance, [69] shows that feature selection followed by undersampling will lead to generation of better Support Vector Machines in order to account for the class imbalance in predicting protein function from sequence. Sun et al. [68] proposes a biased random forest that employs k-nearest neighbours (k-NN) algorithm in order to identify the critical areas in a training set, and based on the critical areas, the standard random forest is fed with more random trees. Another instance

Fig. 5 The ROC space for a “better” and “worse” classifier [77]. The diagonal shows the performance of a random classifier. Three example classifiers (blue, orange, green) are shown



is the proposal of an undersampling technique guided by evolutionary algorithms to perform a training set selection [4]. The viability of the models acquired is contrasted and new hybrid of oversampling through star plots and as discovered to be extremely successful in mitigating the skew.

4 Applicable Domains

The selection of a particular technique is dependent on a domain and the nature of the problem. We have listed a few domains in Table 3 on page 12, where the researchers have applied techniques for minimizing the effect of class imbalance (Fig. 5).

5 Evaluation Metrics

Choosing metrics is particularly challenging when the distribution of the classes is skewed. A higher score in a metric may be misleading, and it may not always mean that the classifier is performing well for achieving the task in hand. The work in [78] presents many metrics, out of which we are going to focus on the ones which are significant. Table 1 presents a confusion matrix for binary classification, which is required to derive the values of the metrics. Accuracy as shown in Eq. 1 being the traditional measurement metric is apt for balanced data and does not work well for dataset with a significant skew in class, as it may be poor when the misclassification of the minority class is the most crucial [25]. A high accuracy may actually result in poor performance in discriminating the minority classes, which are typically much more important to be classified than the majority classes. Hence, other metrics like precision and recall shown in Eqs. 6 and 7, respectively, are used to determine the precision, also known as positive predictive value, of the model in predicting the

minority classes which are usually the positive classes. Recall is the count of true positives divided by the sum of the count of false negatives and true positives. It is also known as sensitivity. Equation 8 is F-measure which is derived from the scores of precision and recall [79], and it is the harmonic mean of both of the above-discussed quantities, intended to balance them. G-mean provides a score which indicates the capability of a classifier to balance between the accuracies of positive class and negative class and is shown in Eq. 5. True-positive rate (TPR) is a measure of the proportion of actual positive instances that were predicted as positive. Recall, also known as sensitivity, is a measure of the proportion of actual positive cases that were predicted as positive. Equation 3 shows how it is done. A model is better at correctly identifying positive cases if the TPR is higher. Equation 4 represents specificity as the true-negative rate (TNR), which suggests that there will be a fraction of actual negatives that are forecasted as positives, which could be referred to as false positives. The sum of specificity and false-positive rate given in Eq. 2 would always be 1. A good ML model should have high specificity, which means a low FPR and a high TNR. Fawcett [80] presents additional metrics like adjusted GM, optimized precision, Mean-Class-Weighted Accuracy, Kappa, etc. The formulae for the metrics described in this section can be found below.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (1)$$

$$\text{False positive rate} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (2)$$

$$\text{True positive rate(Sensitivity)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{True positive rate(Specificity)} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (4)$$

$$G - \text{mean} = \sqrt{\text{Sensitivity} \times \text{Specificity}} \quad (5)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

$$F - \text{measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

Performance metrics receiving operating characteristic (ROC) curve and precision-recall (PR) curves are quite popular among researchers since they provide better visualization. ROC for getting a holistic graphical representation of a classifier's performance can be used to compare models using TPR and TNR, whereas PR

curves focus on the performance of the classifiers on the minority classes where it is a plot of precision, i.e. positive predictive power (x -axis) versus sensitivity (y -axis). The relationship between PR curves and ROC curves is detailed in [81]. In these curves, the performance of individual classifiers is indicated as curves in a 2D graph. The curve having the maximum area under the curve (AUC) is considered to be the best among a set of classifiers. There are other metrics which focus on improving upon ROC curves, namely [82] which introduces cost curves, which in addition to sharing many properties of ROC curves also introduces performance assessment that cannot be done using ROC curves. [5] discusses Matthews correlation coefficient as better than other metrics, as it provides high scores only if the prediction obtained good results in all cells of the confusion matrix.

6 Future Research Directions

The techniques for handling class imbalance are very diverse and humongous. As the domains which class imbalance affects are quite a lot, there are many open issues and challenges to be resolved for better classification results. A few of the future research directions that we propose are as follows:-

1. Clustering-based undersampling techniques used k-means clustering algorithms primarily. Usage of other clustering algorithms like the work [3] which uses affinity propagation [27] for undersampling has to be explored further.
2. ROC curves are the most popular metric of a classifier's performance while learning under class imbalance. However, other techniques like cost curves [82] and MCC curves [5] must also be covered in detail.
3. An effort must be made to port the existing algorithms which consist of multi-class imbalance techniques to binary class imbalance techniques [39]. However, decomposition itself involves many complexities and there is a need for a comprehensive study to be performed on all of them.
4. The datasets for training the model are usually prepared with expert supervision and sometimes may perform poorly post-training due to the difference in the underlying distribution of the training and testing datasets; the development of algorithms that are mostly independent of the distribution is an open research problem.
5. The work in [38] deals with feature selection techniques for small-scale datasets with high dimensionality. However, the authors have suggested additional study in future with respect to applying feature selection techniques and deducing optimal feature selection techniques to handle class imbalance in high dimensional datasets, where sampling methods were found to be ineffective.
6. The effect of noise on the performance of the classifiers has been generally neglected, and there is a need for work to be done in studying the characteristics of noise and its impact in predictive modelling involving imbalanced classes.

7. There is also need to develop techniques to alleviate class imbalance techniques for streaming data. Nearly all problems explore non-streaming data, and this is a concern for domains involving online learning. Non-stationary data streams should also be accounted for, especially for estimation of the cost matrix in cost-sensitive learning.

7 Conclusion

In the domain of machine learning and data mining, class imbalance remains to be one of the most important problems for which there has been a good amount of research work proposed till date. Innovative methodologies and effective evaluation strategies continue to be proposed as the shortcomings of the class imbalance problem are yet to be fully addressed. We also should note that the choice of applying various techniques to handle class imbalance that we have discussed is domain specific. In general, hybrid and ensemble strategies are to be preferred where a combination of sampling, feature selection and algorithmic techniques is to be applied, as seen in Table 3. In this way, multiple strategies compensate for the shortcomings of individual techniques. The paper, apart from including the latest proposed techniques, also dives deep into cost-sensitive and ensemble methods and provides an extensive taxonomy of handling class imbalance and evaluating the techniques, which we found to be limitations of previous review works.

In the light of the boom in data driving the connected world, classification has been affected by the problem of skewed class distribution, and alleviating this problem is a major step towards making the classifiers more performant for better results in predictive modelling.

References

1. Krawczyk B (2016) Learning from imbalanced data: open challenges and future directions. *Prog Artif Intell* 5(4):221–232
2. Bennin KE, Keung J, Phannachitta P, Monden A, Mensah S (2017) Mahakil: diversity based oversampling approach to alleviate the class imbalance issue in software defect prediction. *IEEE Trans Software Eng* 44(6):534–550
3. Tsai C-F, Lin W-C, Hu Y-H, Yao G-T (2019) Under-sampling class imbalanced datasets by combining clustering analysis and instance selection. *Inf Sci* 477:47–54
4. Rajagopal S, Kundapur PP, Hareesha KS (2020) A stacking ensemble for network intrusion detection using heterogeneous datasets. *Secur Commun Netw* 2020
5. Chicco D, Jurman G (2020) The advantages of the Matthews correlation coefficient (mcc) over $f1$ score and accuracy in binary classification evaluation. *BMC Genomics* 21(1):1–13
6. Japkowicz N (2000) The class imbalance problem: significance and strategies. In: *Proceedings of the 2000 international conference on artificial intelligence*, vol 56. Citeseer
7. Chawla NV, Japkowicz N, Kotcz A (2004) Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explor Newsl* 6(1):1–6. [Online]. Available: <https://doi.org/10.1145/1007730.1007733>

8. Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. *Intell Data Anal* 6(5):429–449
9. Das B, Krishnan NC, Cook DJ (2013) Handling class overlap and imbalance to detect prompt situations in smart homes. In: 2013 IEEE 13th international conference on data mining workshops. IEEE, pp 266–273
10. Prati RC, Batista GE, Monard MC (2004) Class imbalances versus class overlapping: an analysis of a learning system behaviour. In: Mexican international conference on artificial intelligence. Springer, pp 312–321
11. Jo T, Japkowicz N (2004) Class imbalances versus small disjuncts. *SIGKDD Explor Newsl* 6(1):40–49. [Online]. Available: <https://doi.org/10.1145/1007730.1007737>
12. Chawla NV (2009) Data mining for imbalanced datasets: an overview. In: *Data mining and knowledge discovery handbook*, pp 875–886
13. Batista GE, Bazzan AL, Monard MC et al (2003) Balancing training data for automated annotation of keywords: a case study. In: *WOB*, pp 10–18
14. Ali A, Shamsuddin SM, Ralescu AL (2013) Classification with class imbalance problem. *Int J Adv Soft Comput Appl* 5(3)
15. Kotsiantis S, Pintelas P (2003) Mixture of expert agents for handling imbalanced data sets. *Ann Math Comput Teleinform* 1(1):46–55
16. Two modifications of CNN. *IEEE Trans Syst Man Cybern SMC*-6(11):769–772 (1976)
17. Hart P (1968) The condensed nearest neighbor rule (corresp.). *IEEE Trans Inf Theory* 14(3):515–516
18. Batista GE, Prati RC, Monard MC (2004) A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explor Newsl* 6(1):20–29
19. Kubat M, Matwin S et al (1997) Addressing the curse of imbalanced training sets: one-sided selection. In: *ICML*, vol 97. Citeseer, pp 179–186
20. Wilson DL (1972) Asymptotic properties of nearest neighbor rules using edited data. *IEEE Trans Syst Man Cybern* 3:408–421
21. Laurikkala J (2001) Improving identification of difficult small classes by balancing class distribution. In: *Conference on artificial intelligence in medicine in Europe*. Springer, pp 63–66
22. Mani I, Zhang J (2003) kNN approach to unbalanced data distributions: a case study involving information extraction. In: *Proceedings of workshop on learning from imbalanced datasets*, vol 126. *ICML United States*
23. Fernández A, García S, Herrera F, Chawla NV (2018) Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *J Artif Intell Res* 61:863–905
24. Sun Z, Song Q, Zhu X, Sun H, Xu B, Zhou Y (2015) A novel ensemble method for classifying imbalanced data. *Pattern Recogn* 48(5):1623–1637
25. Galar M, Fernández A, Barrenechea E, Bustince H, Herrera F (2011) A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Trans Syst Man Cybern Part C (Appl Rev)* 42(4):463–484
26. Raykov YP, Boukouvalas A, Baig F, Little MA (2016) What to do when k-means clustering fails: a simple yet principled alternative algorithm. *PLoS ONE* 11(9):e0162259
27. Wang K, Zhang J, Li D, Zhang X, Guo T (2008) Adaptive affinity propagation clustering. *arXiv preprint arXiv:0805.1096*
28. Yen S-J, Lee Y-S (2009) Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Syst Appl* 36(3):5718–5727
29. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: synthetic minority over-sampling technique. *J Artif Intell Res* 16:321–357
30. Chawla NV (2003) C4. 5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In: *Proceedings of the ICML*, vol 3, p 66
31. Han H, Wang W-Y, Mao B-H (2005) Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: *International conference on intelligent computing*. Springer, pp 878–887
32. He H, Bai Y, Garcia EA, Li S (2008) Adasyn: adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence). IEEE, pp 1322–1328

33. Liang X, Jiang A, Li T, Xue Y, Wang G (2020) Lr-smote: an improved un-balanced data set oversampling based on k-means and Svm. *Knowl-Based Syst* 196:105845
34. Lumen. Genetics and inheritance. [Online]. Available: <https://courses.lumenlearning.com/sanjacinto-biology1/chapter/chromosomal-theory-of-inheritance-and-genetic-linkage>
35. Wong GY, Leung FH, Ling S-H (2013) A novel evolutionary preprocessing method based on over-sampling and under-sampling for imbalanced datasets. In: *IECON 2013-39th annual conference of the IEEE industrial electronics society*. IEEE, pp 2354–2359
36. Barua S, Islam MM, Yao X, Murase K (2012) Mwmote—majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Trans Knowl Data Eng* 26(2):405–425
37. Menzies T, Dekhtyar A, Distefano J, Greenwald J (2007) Problems with precision: a response to comments on ‘data mining static code attributes to learn defect predictors.’ *IEEE Trans Software Eng* 33(9):637–640
38. Wasikowski M, Chen X-W (2009) Combating the small sample class imbalance problem using feature selection. *IEEE Trans Knowl Data Eng* 22(10):1388–1400
39. Van Hulse J, Khoshgoftaar TM, Napolitano A, Wald R (2012) Threshold-based feature selection techniques for high-dimensional bioinformatics data. *Netw Model Anal Health Inform Bioinform* 1(1–2):47–61
40. Threshold-based feature selection techniques for high-dimensional bioinformatics data. *Netw Model Anal Health Inform Bioinform* 1(1–2):47–61
41. Zhou Z-H, Liu X-Y (2010) On multi-class cost-sensitive learning. *Comput Intell* 26(3):232–257
42. Sammut C, Webb GI (2011) *Encyclopedia of machine learning*. Springer Science & Business Media
43. Ling CX, Sheng VS (2008) Cost-sensitive learning and the class imbalance problem. *Encycl Mach Learn* 2011:231–235
44. Fernández A, García S, Galar M, Prati RC, Krawczyk B, Herrera F (2018) *Learning from imbalanced data sets*. Springer, vol 10
45. Turney PD (1994) Cost-sensitive classification: empirical evaluation of a hybrid genetic decision tree induction algorithm. *J Artif Intell Res* 2:369–409
46. Ling CX, Yang Q, Wang J, Zhang S (2004) Decision trees with minimal costs. In: *Proceedings of the twenty-first international conference on machine learning*, p 69
47. Drummond C, Holte RC et al (2003) C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: *Workshop on learning from imbalanced datasets II*, vol 11. Citeseer, pp 1–8
48. Zadrozny B, Langford J, Abe N (2003) Cost-sensitive learning by cost-proportionate example weighting. In: *Third IEEE international conference on data mining*. IEEE, pp 435–442
49. Domingos P (1999) Metacost: a general method for making classifiers cost-sensitive. In: *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining*, pp 155–164
50. Fan W, Stolfo SJ, Zhang J, Chan PK (1999) Adacost: misclassification cost sensitive boosting. In: *Icml*, vol 99. Citeseer, pp 97–105
51. Fernández, García S, Galar M, Prati RC, Krawczyk B, Herrera F (2018) *Cost-sensitive learning*. Springer International Publishing, Cham, pp 63–78
52. Elkan (2001) The foundations of cost-sensitive learning. In: *International joint conference on artificial intelligence*, vol 17(1). Lawrence Erlbaum Associates Ltd, pp 973–978
53. Veropoulos, Campbell C, Cristianini N et al (1999) Controlling the sensitivity of support vector machines. In: *Proceedings of the international joint conference on AI*, vol 55. Stockholm, p 60
54. TAX MJ (2001) *One-class classification*. PhD dissertation, Delft University of Technology, Delft, Netherlands
55. Attenberg J, Ertekin S (2013) Class imbalance and active learning. In: *Imbalanced learning: foundations, algorithms, and applications*, pp 101–149
56. Bellinger C, Sharma S, Japkowicz N (2012) One-class versus binary classification: Which and when? In: *2012 11th International conference on machine learning and applications*, vol 2. IEEE, pp 102–106

57. GeeksForGeeks (2021) Auto-associative neural networks. [Online]. Available: <https://www.geeksforgeeks.org/auto-associative-neural-networks>
58. Zhang H, Huang W, Huang Z, Zhang B (2005) A kernel autoassociator approach to pattern classification. *IEEE Trans Syst Man Cybern Part B (Cyber)* 35(3):593–606
59. Manevitz M, Yousef M (2001) One-class SVMs for document classification. *J Mach Learn Res* 2:139–154
60. Batista GEAPA, Prati RC, Monard MC (2005) Balancing strategies and class overlapping. In: *IDA*
61. Visa S (2007) Fuzzy classifiers for imbalanced data sets. PhD dissertation, University of Cincinnati
62. Aljamaan H, Elish M (2009) An empirical study of bagging and boosting ensembles for identifying faulty classes in object-oriented software, pp 187–194
63. Aljamaan HI, Elish MO (2009) An empirical study of bagging and boosting ensembles for identifying faulty classes in object-oriented software. In: 2009 IEEE symposium on computational intelligence and data mining, pp 187–194
64. Sun Y, Kamel MS, Wong AK, Wang Y (2007) Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn* 40(12):3358–3378
65. Zhang Y, Liu G, Luan W, Yan C, Jiang C (2018) An approach to class imbalance problem based on stacking and inverse random under sampling methods. In: 2018 IEEE 15th international conference on networking, sensing and control (ICNSC), pp 1–6
66. Liu X-Y, Wu J, Zhou Z-H (2008) Exploratory undersampling for class-imbalance learning. *IEEE Trans Syst Man Cybern Part B (Cyber)* 39(2):539–550
67. Sun, Song Q, Zhu X, Sun H, Xu B, Zhou Y (2015) A novel ensemble method for classifying imbalanced data. *Pattern Recogn* 48(5):1623–1637
68. Bader-El-Den, Teitei E, Perry T (2018) Biased random forest for dealing with the class imbalance problem. *IEEE Trans Neural Networks Learn Syst* 30(7):2163–2172
69. Al-Shahib, Breitling R, Gilbert D (2005) Feature selection and the class imbalance problem in predicting protein function from sequence. *Appl Bioinform* 4(3):195–203
70. Suntoro J, Christanto FW, Indriyawati H (2018) Software defect prediction using *aweig* + *adacost* Bayesian algorithm for handling high dimensional data and class imbalance problem. *Int J Inf Technol Bus* 1(1):36–41
71. Rodda S, Erothi USR (2016) Class imbalance problem in the network intrusion detection systems. In: 2016 International conference on electrical, electronics, and optimization techniques (ICEEOT), pp 2685–2688
72. Bedi, Gupta N, Jindal V (2021) I-Siamids: an improved Siam-IDs for handling class imbalance in network-based intrusion detection systems. *Appl Intell* 51(2):1133–1151
73. Di Martino M, Decia F, Molinelli J, Fernandez A (2012) Improving electric fraud detection using class imbalance strategies. In: *ICPRAM* (2):135–141
74. Vani KS, Bhavani SD (2013) Smote based protein fold prediction classification. In: *Advances in computing and information technology*. Springer, pp 541–550
75. Liao TW (2008) Classification of weld flaws with imbalanced class data. *Expert Syst Appl* 35(3):1041–1052
76. Chen Z, Yan Q, Han H, Wang S, Peng L, Wang L, Yang B (2018) Machine learning based mobile malware detection using highly imbalanced network traffic. *Inf Sci* 433:346–364
77. W Commons (2018) Receiver operating characteristic (ROC) curve with false positive rate and true positive rate. The diagonal shows the performance of a random classifier. Three example classifiers (blue, orange, green) are shown. [Online]. Available: <https://commons.wikimedia.org/wiki/File:Roccurve.svg>
78. Sokolova M, Lapalme G (2009) A systematic analysis of performance measures for classification tasks. *Inf Process Manage* 45(4):427–437
79. Buckland M, Gey F (1994) The relationship between recall and precision. *J Am Soc Inf Sci* 45(1):12–19
80. Fawcett T (2004) ROC graphs: notes and practical considerations for researchers. *Mach Learn* 31(1):1–38

81. Davis J, Goadrich M (2006) The relationship between precision-recall and ROC curves. In: Proceedings of the 23rd international conference on Machine learning, pp 233–240
82. Drummond C, Holte RC (2006) Cost curves: an improved method for visualizing classifier performance. *Mach Learn* 65(1):95–130

An Ameliorate Analysis of Cryptocurrencies to Determine the Trading Business with Deep Learning Techniques



Neeshad Kumar Sakure, M. V. Manoj Kumar, B. S. Prashanth, H. R. Sneha, and Likewin Thomas

1 Introduction

COVID-19 is a worldwide pandemic. Close touch with an inflamed individual permits the virus to unfold via air droplets from coughing, sneezing, and different breathing activities. There is presently no recognized cure. The cure is top-rated for prevention. The World Health Organization (WHO) has declared the COVID-19 outbreak a public fitness emergency of worldwide concern [4] because it has unfolded across the world. The pandemic has had a significant effect on worldwide monetary growth. The vital lockdown has harmed the economic system extra than it has harmed people's lives. As a result of the extended country-wide lockdown, the financial system is dealing with severe problems, including accelerated unemployment and a lack of human sources with every death. AI, in particular, is helping in this time of great distress. It aids in disorder prediction, diagnosis, and prevention. Its precise cap potential to mimic people lets it tour locations where people cannot. AI fashions can revitalize the gradual economic system by producing professional labor and automating machinery. The devastation due to COVID-19 is presently affecting all the world's most important corporations. Following the lethal virus, the 12 months, 2020 has visible a shift in trade and enterprise operations from manufacturing to marketing. The abrupt halt of recent locations at some point of the COVID-19 flu epidemic had a widespread effect on enterprise throughout sectors, especially in

Supported by Nitte Meenakshi Institute of Technology

N. K. Sakure · M. V. Manoj Kumar (✉) · B. S. Prashanth · H. R. Sneha
Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology,
Bengaluru, India
e-mail: manojmv24@gmail.com

L. Thomas
Department of Computer Science and Engineering, PESIT—M, Shivamogga, India

operationally extensive sectors wherein the COVID-19 crises extended digitalization and adoption.

An enterprise is a for-income employer made of human beings and sources that engages in professional, commercial, or business hobbies to earn money. A company may be the idea of because of the current economy's backbone. A business enterprise may be as small as an unmarried enterprise in a small city or as big as a big organization of industries unfold throughout numerous countries. A single man or woman or a set of lots of humans can personal a business [21]. There are several kinds and kinds of businesses, and all of them, mainly the bigger corporations, were stimulated through cutting-edge technology. Together with Google, Facebook, and Amazon, many massive corporations are engaged in a technological hands race.

In the modern contemporary-day world, the era is constantly advancing and having a full-size effect throughout all sectors [1]. It has prompted clinical diagnosis, business, and lots of different critical industries [27]. Many companies in brand new global have started to rent contemporary-day era to reinforce their income and expedite their growth. Artificial intelligence (AI), data science (DS), big data (big data), and the Internet of Things (IoT) have modified company environments and the manner people behave in business. There is not always a single area of labor these days that has not seemed in AI's applications [29]. It is viable to study how AI and computational technology are implemented within commercial and era industries. It has long been set up that machines can outperform or, as minimum healthy human beings in numerous tasks, which include emotion detection, tacit judgment, and automation [28]. According to a few estimates, the computational era may want to take over as many as forty-seven percentages of the world's gift jobs in as low as ten years [30].

Technology has had a terrific effect on the economy, and speedy technological improvements inside the subsequent years will appreciably adjust the contemporary financial environment. As a result, it is vital to understand how trendy contemporary technology is affecting the company sector. AI has been successfully hired to do occupations that call for higher-order innovative thinking, along with the paintings of journalists, attorneys, lab technicians, and paralegals. Many jobs are being changed through pc technology, AI, and different computational techniques, but that is best for low-professional jobs like clerical labor. The call for high-professional employment and jobs that necessitate improved computing structures has risen and maintains to push rapidly upward. Soni et al. [26] have a look at a hundred AI start-ups round the arena and document a fascinating conclusion. In 2011, they invested \$25.88 million inside the hundred organizations they reviewed, which climbed to \$1866.6 million in 2016. In simply six years, those organizations' funding surged via way of means of a lovely 7112—fifty-two percent. The current company surroundings are being shaken via a form of means of synthetic intelligence and massive data.

The outline of this paper is as follows. Section 3 discusses the type of cryptocurrency. Section 2 explains the various related work. Section 4 depicts the proposed methodology of this work. Section 5 shows the performance analysis for the conventional and proposed methodology. Finally, Sect. 6 concludes this work.

2 Related Works

Artificial intelligence and massive facts answers are utilized in a whole lot of regions in finance and business, which include loan/coverage underwriting, fraud detection, patron service, sentiment/information analysis, algorithmic trading, portfolio management, marketing, economical product advice systems, commercial advice, and so on. The following sections undergo a few AI strategies and their packages in many domains. The objective of this project is to give technical and fundamental analysis using machine learning approaches.

One of the maximum essential thoughts in economics and commercial enterprise is buying and selling. The shopping and promoting of a financial entity, including goods, stocks, currencies, and so forth, maybe kind of characterized as buying and selling. Individuals and companies have interaction in buying and selling to make a profit. Pre-change evaluation, buying and selling sign creation, change execution, and post-change evaluation are the four additives of the buying and selling process [19]. Algorithmic buying and selling is an extended period that refers back to the automation of any aggregate of those procedures or all of them. Artificial intelligence (AI) has essentially altered this enterprise by automating the buying and selling process, and lots of buying and selling algorithms can generate income without the intervention of a human.

One of the maximum urgent financial troubles of our time is fraud. Across the globe, fraudulent behavior prices companies billions of dollars. According to several estimates, overall fraud losses were \$27. Eighty-five billion in 2018 and this parent is predicted to upward thrust to \$40. Sixty-three billion over the following ten years [23]. This sum is more than numerous growing countries' annual GDP. As a result, pc structures able to figure out and stop fraud are essential, as doing so could extensively boost up the corporation and financial growth. Researchers have devised structures that integrate synthetic intelligence and system learning to assemble such structures. Anomaly detection and misuse detection are the two-word classes that each one of anomaly detection and misuse detection is fraud detection techniques fall into [31]. Anomaly detection learns a consumer's transaction behavior. Any new transaction completed via way of that consumer is assessed as every day or bizarre primarily based totally on the consumer's initial transactions. The version has constructed the usage of tagged facts set of all consumers, and fraudulent behavior is decided primarily based totally on popular fraudulent trends.

In [13], advise a learning-primarily based inventory fee prediction technique. They use the New York inventory trade dataset, incorporating the records, open fee, near fee, and volume. They use the long short-term memory recurrent neural network (LSTM). LSTMs function a reminiscence molecular that correlates to neurons, much like conventional synthetic neural networks. These reminiscence cells might also offer a hyperlink among reminiscences within the enter and the draw close records structure, resulting in a unique prediction. The LSTM they hired had a sequential enter layer, LSTM layers, a dense layer that used the ReLU activation characteristic, and a linear activation characteristic output layer. Roondiwala et al. ran a sequence of

tests, enhancing numerous parameters, and observed that the very best acting LSTM had an RMSE (Root suggest rectangular error) of 0.00859, which is relatively low, proving that synthetic intelligence may be used to as it should be are expecting inventory values.

Customer assist has additionally been automatic the usage of synthetic Intelligence and statistics technology techniques [12]. Businesses now do not want to recruit expert customer service representatives due to AI technology. A chatbot is to be had 24 h a day, seven days per week can reply to many client questions, and this technique has proven to be very famous in the latest years in a lot of industries. AI can control customer questions from a good-sized expertise database from which it has learned. This notably lowers a firm's employment costs, and the identical paintings can be performed for much less money, allowing the agency to flourish.

Marketing has additionally been impacted through synthetic intelligence (AI) and massive data. Data-pushed advertising techniques are appreciably greater powerful than human-primarily based advertising tactics. It is all approximately YouTube those days. In reality, Facebook, Instagram, and all different social media web websites leverage synthetic intelligence and device gaining knowledge of technology to offer specific and personalized advertisements. A guitar fanatic is more likely to come upon a guitar-associated commercial on such networks. In the enterprise realm, synthetic intelligence has arguably had the most satisfactory effect on advertising [10].

Table 1 summarizes previous work particularly that completed in recent years. According to the survey, artificial intelligence can be utilized to help businesses flourish in a variety of ways, from fraud detection to product recommendations. Business analytics is used by stock traders and Bitcoin traders to build predictive models.

3 Cryptocurrency

Bitcoin is a peer-to-peer (p2p) price coins machine that turned into created in 2008 as non-regulated virtual forex without a felony standing. It is classed as a type of cryptocurrency due to its cryptographic characteristic inside the technology and switch of funds. Bitcoin has been the famous maximum forex inside the region of quantity buying and selling in current years, making it the leading promising economic medium for investors [18]. It secures the transaction with the aid of using encrypting the sender, receiver, and transaction quantity [11].

Ethereum (XRP) is a Turing-entire decentralized blockchain primarily based totally framework for growing and executing clever contracts, and disbursed systems [7, 25]. The cost of the coin is known as ether. Buterin based it in 2013, and it turned into funding 12 months later with a complete of US\$18 million in Bitcoins raised via a web public crowd sale. Ether has no regulations on its move and may be traded on cryptocurrency exchanges. It is now no longer supposed to be a fee system; instead, it is considered for use inside the Ethereum network [5].

Table 1 Summary of the survey on data analysis of business trading using Cryptocurrency

Authors	Method used	Cases used
Gupta et al. [14]	ANN with PSO	Phishing detection in cyber security
Kim et al. [17]	LSTM	Intrusion detection in cyber security
Cui et al. [9]	NPL + Deep ANN	Customer service
Colianni et al. [8]	NPL + SVM/NB	Bitcoin price prediction
Pramod and Mallikarjuna Shastry [22]	LSTM	Stock price prediction
Randhawa et al. [24]	ANN + NB	Fraud detection
Xuan et al. [31]	Random forest	Fraud detection
Awoyemi et al. [3]	KNN/NB	Fraud detection
Paradarami et al. [20]	Deep ANN	Product recommendation system

Charles Lee created Litecoin (LTC), which was launched in October 2011 and made use of a comparable generation to Bitcoin. The block era time has been reduced in half (from 10 to 5 min according to block), and the most restriction has been raised to eighty-four million that is four instances that of Bitcoin [15]. Litecoin is the cryptocurrency silver standard, and it is far now the second one maximum extensively regular through each miner and exchanges. It makes use of the scrypt encryption set of rules, which differs from SHA-256, and became designed to hurry up transaction affirmation at the Bitcoin network. It also uses a set of rules, which is proof against hardware advancements.

NEM is a peer-to-peer community and blockchain notarization platform that allows customers to ship and acquire cash online. Because it has a collectively owned notarization, NEM turns into the primary public/personal blockchain combo [7]. Ripple is a dispensed peer-to-peer community price medium owned and maintained via way of means of a single company [11]. It became based via Jed McCaleb and Chris Larsen as an open supply virtual cash. It additionally provides some other layer of security. The Byzantine Consensus Protocol became used to construct Ripple, and the most range of Ripple is one hundred million.

Stellar, like Ripple, is a complete safety tool this is built the usage of the Byzantine Consensus Protocol. Stellar has installed a brand new device to execute economic transactions that consist of open source, dispersed ownership, and countless ownership [11].

4 Proposed Methodology

All models on this have a look at are fed with time collection records primarily based totally on five years of everyday history; however, this could alternate relying upon the datasets to be had from the source. Between 2013 and 2018, the records become generated from every day open, closing, high, and occasional fees, of every day, buying and selling for a complete of six distinct varieties of cryptocurrencies, and it becomes obtained from the marketplace capitalization database.

Mastering evaluation is important for buying and selling success. Technical evaluation and essential evaluation are strategies for figuring out destiny value. Technical evaluation forecasts destiny charge the use of buying and selling statistics from the market, which includes charge and buying and selling volume, while different strategies use statistics from out of doors the market, which includes monetary conditions, hobby rates, and geopolitical events, to estimate destiny direction [6]. Many buyers give attention to technical evaluation, while others give attention to essential evaluation. Some buyers, on the alternative hand, are inquisitive about the overlaps among essential and technical evaluation. The goal of this project is to use machine learning techniques to provide technical analysis. Machine learning has been established as a serious model in classical statistics in the forecasting sector for more than two decades [2, 16]. The proposed methodology is shown in Fig. 1.

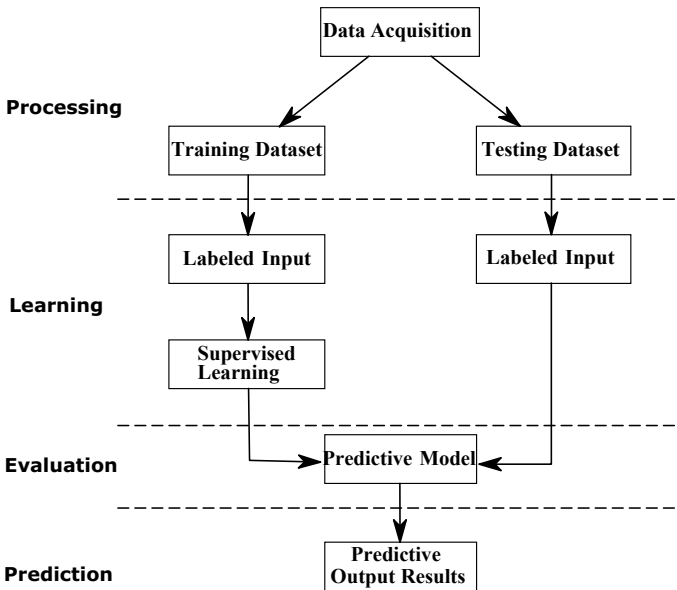


Fig. 1 Proposed methodology

Table 2 Common terminologies

Variable	Description
Open price	The first price of a given cryptocurrency in a daily trading
Close price	The price of the last transaction for a given cryptocurrency at the end of a daily trading
High price	The highest price that was paid for a cryptocurrency during a daily trading
Low price	The lowest price of a cryptocurrency reached in a daily trading

4.1 Artificial Intelligence Methods—Useful in the Business and Economic Sectors

To get more value out of business analytics, use advanced artificial intelligence approaches,

1. Create strong prediction and classification models using neural networks, genetic algorithms, support vector machines, and fuzzy systems, among other methods
2. Improve fraud detection, cross-selling, credit score analysis, and profiling
3. New case studies and examples from around the company are included.

Artificial intelligence can help people get more value out of business analytics, account for uncertainty and complexity more effectively, and make smarter decisions. This book delves into today's most important artificial intelligence principles, tools, knowledge, and tactics, as well as how to put them to use in the real world. Some of the common terminologies are specified in Table 2.

5 Results and Discussion

The performance measures for each cryptocurrency type according to classifiers are displayed first in the result Sect. 4. These serve as a checkpoint for the rest of the conversation. The investigation is divided into two major experiments: (i) various classifiers' performance measures and (ii) machine learning algorithms' predicted Bitcoin value versus actual value. On the cryptocurrency market capitalization, Fig. 3 demonstrates the performance accuracy in relation to four classifiers. The training and testing datasets in our time series data are shown in Fig. 2 (Table 3).

6 Conclusion and Future Scope

The worldwide economy has been significantly impacted by restrictions in human activities around the world as a result of the breakup of diverse company operations. Several local and worldwide business and commerce sectors have been affected by

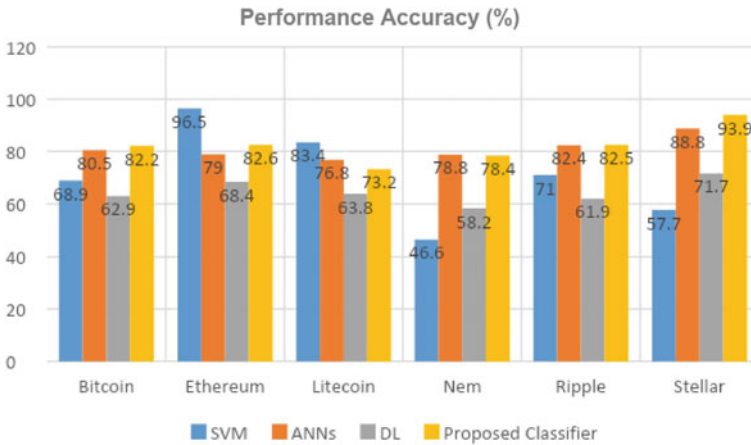


Fig. 2 Results obtained

Table 3 Cryptocurrencies with different testing and training data

Cryptocurrencies	Training data	Testing data
	Observation	Observation
Bitcoin	1388	364
Ethereum	526	364
Litecoin	1358	364
NEM	657	364
Ripple	1262	364
Stellar	896	364

the pandemic. As a result, it throws the demand–supply equation into disarray. AI approaches and strategies are being used by both large corporations and small businesses. In real-time data analysis, the rise of AI and big data may be utilized to identify, track, and forecast. The digitization of commerce enterprises allows them to bring their services and products to the doorsteps of their customers. Despite the fact that AI’s contribution to the business and commerce industries appears to be promising, it does have significant limits. The real Bitcoin value was compared to the predicted cryptocurrency value by machine learning. The finding is also investigated further using the mean absolute percentage error (MAPE) calculation. The incorporation of new technology into financial institutions may cause Bitcoin values to skyrocket. In the current scenario, I cannot guarantee or anticipate the future growth of cryptocurrencies. But I am clear regarding one thing: decentralization is a technological revolution in the making, and cryptocurrencies have the potential to revolutionize the financial world (Table 4).

Table 4 Performance measures by various classifiers

Performance accuracy (%)						
Classifiers	Bitcoin	Ethereum	Litecoin	NEM	Ripple	Stellar
SVM [6]	68.90	96.50	83.40	46.60	71.00	57.70
ANN [16]	80.50	79.00	76.80	78.80	82.40	88.80
DL [2]	62.90	68.40	63.80	58.20	61.90	71.70
Proposed classifier	82.20	82.60	73.20	78.40	82.50	93.90

References

- Adhikari S, Thapa S, Shah BK (2020) Oversampling based classifiers for categorization of radar returns from the ionosphere. In: 2020 international conference on electronics and sustainable communication systems (ICESC). IEEE, pp 975–978
- Ahmed NK, Atiya AF, Gayar NE, El-Shishiny H (2010) An empirical comparison of machine learning models for time series forecasting. *Economet Rev* 29(5–6):594–621
- Awoyemi JO, Adetunmbi AO, Oluwadare SA (2017) Credit card fraud detection using machine learning techniques: a comparative analysis. In: 2017 international conference on computing networking and informatics (ICCNi). IEEE, pp 1–9
- Bhatt G (2020) Agriculture and food e-newsletter
- Buterin V et al (2014) A next-generation smart contract and decentralized application platform. White Paper 3(37)
- Chaigusin S (2014) An application of decision tree for stock trading rules: a case of the stock exchange of Thailand
- Chuen DLK, Guo L, Wang Y (2017) Cryptocurrency: a new investment opportunity? *J Altern Investments* 20(3):16–40
- Colianni S, Rosales S, Signorotti M (2015) Algorithmic trading of cryptocurrency based on twitter sentiment analysis. CS229 Project, pp 1–5
- Cui L, Huang S, Wei F, Tan C, Duan C, Zhou M (2017) Superagent: a customer service chatbot for e-commerce websites. In: Proceedings of ACL 2017, system demonstrations, pp 97–102
- Erevelles S, Fukawa N, Swayne L (2016) Big data consumer analytics and the transformation of marketing. *J Bus Res* 69(2):897–904
- Farell R (2015) An analysis of the cryptocurrency industry. *Wharton Res Scholars J Paper* 130
- Ghimire A, Thapa S, Jha AK, Adhikari S, Kumar A (2020) Accelerating business growth with big data and artificial intelligence. In: 2020 fourth international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC). IEEE, pp 441–448
- Ghosh A, Bose S, Maji G, Debnath N, Sen S (2019) Stock price prediction using LSTM on Indian share market. In: Proceedings of 32nd international conference on computer applications in industry and engineering, vol 63, pp 101–110
- Gupta S, Singhal A (2017) Phishing URL detection by using artificial neural network with PSO. In: 2017 2nd international conference on telecommunication and Networks (TEL-NET). IEEE, pp 1–6
- Heid A (2013) Analysis of the cryptocurrency marketplace. Retrieved 15 Feb 2014
- Hitam NA, Ismail AR (2018) Comparative performance of machine learning algorithms for cryptocurrency forecasting. *Ind J Electr Eng Comput Sci* 11(3):1121–1128
- Kim J, Kim J, Thu HLT, Kim H (2016) Long short term memory recurrent neural network classifier for intrusion detection. In: 2016 international conference on platform technology and service (PlatCon). IEEE, pp 1–5
- Krause D, Pham N (2017) Bitcoin a favourable instrument for diversification? A quantitative study on the relations between bitcoin and global stock markets

19. Nuti G, Mirghaemi M, Treleaven P, Yingsaeree C (2011) Algorithmic trading. *Computer* 44(11):61–69
20. Paradarami TK, Bastian ND, Wightman JL (2017) A hybrid recommender system using artificial neural networks. *Expert Syst Appl* 83:300–313
21. Prakash S, Joshi S, Bhatia T, Sharma S, Samadhiya D, Shah RR, Kaiwartya O, Prasad M (2020) Characteristic of enterprise collaboration system and its implementation issues in business management. *Int J Bus Intell Data Min* 16(1):49–65
22. Pramod B, Mallikarjuna Shastry PM (2020) Stock price prediction using LSTM
23. Rajora S, Li DL, Jha C, Bharill N, Patel OP, Joshi S, Puthal D, Prasad M (2018) A comparative study of machine learning techniques for credit card fraud detection based on time variance. In: 2018 IEEE symposium series on computational intelligence (SSCI). IEEE, pp 1958–1963
24. Randhawa K, Loo CK, Seera M, Lim CP, Nandi AK (2018) Credit card fraud detection using adaboost and majority voting. *IEEE Access* 6:14277–14284
25. Seys J, Decaestecker K (2016) The evolution of bitcoin price drivers: moving towards stability. Unpublished Masters' Thesis. University of Ghent, Ghent
26. Soni N, Sharma EK, Singh N, Kapoor A (2019) Impact of artificial intelligence on businesses: from research, innovation, market deployment to future shifts in business models. arXiv preprint [arXiv:1905.02092](https://arxiv.org/abs/1905.02092)
27. Thapa S, Adhikari S, Ghimire A, Aditya A (2020) Feature selection based twin support vector machine for the diagnosis of Parkinson's disease. In: 2020 IEEE 8th R10 humanitarian technology conference (R10-HTC). IEEE, pp 1–6
28. Thapa S, Adhikari S, Naseem U, Singh P, Bharathy G, Prasad M (2020) Detecting Alzheimer's disease by exploiting linguistic information from Nepali transcript. In: International conference on neural information processing. Springer, pp 176–184
29. Thapa S, Singh P, Jain DK, Bharill N, Gupta A, Prasad M (2020) Data-driven approach based on feature selection technique for early diagnosis of Alzheimer's disease. In: 2020 international joint conference on neural networks (IJCNN). IEEE, pp 1–8
30. Wright SA, Schultz AE (2018) The rising tide of artificial intelligence and business automation: developing an ethical framework. *Bus Horiz* 61(6):823–832
31. Xuan S, Liu G, Li Z, Zheng L, Wang S, Jiang C (2018) Random forest for credit card fraud detection. In: 2018 IEEE 15th international conference on networking, sensing and control (ICNSC). IEEE, pp 1–6

Gender Prediction Using Iris Features



Bhuvaneshwari Patil and Mallikarjun Hangarge

1 Introduction

The abundant research in human authentication features used was extracted from the face. In recent years, texture feature extraction [10] from the iris image has drawn attention as a means of the soft biometric attribute in identifying the gender of a person. The major advantage of using soft biometrics is that it helps in the faster retrieval of identities when aggregated with corresponding biometric data. Iris information had effectively applied in diverse areas as airport check-in or refugee control [1] and can be used in cross-spectral matching scenarios [5] while comparing RGB images and NRI images. By improving the recognition attributes and accuracy provides additional semantic information about an unfamiliar area that fills the gap between machine and human descriptions about entities [1].

Iris texture feature extraction is well protected as it is an internal organ of the eye and externally visible from a distance, unique and has a highly complex pattern. The pattern is stable over the lifetime except for pigmentation. Images of the iris are taken in visible and near-infrared light. The outside layer, which includes the sclera and cornea, is fibrous and protective; the middle layer, which includes the choroid, ciliary body, and iris, is vascular; and the innermost layer, which includes the retina, is nerve or sensory [11].

The major challenges in extracting iris information are the distance between camera and eyes, occlusion by the eyelid, eyelashes, eye rotation, and the light effect in acquiring the image. The camera placed at a distance will capture inconsistent iris size. Occlusion by eyelids and eyelashes may result in inappropriate and/or

B. Patil (✉)
Gulbarga University, Kalaburagi, India
e-mail: bsp14052001@gmail.com

M. Hangarge
Department of Computer Science, KASC College Bidar, Bidar, India

insufficient features. The variation in light will cause pupil dilation, which affects the segmentation method. Eye rotation or tilting head adds variations in the segmentation process because of intra-class variations.

The aim of this paper is to experiment the gender prediction dependencies like whole eye image or normalized iris image, the split dataset as between training and testing data, feature extraction methods traditional machine learning models or neural network models, small dataset or augmented dataset. Rest of the paper discusses about the general gender prediction steps, related work in gender prediction using iris images, discussion of the results, and conclusion of the work.

2 General Steps

As iris recognition is safe, authentic, stable, it is regarded as accurate soft biometrics, and the same steps are adopted for predicting gender. Researchers might experiment with freely available database resources as listed in Table 1 to extract information specific to humans. The major and common steps involved in these areas are listed in Fig. 1. The first important step in iris recognition is the iris localization or segmenting the iris portion from the eye image. The major challenges to be addressed in localization are occlusion by eyelashes, eyelid, tilted head while capturing and illumination effect. Once the iris region is localized, it needs to be normalized to reduce or suppress the unwanted or noise information, also called enrollment. The iris information is in a circular, polar coordinate system until this phase. Daugman's rubber sheet model [6] converts iris information from a polar coordinate system to a Cartesian coordinate system, i.e., unwrapping. After unwrapping, feature extraction algorithms like LBP, BSIF, LPQ, Gabor filter, CNN are applied to extract features used for classification based on the type of application.

Images in the visible spectrum (380–750 nm) or the near-infrared band (700–900 nm) are collected by the sensors. The visible spectrum images can be saved as either a color or an intensity image; however, the NIR images are always saved as an intensity image. Literature study shows that higher accuracy is obtained for the experiments done on a person-disjoint dataset for testing and training model for NIR images than visible light images because visible light sensors are more prone to noise.

3 Related Work

Thomas et al. [18] published the first paper on gender prediction from geometric and texture features of iris images. The researchers combined the CASIA Dataset, UPOL Dataset, and UBIRIS Dataset (a total of 57,137 images) with equal distribution of all genders, generated a feature vector by applying 1D Gabor filters to the normalized iris image using Daugman's rubber sheet method, used information gain for feature

Table 1 Iris dataset

S. No.	Dataset	Creator	Size (no. images)	Resolution	Format	Remark
1	CASIA	The Centre for Biometrics and Security Research (CBSR) at Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China	756 (V1)	320 × 280	BMP	NIR
			1200 (V2)	640 × 480	BMP	
			22,034 (V3)		JPEG	
4	UPOL	The University of Palack'eho and Olomouc	384	576 × 768	PNG	RGB
5	BATH	The researchers in Biometric Signal Processing group of Department of Electronics and Electrical Engineering, University of Bath, UK	1000	1280 × 960	JPEG	
6	UBIRIS	Soft Computing and Image Analysis Group (SOCIA Lab.), Department of Computer Science, University of Beira Interior, Covilhã, Portugal	1877 (UBIRIS.v1)	2560 × 1704	JPEG	Visible wavelength
			11,102 (UBIRIS.v2)	400 × 300	TIFF	
			UBIPr	A version of the UBIRIS.v2 database, periocular recognition		
9	MMU	The research group at Multimedia University, Malaysia	450 (MMU1) 995 (MMU2)	320 × 240	BMP	

(continued)

Table 1 (continued)

S. No.	Dataset	Creator	Size (no. images)	Resolution	Format	Remark
10	WVU	The research group at West Virginia University USA	1852			Synthetic Iris dataset collection
11	ND (GFI) Iris database	The Computer Vision Research Lab (CVRL) at University of Notre Dame, USA	64,980			
12	IIT Delhi	IIT Delhi, New Delhi, India	1120	320 × 240	BMP	

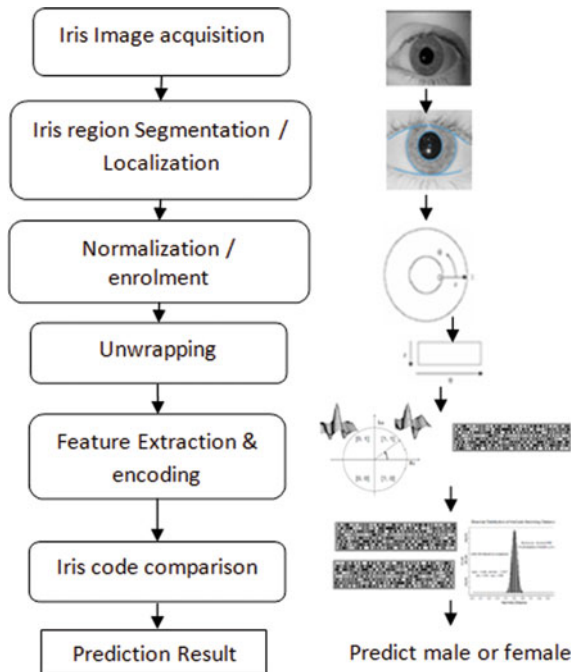


Fig. 1 Steps involved in gender prediction using iris data

selection, and later applied C4.5 decision tree algorithm for classification. Initially, the authors have used SVM and neural networks for classification. However, they could not get better results than the decision tree techniques. The authors achieved 75% accuracy and enhanced it to 80% by collecting bagging and random subspaces with a decision tree. Here, the authors have considered only the left iris for the experimentation.

Lagree and Bowyer [9] carried the gender prediction based on the SVM classifier training. The classification is based on the features generated by applying simple texture feature extraction methods like spot detector, line detector, laws texture features on normalized iris image of size 40×240 and eliminated the occlusions like the eyelid, eyelash, etc. The accuracy achieved by the authors using twofold, fivefold, and tenfold cross-validation with the Weka SMO SVM classifier was about 62%. The authors claim that their accuracy is less than Thomas et al. because of smaller size of the dataset. The researchers have used the same dataset for predicting both gender and ethnicity.

Tapia et al. [16] claimed accuracy of 91.33% in gender prediction using SVM classifier for uniform LBP and conventional LBP for subject-disjoint dataset for training and testing and also used tenfold validations. The Gabor filters were applied to the normalized image and then transformed into binary iris code with four levels, which was considered as more stable iris information for predicting gender.

Tapia et al. [17] clarified that authors had used 1500 images from unique subjects in [16] with incorrect labels. They were able to achieve 91% accuracy and were able to get this due to overlapping training and testing datasets. In [17], the disjoint train-test sets were created concerning the subject and used mutual information measures (mRMR, CMIM, weighted mRMR, and weighted CMIM) for feature selection tested for statistical significance of gender information distribution across the different bands of the iris using ANOVA test. In this current work, the three datasets used are: the UND Dataset, ND-Gender-From-Iris (NDGFI) Dataset, and a subject-disjoint validation set (UND V). The authors observed that CMIM gives better accuracy than mRMR and obtained 89% of prediction accuracy by fusing the best features from left and right iris code.

Tapia and Aravena [14] proposed a modified Lenet-5 CNN model for achieving a better gender prediction rate. The modified network consists of four convolution layers and one fully connected layer with a minimum number of neurons. A minimum number of neurons are considered to reduce the risk of over-fitting and solve the two-class gender prediction problem. The authors adopted data augmentation to increase the dataset size from 1500 to 9500 images for each eye. The authors conclude that the fusion of CNN for the right and left eye gives better prediction than the single eye, separately.

Tapia and Perez [14] used 2D quadrature quaternionic filter for classification and replaced the 1D log-Gabor filter with 2D Gabor filters. The 2D Gabor filters encoded with the normalized image phase information consist of 4 bits per pixel. The authors conducted five experiments. At first, using all the features from the normalized image for classification and other experiments are built over this model. The second experiment used transfer learning with a VGG19 model for extracting features. The next

experiment applied a genetic algorithm for selected blocks of normalized images and used raw pixel values, principal component analysis (PCA), and local binary patterns (LBP) as features. The fourth experiment was conducted using different variants of mutual information for feature extraction and used SVM and ten ensemble classifiers for classification. In the last experiment, gender classification was done using the encoding images with quaternioc code (QC) with 3 and 4 bits per pixel and observed that 4 bits per pixel show better results than 3 bits per pixel. The authors achieved maximum accuracy of 95.45% for gender prediction.

Tapia and Arellano [15] proposed modified binary statistical image features (mBSIF) for gender prediction. The experiments were carried out with different filter sizes ranging from 5×5 to 13×13 and number of bits from 5 to 12 and observed that 11×11 shows better prediction accuracy for MBSIF histogram with 94.66% for the left eye and 92% for right eye with 10 bits per pixel.

Bobeldyk and Ross [1] made an attempt to find the extended ocular region, the iris-excluded ocular region, the iris-only region, and the normalized iris-only region was used to determine the gender prediction accuracy. The authors used BSIF code for feature extraction and applied SVM classifier for the classification of males and females. They made the geometric adjustment so that the iris was at the center of the image and tessellated it into blocks. Then, the histogram of BSIF is evaluated for each region. The histograms are normalized before concatenating them into a feature vector. The resulting feature vector is used for classification. The authors also observed the prediction accuracy by varying window sizes for BSIF and obtained an accuracy of 85.7%. For the research, BioCOP2009 Dataset was used.

Bobeldyk and Ross [2, 3] expanded their earlier work [1] by considering local binary pattern (LBP) features along with BSIF features and were able to achieve maximum accuracy of 87.9%. The author also observed the impact of a number of bits in BSIF code with respect to the computational time and memory. And the impact of race on gender prediction also tested the results with the cross dataset. They used three different datasets (BioCOP2009 Dataset, Cosmic contact Dataset, and GIF Dataset) for their research.

Bobeldyk and Ross [4] have investigated the impact of resolution on gender prediction without reconstructing the low-resolution image to a high-resolution image. Used BioCOP2009 Dataset and Cosmic contact Dataset for their research. In this work, researchers used BSIF code with SVM classifier and CNN-based classifier and observed 72.1% and 77.1% accuracy for the 30-pixel image, respectively. Authors have used small networks with fewer neurons for CNN as the input image's size is small and needs smaller training samples. Also, they carried out experiments on gender prediction accuracy by varying the window size from 340×400 to 2×3 and concluded that 5×6 ocular images contain gender information with reduced complexity.

Singh et al. [12] utilized a variation of an auto-encoder in which the attribute class label has been included in conjunction with the reconstruction layer. They used NIR ocular pictures that had scaled down to 48×64 pixels. The GFI and ND-Iris-0405 Datasets were used for their method. The authors applied RDF and NNet classifiers and achieved an accuracy of 83.17%. They claim that the deep class encoder only

takes a quarter of the overall training time, and their results outperform the outcomes of Tapia et al. [17].

Sreya and Jones [13] used the IITD Dataset for investigation and ANN for iris recognition. The authors explained the steps involved in recognition in detail. The experiments were carried out on cropped NIR images to locate the pupil region. The authors conclude that the prediction accuracy depends on processing.

Kuehlkamp and Bowyer [7] investigated the impact of mascara on iris gender prediction. They got a 60% gender prediction accuracy using only the occlusion mask from each image and 66% accuracy when LBP was used in conjunction with an MLP network. Also, they were able to attain up to 80% accuracy using the complete ‘eye’ image using CNNs and MLP’s. The authors used the GFI Dataset and classified it as Males, Females With Cosmetics (FWC) and Females No Cosmetics (FNC).

4 Experiments and Results

In this work, the experiments are conducted by adopting different approaches to know the suitable criteria for the prediction. We have used two publicly available datasets: IITD Dataset [8] with image size of 320×240 and SDUMLA-HMT Dataset with 768×576 . Both the datasets have female eye image count less than that of male eye image count, so the eye images are augmented to generate 11,512 male eye images and 11,906 female eye images that meet experimentation purpose.

Initial experiments were conducted using traditional machine learning classification methods based on normalized iris texture features as shown in Fig. 1, as cited in literature study. We have used local binary pattern (LBP), Gabor filter-based feature extraction methods for getting the texture features from the normalized iris image and used SVM and random forest for classification. The experiments are carried out using the IITD Dataset [8] and SDUMLA-HMT Dataset, the results are given as in Table 2, and SVM for Gabor features shows enhanced results.

Next experiment was done using dense neural network for classification with 20% dropout and convolution neural network for feature extraction from whole eye image and normalized iris images. Deep neural network gives an accuracy of 73.96% and 90.97% for SDUMLA-HMT Dataset when trained using whole eye image and normalized iris image, respectively, and an accuracy of 98.92% for normalized IITD Dataset.

Another experiment is conducted by varying the split ratio of training and testing data. The split is done as 60:40, 80:20, and 90:10 for training and testing and observed that the results show better results for the support vector machine (SVM) for smaller dataset and deep neural network shows better accuracy for larger dataset independent of the split ratio of training and testing data, as shown in Table 2.

Table 2 Experimentation results

Training–testing data split	Classifier	Accuracy (%)			Accuracy (%)		
		Male	Female	Overall	Male	Female	Overall
Gabor filter features		IITD dataset			SDUMLA-HMT dataset		
60:40	Random forest	84.76	82.92	84.60	67.32	67.42	67.36
80:20		85.22	85.71	85.27	72.34	61.64	67.64
90:10		88.23	87.90	89.28	73.07	62.50	69.04
60:40	SVM	95.59	94.12	95.31	67.34	57.22	67.36
80:20		98.31	97.82	98.21	63.94	69.0	67.06
90:10		97.80	95.24	97.32	62.02	70.02	63.1
LBP features		IITD dataset			SDUMLA-HMT dataset		
60:40	Random forest	78.61	44.44	77.76	59.73	62.74	60.49
80:20		79.77	72.72	79.56	62.69	50.64	58.12
90:10		78.09	40.40	77.35	59.72	60.0	59.80
60:40	SVM	85.83	62.39	82.04	57.43	50.92	54.81
80:20		87.34	74.07	85.35	63.79	50.57	58.01
90:10		89.93	78.12	87.84	57.62	60.6	59.8

5 Conclusion

The experiments are carried out to study feature extraction and classification methods’ appropriate for gender prediction. The results in Table 2 show that SVM shows better outcome for smaller dataset, independent of the feature extraction method. The gender prediction accuracy increases when normalized iris images are used as input for feature extraction methods, and Gabor filter-based feature extraction shows better gender prediction accuracy. The neural network model was trained for gender prediction using whole eye images and normalized iris images; the gender prediction accuracy is high for normalized input with greater dataset size. Observations are made that SDUMLA-HMT images contain full eye image including eyelids, noisy images, and pupil is not the center of the images which makes iris localization and normalization more challenging. So it is observed that IITD Dataset shows good accuracy as compared with the SDUMLA-HMT Dataset as images are focused on region of interest with minimum noise. Further, the same setup can predict other soft biometric predictions which are like age and ethnicity.

References

1. Bobeldyk D, Ross A (2016) Iris or periocular? Exploring sex prediction from near infrared ocular images. Lecture notes in informatics (LNI). In: Proceedings—series of the Gesellschaft Fur Informatik (GI), p 260. <https://doi.org/10.1109/BIOSIG.2016.7736928>

2. Bobeldyk D, Ross A (2018a) Predicting eye color from near infrared iris images. In: Proceedings—2018 international conference on biometrics, ICB 2018, pp 104–110. <https://doi.org/10.1109/ICB2018.2018.00026>
3. Bobeldyk D, Ross A (2018b) Analyzing covariate influence on gender and race prediction from near-infrared ocular images. <https://doi.org/10.1109/ACCESS.2018.2886275>
4. Bobeldyk D, Ross A (2019) Predicting soft biometric attributes from 30 pixels: a case study in NIR ocular images. In: Proceedings—2019 IEEE winter conference on applications of computer vision workshops, WACVW 2019, pp 116–124. <https://doi.org/10.1109/WACVW.2019.00024>
5. Dantcheva A, Elia P, Ross A (2016) What else does your biometric data reveal? A survey on soft biometrics. *IEEE Trans Inf Forensics Secur* 11(3):441–467. <https://doi.org/10.1109/TIFS.2015.2480381>
6. Daugman J (2004) How iris recognition works. *IEEE Trans Circuits Syst Video Technol* 14(1):21–30. <https://doi.org/10.1109/TCSVT.2003.818350>
7. Kuehlkamp A, Becker B, Bowyer K (2017) Gender-from-iris or gender-from-mascara? <http://arxiv.org/abs/1702.01304>
8. Kumar A, Passi A (2008) Comparison and combination of iris matchers for reliable personal identification. In: 2008 IEEE computer society conference on computer vision and pattern recognition workshops, CVPR Workshops, vol 43, pp 1016–1026. <https://doi.org/10.1109/CVPRW.2008.4563110>
9. Lagree S, Bowyer KW (2011) Predicting ethnicity and gender from iris texture
10. Majumdar J, Patil BS (2013) A comparative analysis of image fusion methods using texture. *Lecture notes in electrical engineering*, 221 LNEE, vol 1, pp 339–351. https://doi.org/10.1007/978-81-322-0997-3_31
11. Ramlee RA, Ranjit S (2009) Using iris recognition algorithm, detecting cholesterol presence. In: Proceedings—2009 international conference on information management and engineering, ICIME 2009, pp 714–717. <https://doi.org/10.1109/ICIME.2009.61>
12. Singh M, Nagpal S, Vatsa M, Singh R, Noore A, Majumdar A (2018) Gender and ethnicity classification of Iris images using deep class-encoder. In: IEEE international joint conference on biometrics, IJCB 2017, pp 666–673. <https://doi.org/10.1109/BTAS.2017.8272755>
13. Sreya KC, Jones BRS (2020) Gender prediction from iris recognition using artificial neural network (ANN). www.ijert.org
14. Tapia J, Aravena CC (2018) Gender classification from periocular NIR images using fusion of CNNs models. In: 2018 IEEE 4th international conference on identity, security, and behavior analysis, ISBA 2018, pp 1–6. <https://doi.org/10.1109/ISBA.2018.8311465>
15. Tapia JE, Perez CA (2019) Gender classification from NIR images by using quadrature encoding filters of the most relevant features. *IEEE Access* 7:29114–29127. <https://doi.org/10.1109/ACCESS.2019.2902470>
16. Tapia JE, Perez CA, Bowyer KW (2015) Gender classification from iris images using fusion of uniform local binary patterns. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, vol 8926, pp 751–763. https://doi.org/10.1007/978-3-319-16181-5_57
17. Tapia JE, Perez CA, Bowyer KW (2016) Gender classification from the same iris code used for recognition. *IEEE Trans Inf Forensics Secur* 11(8):1760–1770. <https://doi.org/10.1109/TIFS.2016.2550418>
18. Thomas V, Chawla NV, Bowyer KW, Flynn PJ (2007) Learning to predict gender from iris images

Hardware Implementation of an Activation Function for Neural Network Processor



Shilpa Mayannavar  and Uday Wali 

1 Introduction

Processor design is undergoing a grand upheaval. Large part of this evolution is due to the introduction of neural computation in virtually every field of daily life. Computational requirements of a neural network processor (NNP) are vastly different than the conventional processors. The workload of a NNP requires implementation of massively parallel compute cores. The network architecture tends to depend on the application area, and hence, there is a need to develop customized processors. Many companies are using their own processors to implement such hardware. For example, a special purpose processor for software defined radio and heuristic cognitive radio algorithms has been reported by Saha et al. in [1]. Use of graphic processor units (GPU) has shown significant performance advantages [2]. Many new neural architectures are being proposed to address domain-specific needs like robotic motion control [3], image recognition [4] etc. A domain-specific instruction set architecture (ISA) for neural accelerators, called Cambricon, is reported by Liu [5]. Its load-store architecture integrates scalar, vector, matrix, logical, data transfer and control instructions, based on a comprehensive analysis of existing neural networks (NN).

Recently, Intel has revealed a new processor called Nervana NNP capable of performing tensor operations at processor level [6]. Nervana NNP uses a fixed-point number format named flexpoint that supports a large dynamic range using a shared exponent. Mantissa is handled as a part of the op-code. IBM has developed a brain-like chip called TrueNorth, with 4096 processor cores, each capable of emulating 256 neurons with 256 synapses each. Neurons and synapses are two of the fundamental

S. Mayannavar (✉)

Nitte Meenakshi Institute of Technology, Yelahanka, Bangalore 560064, India

e-mail: mayannavar.shilpa@gmail.com

U. Wali

C-Quad Research, Desur IT Park, Belagavi 590014, India

biological building blocks that make up the human brain. Hence, the chip mimics one million human neurons and 256 million synapses [7]. DynamIQ is a new technology for ARM Cortex. It has dedicated processor instructions for artificial intelligence (AI) and machine learning (ML) with faster and more complex data processing. Wathan [8] has reported that dynamIQ performs AI computations with $50 \times$ increased performance compared to Cortex-A73 systems. It supports flexible computation with up to 8 cores on a single cluster with SoC, where each core can have different performance and power characteristics. Therefore, it is not difficult to foresee a situation where many companies would like to design their own processors to support deep learning neural network processors.

Implementation of sigmoid function using piecewise linear (PWL) [9] and second-order nonlinear function (SONF) and look-up table [10] have been reported. They have reported maximum error between software and hardware-based computation as 0.002. Similar work using PWL approximation is reported in [11].

In this paper, improved implementations using two-point (PWL) approximation and second-order interpolation (SOI) are reported. For PWL, maximum error of 0.0974 and 0.0238% was observed with uniform and non-uniform spacing. Further reduction in error, down to 0.0005%, has been observed using SOI. These modules use a specific number format suitable for implementing the activation function. The module has two modes of operation with user selectable speed and error behavior. Programmers will be able to shift between these modes depending on the end application. This work is part of an ongoing work on a design framework for development of special purpose processors [12].

2 Approximation of Sigmoid Function

One of the most frequently used operations in the neural networks is the activation function. It is possible to define various activation functions but the S-shaped sigmoid function is most widely used because of its gradient descent properties [13]. Sigmoid can be defined by the formula given below:

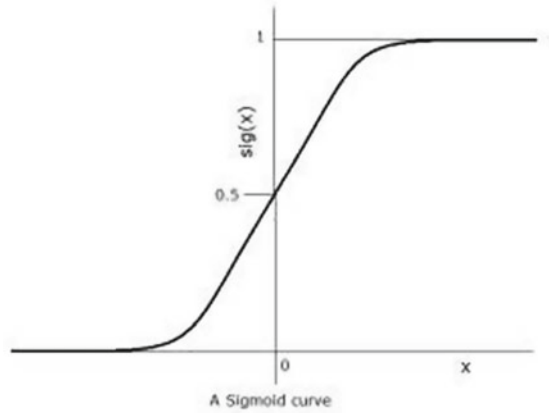
$$y = \frac{1}{1 + e^{-x}} \quad (1)$$

Expanding the equation with Taylor series, we get

$$y = \frac{1}{1 + 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \frac{x^4}{4!} - \dots} \quad (2)$$

Therefore, computing the sigmoid function directly is computationally intensive. Using approximations can ease the computational overhead, especially in multi-layer networks (Fig. 1).

Fig. 1 Sigmoid curve



We have proposed two methods of approximating the sigmoid function, viz. two-point piecewise linear (PWL) and three-point polynomial approximation (SOI). Since the sigmoid function is symmetric, only the first quadrant of the $(x-y)$ plane is considered.

2.1 Two Point Piece-Wise Linear Approximation (PWL)

Consider the $x-y$ plane as shown in Fig. 2. Points (x_1, y_1) and (x_2, y_2) are the two known points on the curve. The sigmoid of x can be interpreted using two-point approximation as given in (3).

$$y = \left(\frac{y_2 - y_1}{x_2 - x_1} \right) (x - x_1) + y_1$$

$$y = m(x - x_1) + y_1 \tag{3}$$

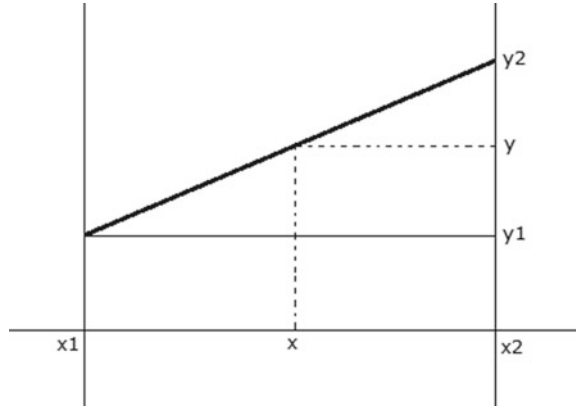
The stored values of (x_1, y_1) and m are obtained from look-up table such that given value of x lies between two sets of (x, y) . This approximation may be carried out with both uniform and non-uniform spacing between two set of points.

Uniform spacing of 0.5

The two known points are assumed to be at uniform distance of 0.5. For example, sigmoid of x between $x_1 = 0$ and $x_2 = 0.5$ is depicted in Fig. 3. This has maximum error deviation of 0.3838%. Note that Fig. 3 shows error scaled up by 100 for clarity.

By decreasing the interval between points, we can improve the accuracy. For example, Fig. 4 shows two-point approximation with uniform spacing of 0.25. This

Fig. 2 Two-point approximation of a curve



Sigmoid Function Appx. using PWL with Uniform distance of 0.5

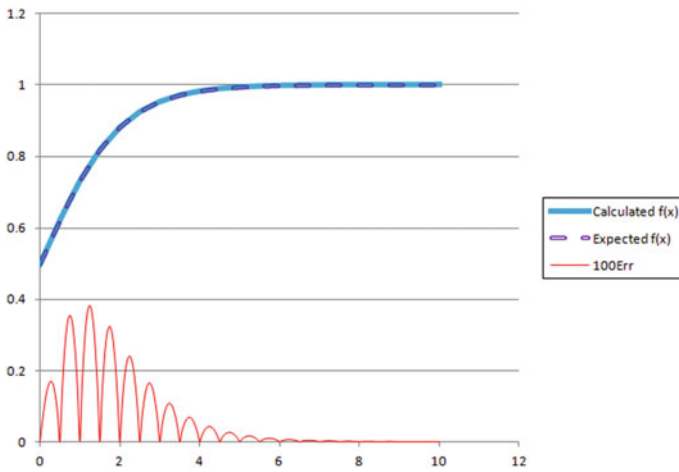


Fig. 3 Two-point approximation of a curve with uniform distance of 0.5

has maximum error deviation of 0.0974%, an improvement of nearly 75%. Here also error is scaled by 100 for clarity.

Uniform spacing of 0.25

By decreasing the interval between points, we can improve the accuracy. For example, Fig. 4 shows two-point approximation with uniform spacing of 0.25. This has maximum error deviation of 0.0974%, an improvement of nearly 75%. Here also error is scaled by 100 for clarity.

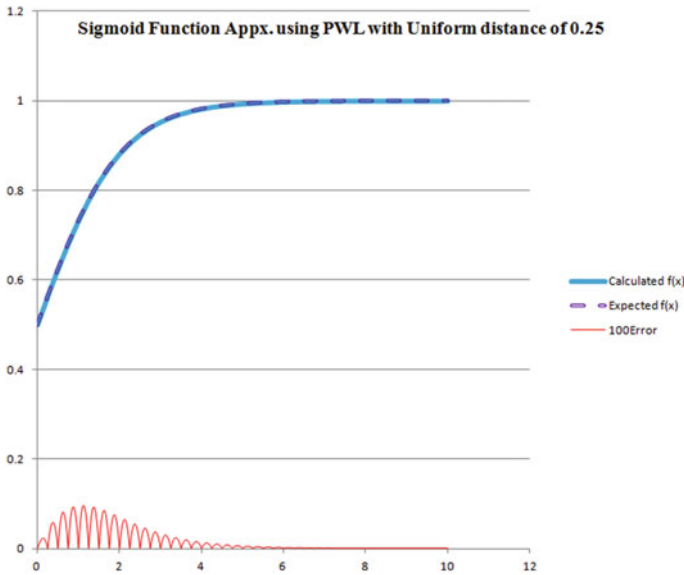


Fig. 4 Two-point approximation of a curve with uniform distance of 0.25

Non-uniform spacing

From Figs. 3 and 4, it is observed that the gradient of sigmoid curve is significant for $|x| < 6$, and therefore, a non-uniform spacing gives better approximation with less number of computations compared to uniform spacing. Figure 5 shows the approximation of a curve with non-uniform spacing. Maximum error of 0.0238% is noted.

Look-Up Table format

From Eq. 3, it is clear that we will need three variables to be stored per point. If we store them as a tuple $\{x, y, m\}$, sequentially, and assuming 16-bit format suggested in Sect. 3.1, we will need fairly small memory. For a 15-point approximation, we will need $16 \times 3 \times 15 = 720$ bits or 90 bytes only. Reducing the number of points in LUT reduces the memory requirement but affects the accuracy.

2.2 Three-Point Second-Order Interpolation (SOI)

The approximation method using two-point approximation has maximum error of 0.0238%. This error can be further reduced by using three-point approximation.

The three-point approximation method is shown in Fig. 6. There are three known points $(x_1, y_1), (x_2, y_2), (x_3, y_3)$, and using these points, the sigmoid of x can be computed.

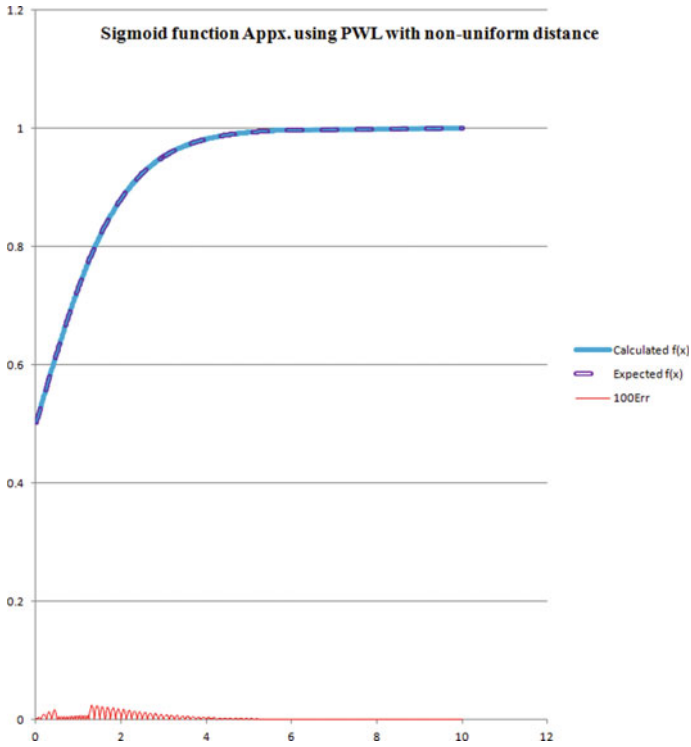
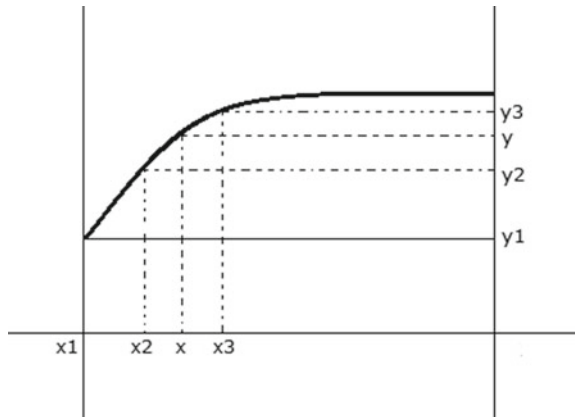


Fig. 5 Two-point approximation of a curve with non-uniform distance

Fig. 6 Three-point approximation of a curve



The three point approximation is done using the quadratic equation $y = ax^2 + bx + c = (ax + b)x + c$. The method uses three points in x - y plane, viz. (x_1, y_1) , (x_2, y_2) and (x_3, y_3) . These three points are assumed to be known and stored in the memory and are used to interpret the sigmoid of given x . There are three equations for these three known points as given in (4)–(6).

$$y_1 = ax_1^2 + bx_1 + c \quad (4)$$

$$y_2 = ax_2^2 + bx_2 + c \quad (5)$$

$$y_3 = ax_3^2 + bx_3 + c \quad (6)$$

The coefficients a , b and c are calculated by solving the Eqs. (4)–(6), simultaneously. The simplified equations for a , b and c are given in (7)–(9).

$$a = \frac{((x_2 - x_1)(y_3 - y_1)) - ((x_3 - x_1)(y_2 - y_1))}{(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)} \quad (7)$$

$$b = \frac{(y_2 - y_1) - a(x_2^2 - x_1^2)}{(x_2 - x_1)} \quad (8)$$

$$c = y_1 - ax_1^2 - bx_1 \quad (9)$$

$$y = (ax + b)x + c \quad (10)$$

The pre-computed values of a , b and c are loaded from the look-up table, and the sigmoid of any given x is computed using Eq. (10). This method is carried out for uniform spacing between $x = 0$ and $x = 6$. The sigmoid value is assumed to be 1 for $x \geq 6$. This method has maximum error of 0.0030%. The graph of SOI with uniform spacing of 0.125 is shown in Fig. 7. For visibility purpose, the error is multiplied by 10,000.

In Fig. 7, we can see that for the values of x greater than 6, curve remains at 1, which means that the gradient of curve is significant for the values less than 6, and hence, there is no need for any calculation for $x \geq 6$.

As we can see from Fig. 7, the error is maximum at $x = 0.05$. This can be reduced by decreasing the interval between 0 and 1–0.0625 and by keeping interval outside this range at 0.125. This non-uniform spacing reduces the error down to 0.00057% as shown in Fig. 8.

The error deviation is calculated using the formula given in Eq. (11) below.

$$\%Error = \frac{|\text{Interpolated value} - \text{Theoretical value}|}{\text{Theoretical value}} * 100 \quad (11)$$

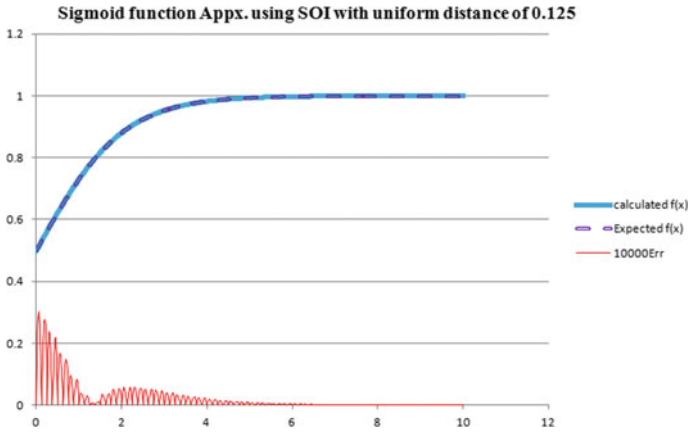


Fig. 7 Three-point approximation with uniform spacing of 0.125

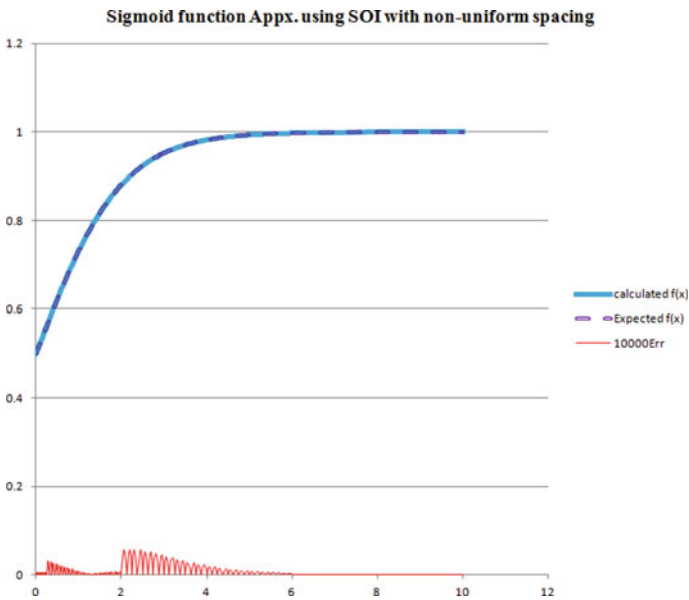


Fig. 8 Three-point approximation with non-uniform spacing

The above-described method of interpreting the sigmoid of a given x has been implemented using Verilog hardware description language (HDL). The method of implementation is explained with the help of finite state machine (FSM).

3 Approximation of Sigmoid Function

The basic operations required to implement the sigmoid functions are addition and multiplication. The algorithm to calculate the sigmoid function is explained in Table 1. Input x is compared with the known values of (x_1, y_1) , (x_2, y_2) and (x_3, y_3) . The corresponding a, b and c values are loaded from the look-up table which are used to evaluate the sigmoid.

3.1 Algorithm to Compute Sigmoid of Given X

```

Step1: On load = 1, load the input to the internal register
Step2: On start = 1, start the linear search and obtain the a, b, c
from the look up table, load these values into the registers. Go to
step 3
Step3: start the multiplication to get a*x, increment the mCount to
1, go to step 4
Step4: add the result from step 3 with b and increment the aCount to
1. Go to step 5
Step5: multiply the result of step 4 with x, increment the mCount to
2. Go to step 6
Step6: add the result from step 5 with c and increment the aCount to
2, got to step 7
Step7: exit.
    
```

The basic idea of implementing the sigmoid function is explained with the help of finite state machine as shown in Fig. 9.

The FSM shown in Fig. 9 has six states, viz. *idle*, *start*, *linear search*, *multiplier*, *adder* and *exit*. In the *idle* state, if there is a *load* command, then input is loaded to the internal register and the state changes to *linear search*. In the *linear search* state, the input x is compared with stored (x, y) values and when the x matches with one of the stored set and the corresponding a, b and c values are loaded from the memory and state will change to multiplier state. There are two multiplication and two addition steps involved. In the first step of multiplication, ax is computed, and in the first step of addition, $(ax + b)$ is computed. In the second step of multiplication, $(ax + b)x$ is

Table 1 Analysis of error due to optimized bit format-PWL method

x	y_a (theoretical)	y_i (interpolated)	y_o (optimized)	Error _i %	Error _o %
0.0125	0.503124959	0.503120937	0.503088474	0.000799	0.007252
0.025	0.506249674	0.506241875	0.506176948	0.001541	0.014366
0.0375	0.509373902	0.509362812	0.509265423	0.002177	0.021297
0.05	0.512497396	0.512483749	0.512353875	0.002663	0.028004
0.0625	0.515619916	0.515604687	0.51550293	0.002954	0.022688

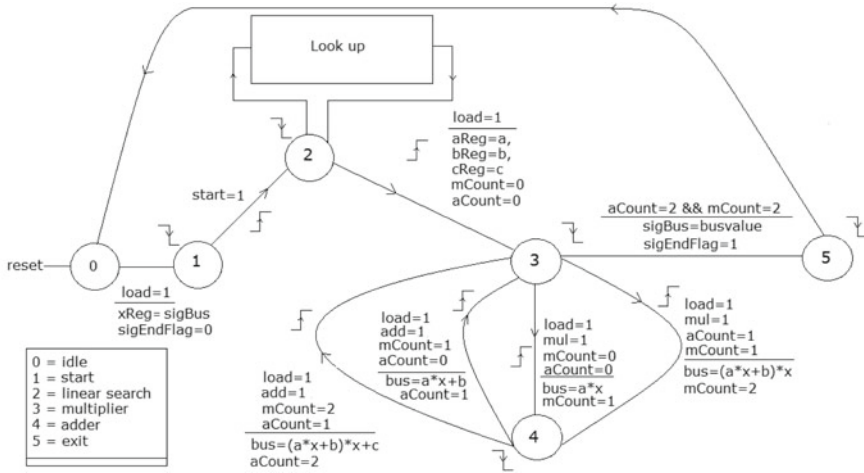


Fig. 9 FSM for calculating sigmoid function

computed. Finally, in the last step of addition, $(ax + b)x + c$ is computed and the process terminates.

Optimized bit format

The neural network structure deals with real-time inputs, and therefore, there is need to implement a floating-point arithmetic unit. In order to get the precision up to 4 significant figures, we have fixed 12 bits for the fraction as shown in Fig. 10. For example, the number 0.0125 is represented in the optimized bit format as 0000000000110011_b.

Errors due to Optimized bit format

The optimized bit format has 12 bits to represent the fraction part of floating-point number. Due to this restriction, there will be small amount of numerical errors in the calculated result. Tables 1 and 2 summarize the errors due to optimized bit format for both two-point and three-point approximation methods.

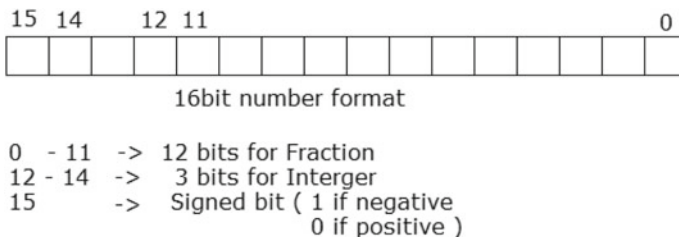


Fig. 10 Optimized bit format for 16-bit number

Table 2 Analysis of error due to optimized bit format-SOI method

x	y_a (theoretical)	y_i (interpreted)	y_o (optimized)	Error _i %	Error _o %
0.0125	0.503124959	0.503131838	0.503119714	0.001367	0.001443
0.025	0.506249674	0.506261254	0.506233052	0.002287	0.003284
0.0375	0.509373902	0.509388248	0.509346058	0.002816	0.005466
0.05	0.512497396	0.512512819	0.512456695	0.003009	0.007942
0.0625	0.515619916	0.515634967	0.515625954	0.002919	0.001171

Table 3 Comparison between actual and proposed methods

Method	Operations				
	Lookup	Add	Sub	Mul.	Div.
Taylor Exp.	0	6	6	65	10
PWL	3	1	1	1	0
SOI	3	2	0	2	0

4 Justification of the Novel Approach

It can be concluded from Tables 2 and 3 that the computational accuracy can be controlled by selecting the interval. Increasing the number of intervals improves accuracy but also increases the size of LUT, providing a trade-off between the two. However, using non-uniform spacing of points, this problem can be overcome to a certain extent. The other aspect of calculation of the activation function is the speed with which computations can be carried out. Table 4 shows that the number of mathematical operations reduces considerably for interpolation methods over direct computation. Therefore, the time required for the computation is considerably reduced.

5 Conclusion and Future Work

The data presented in this paper shows that interpolation methods can be used to achieve sufficiently accurate computation of the activation function. The computational complexity is also considerably reduced for proposed approximations. Therefore, we could implement hardware modules using the methods discussed in the above sections. The SOI approximation is implemented using Verilog. With a serial multiplier, maximum of 30 clock cycles are required to compute the sigmoid function. Therefore, these methods can be used to achieve better performance with reduced computational complexity.

There are several other types of nonlinear functions used in deep neural networks. For example auto resonance networks, radial bias functions, self-optimizing maps, etc., use other nonlinear functions which can be implemented using similar interpolation methods.

Acknowledgements The authors would sincerely like to thank C-Quad Research, Belagavi, for all the help and support.

References

1. Saha A et al (2010) Mechanism for efficient implementation of software pipelined loops in VLIW processors. USPTO Application #20100211762, Saankhya Labs
2. Yang Y, Li C, Feng M, Chakradhar S (2016) Memory efficiency for convolutional neural networks operating on graphics processing units. USPTO Application #20160342888, NEC Laboratories America Inc.
3. Aparanji VM, Wali U, Aparna R (2016) A novel neural network structure for motion control in joints. In: ICECCOT—2016. IEEEExplore digital library, pp 227–232, document no. 7955220
4. Jia Y, Shelhammer E, Donahue J, Karayev S et al (2014) Caffe, convolutional architecture for fast feature embedding. Cornell University Archives. arXiv:1408.5093v1[cs.CV] 20 Jun 2014
5. Liu S (2016) Cambricon: an instruction set architecture for neural networks. In: ACM/IEEE 43rd annual international symposium on computer architecture
6. Rao N (2017) Intel® Nervana™ neural network processors (NNP) redefine AI silicon. 17 Oct 2017. Available online: <https://www.intelnervana.com/intel-nervana-neural-network-processors-nnp-redefine-ai-silicon/>
7. Hernandez D (2014) IBM unveils a ‘brain-like’ chip with 4000 processor cores. July 2014. Available online: <https://www.wired.com/2014/08/ibm-unveils-a-brain-like-chip-with-4000-processor-cores/>
8. Wathan G (2017) ARM dynamIQ—technology for the next era of compute. 21 March 2017. Processors blog, ARM Community. Available online: <https://community.arm.com/processors/b/blog/posts/arm-dynamiq-technology-for-the-next-era-of-compute>
9. Saichand V et al (2008) FPGA realization of activation function for artificial neural networks. In: Eighth international conference on intelligent systems design and applications. IEEE
10. Ngah S et al (2016) Two-steps implementation of sigmoid function for artificial neural network in FPGA. ARPN J Eng Appl Sci 11(7)
11. Tisan A et al (2009) Digital implementation of the sigmoid function for FPGA circuits. ACTA Technica NAPOCENSIS Electron Telecommun 50(2)
12. Mayannavar S, Wali UV (2016) Design of modular processor framework. Int J Technol Sci (IJTS) IX(1):36–39. ISSN (Online) 2350-1111, 2350-1103 (Print)
13. Ruder S (2017) An overview of gradient descent optimization algorithms. Cornell University Library, June 2017, Available online: <https://arxiv.org/pdf/1609.04747.pdf>

Using Big Data and Gamification to Incentivize Sustainable Urban Transportation



A. Mariyan Richard and Prasad N. Hamsavath

1 Introduction

Creating an environment that facilitates and promotes sustainable mobility policies is the need of the hour for most of the cities across the planet. In metropolitan cities, a sustainable urban mobility has created a lot of possibilities for itself for being one of the important dimensions in a smart city [1]. Leveraging more specific information and communication technology to arrive at serving an objective is a task in itself, yet a great way to make an impact that makes the city, its citizens, and the institutions that govern them smarter.

In this paper, we bring out an idea that describes the Gamification affordances developed and report the findings related to the SUM possibilities. The obtained results clearly depict the significant effect of Gamification in being a pillar for Voluntary Travel Behavior Change Community, and it is also observed how a gamified framework can play a vital role in sustainable behaviors and policies being adapted to make a complex sociotechnical system [2].

2 Literature Review

The research and development that pertain to sustainable transportation are to eventually uphold the self-explanatory term sustainable transportation. Without it, we

A. Mariyan Richard (✉) · P. N. Hamsavath
Department of Masters of Computer Applications, NITTE Meenakshi Institute of Technology,
Bangalore, India
e-mail: mariyanrich01@gmail.com

P. N. Hamsavath
e-mail: naikphd@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_52

609

would not know where to begin, and also miserably fail at spreading the word about it and convincing people to pursue the same. To be specific, if the higher ups do not know what the proposed transportation model is clearly, it is near to impossible for them to go about spreading awareness about it.

This additionally leads to the issue of not having a concrete structure based on which other policies and programs would be created around [3]. There have always been a lot of research programs on sustainable development, but nothing specifically to and exclusive market such as transportation. Many entities have managed to adopt the foresaid definition as their own at various commissions. It provides more information on the existing research and development that has gone into the system by various means. All prototypes have been designed and developed based on partial information, and no concrete references have been adhered to simply because there does not exist one. For each review, multiple definitions were observed and arrived upon. But now, with a better understanding of the scenario, a rough yet stable discussion is in place [4].

Gamification can be termed as the use of simple game design elements and principles in non-gaming environment. It is based on the simple flow of data that is a thought-driven operation which requires the person engaged in an activity to be completely immersed into the process of doing the said activity. This is usually adopted to increase and improve user engagement, productivity, control over activity, easy to understand, and other organizational criteria [5]. A lot of research has been carried out on this and was observed to have its positive effects on users. It is also said to help improve one's ability to process digital content and analyze the area of study, respectively.

Sustainable transportation models should include the following phenomena that help understand the proposed model better:

- Exhaustion of natural resources
- Atmospheric impacts
- Threat to mankind
- Air quality index
- Space constraints
- Equality.

Collectively, from all the work above, it is practically still not possible to define sustainable transportation. This also leads us to the following observations:

- The concept of sustainable transportation is based on a sustainable development approach.
- Sustainable transportation is a balance of multiple entities. At its simplest, ST should be a direct contributor to the local economy of any given place.
- To help define better, it is required by the systems to somehow manage to help understand what sustainable and unsustainable really equate to in terms of numbers. A concrete model for the same has to be defined mathematically to be even adopted.

- The system should emphasize more on the reforms, governing body, interdependence on other factors and sectors in the society.
- The absence of a definition should not stop people from promoting it.

3 Methodology

The gamification elements and technique are something that intend to bring out peoples' natural instincts to socialize, learn, master, compete, achieve, status, self-expressions, observations, and closure. First used gamification policies and strategies used rewards for users who managed to accomplish intended tasks or compete to ensure player engagement. The various type of rewards that were offered were points, badges, levels, progress bar-based status, or allowing players to redeem coupons or virtual currency. Allowing players to view the rewards and accomplished tasks of other players induces the competitiveness on one to outdo the other instantly [6]. These led to dynamic leaderboards and other UI-based frameworks to encourage more and more players to use the system. There have also been instances of clash between two or more players during the use of these frameworks, which are a result of unethical behavior on the players' part, not sportive enough to understand the motive at hand, or simply something as simple as disadvantages in form as demographics such as a woman player. Best-practice gamification designs try to refrain from using this element [7] (Fig. 1).

Another way to using gamification is to make the existing operations and tasks feel more like reward-based games. Some of the examples where this approach can be integrated are onboarding of employees virtually, increasing the context screen engagement time, quick narratives, and additional choices.

Creating new ways of harnessing data as well as giving back value is the only way we can understand the needs of users better. In transportation industry, the vast majority of the participants are end users whose objective is to move from point *A* to point *B*, following which there are much more complex needs present for organizations and their operations [8]. There is a distinction between how the data can be used to create more convenience to the users and also to understand their needs at the same time.



Fig. 1 Conceptual gamification procedure applied to transport use

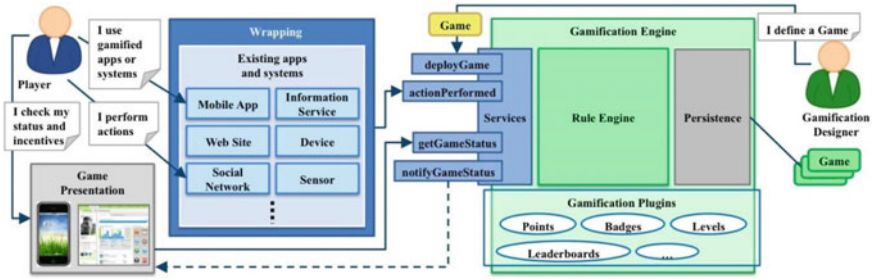


Fig. 2 Architecture of the gamification framework

In the business-to-consumer segment, one of the important things is cost and fuel optimization, something that every end consumer really wants as they want to save money. That is something that data can be extensively used for: to detect driving patterns, for example, to suggest more fuel-optimized routes and greener ways of traveling. Those are all sort of key values where big data can be used, and today, it is totally doable and possible.

The future of efficient transportation is highly multi-modal and highly multi-dimensional. As we all know, multi-modal means using many means of transport [9]. This has been in existence for a few decades now. But, multi-dimensional is when you change the mode of transport, you also change the user experience, the data flow and pricing structures, and there are so many variations that happen when you change modes.

This multi-dimensional transport application is going to be the next big thing in the transport industry. The efficiency is one side of the coin, and the other side of the coin is user experience. People want to choose the most interesting way of traveling [10]. It does not always need to be the most convenient or the cheapest. Of course, cost is a high criterion, but not the only one—people also want a comfortable journey, they want a nice experience, and most importantly, people want to be engaged (Fig. 2).

The general form of a notification is $\langle gamifiableActionID, playerID, timeStamp, parametersMap \rangle$ where *gamifiableActionID* is a unique ID and the *parametersMap* contains a set of key/value pairs that are specific to that gamifiable action [11]. The wrapping layer issues notifications on behalf of the wrapped information and communications technology systems through a simple *actionPerformed* service interface. Moreover, the wrapping layer enables strongly decoupled interactions between the native Smart City functionalities involved in a specific game which is the component responsible to execute that game and managing its status.

4 Future Enhancements

The proposed mode could be used to benchmark and evaluate transportation sustainability within the general existing framework. Since integrating sustainability in itself is self-explanatory, guides about the same for transport planning could be formed. Various options for inter-state departmental collaborations could give raise to much more feedback-oriented development. Efforts to promote some aspects of non-urban transportation too could pave the way for last mile connectivity. Lastly, a one-stop portal to provide development and sustainable transportation could be deployed [12].

5 Conclusion

Demonstration of the scope of gamification that can be used to create awareness about a complete overhaul in urban mobility that is in terms with sustainability and voluntary travel behavior change was successful. To be precise, the addition of a game-controlled environment has led to great increase in the reliability of the users/participants on the information and communications technology mobility services that were available to them. Also, the acceptance rate of recommendations which were provided to those services to help generate new options for daily commute has caused a noticeable change in terms of how SUT can be adapted. A more generic-oriented service framework could be designed and developed in the future to help apply gamification to sustainable transportation polices.

References

1. How will big data impact the future of transportation?—Part 3 by Vinay Venkatraman. <https://www.move-forward.com/how-will-big-data-impact-the-future-of-transportation-part-3/>
2. The role of gamification and big data in today's world of business. Brigg Patten. https://www.hr.com/en/app/blog/2016/05/the-role-of-gamification-and-big-data-in-todays-wo_inuulx6g.html
3. Kazhamiakin R, Marconi A, Perillo M, Pistore M, Piras L, Avesani F, Perri N (2015) Using gamification to incentivize sustainable urban mobility. Trento, Italy 38123
4. Zhou J (2012) Sustainable transportation in the US: a review of proposals, policies, and programs since 2000. *Frontiers Architect Res* 1:150–165. <https://doi.org/10.1016/j.foar.2012.02.012>
5. Steg L (2007) Sustainable transportation. *IATSS Res* 31:58–66. [https://doi.org/10.1016/S0386-1112\(14\)60223-5](https://doi.org/10.1016/S0386-1112(14)60223-5)
6. Bamwesigye D, Hlavackova P (2019) Analysis of sustainable transport for smart cities. *Sustainability* 11. <https://doi.org/10.3390/su11072140>
7. Bran F, Burlacu S, Alpoci C (2018) Urban transport of passengers in large urban agglomerations and sustainable development. Experience of Bucharest municipality in Romania. *Eur J Sustain Dev* 7. <https://doi.org/10.14207/ejsd.2018.v7n3p265>
8. https://www.researchgate.net/publication/281377423_Using_Gamification_to_Incentivize_Sustainable_Urban_Mobility

9. Giffinger R, Haindlmaier G, Kramar H (2010) The role of rankings in growing city competition. *Urban Res Pract* 3(3):299–312
10. Nam T, Pardo T (2011) Conceptualizing smart city with dimensions of technology people and institutions. In: *Proceedings of the 12th annual international digital government research conference: digital government innovation in challenging times*, pp 282–291
11. Merugu D, Prabhakar B, Rama N (2009) An incentive mechanism for decongesting the roads: a pilot program in Bangalore. In: *Proceedings of ACM NetEcon workshop 2009*
12. Gabrielli S, Maimone R, Forbes P, Masthoff J, Wells S, Primerano L et al (2013) Designing motivational features for sustainable urban mobility. In: *CHI'13 extended abstracts on human factors in computing systems*, pp 1461–1466

Optical Character Recognition System of Telugu Language Characters Using Convolutional Neural Networks



K. V. Charan and T. C. Pramod

1 Introduction

Handwriting character recognition is defined as identifying the characters from the text images. This process is difficult as handwriting varies from one person to another. Compared to handwritten character recognition, accuracy is higher for printed character recognition. The recognition method is classified into two types: offline and online. In online, the main task is to identify the characters while the user writes by capturing dynamic handwriting information which includes direction, stroke speed, and length. Offline character recognition is admitted after completion of writing. Techniques like Bayesian classifier, Neural Networks, Support Vector Machines, Hybrid Markov models, and K-Nearest Neighbor Classifier can be used for recognition. However, their accuracies differ from each other. Convolutional Neural Network is one of the most well-known deep learning approaches which is used for many classification problems. This method is capable of finding patterns in 2-D data and is applicable in the fields like classification of images and face recognition. Feature extraction is implicitly done within the CNN. Recognition of characters is very difficult as there is variety of writing styles and slants, specially in a language like Telugu which has composite structure and very similar type of character set.

India is a diversified country. Each state in India has its unique culture of language and fashion as well. Mainly, there are 16 major languages and 100 regional languages. Telugu is one of the languages spoken in the southern part of India. Telugu is

K. V. Charan (✉)

Department of Computer Science and Engineering, Shridevi Institute of Engineering and Technology, Tumkur, Karnataka, India
e-mail: charanssit@gmail.com

T. C. Pramod

Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumkur, Karnataka, India

Fig. 1 Telugu characters



spoken in many states like Andhra Pradesh, Telangana (official language), some parts of Puducherry, and Andaman and Nicobar Islands. It is also one of the six languages nominated as classical language declared by Indian government. The Telugu language consists of 52 letters (shown in Fig. 1). It has 16 vowels and 36 consonants. There are numerous documents that contain both Telugu and English jointly. Some of the documents are acknowledgment of Aadhar card, reports like death and birth, reservation forms of railways, etc.

2 Deep Learning Techniques

Deep Learning is a procedure of Data Mining that utilizes deep neural network architecture, which are particular sorts of machine learning and AI algorithms. There are three procedures that permit deep learning in resolving different problems.

- Fully Connected Neural Networks
- Convolutional Neural Networks
- Recurrent Neural Networks.

2.1 Fully Connected Neural Networks

These neural networks are standard networks utilized in various applications. Fully connected means every neuron in the past layer is associated with each neuron in the ensuing layer. Feed forward implies that neurons in the former layer are simply associated with the neurons in a succeeding layer. Every neuron consists of an activation function that changes the yield of a neuron. It is possible to make a system with various inputs, different outputs, distinct hidden layers, and separate activation functions. Different combinations of these networks permit us to make an incredible network that solves a wide range of problems.

2.2 Convolutional Neural Networks

CNN is a sort of deep neural network architecture intended for explicit works like the classification of images. In this process, extracting the features is done implicitly within the network. CNN is used for various tasks like image processing, segmentation, recognition, and natural language processing.

2.3 Recurrent Neural Networks

RNNs work viably on successions of data with variable input length which implies RNNs utilize information on its previous state as a commitment for its current prediction, and we can repeat this procedure for an optional number of times permitting the network to propagate data by methods for its hidden state through time. With this feature, RNNs become extremely effective to work with sequences of data that happen after some time. It functions well for applications that require a series of information, which changes after some time. Some of them are language translation, recognition of speech, etc.

In this paper, we are using deep learning techniques like Convolutional Neural Networks for recognizing Telugu characters. The rest of the paper is organized as follows: Sect. 3 gives the related work. Proposed solution is discussed in Sect. 4. Experimental results are discussed in Sect. 5, and conclusion is given in Sect. 6.

3 Related Work

Sahara and Dhok [1] proposed a process for recognition and segmentation of characters like Devanagari and Latin languages. For character segmentation, heuristic-based algorithm is integrated with Support Vector Machines. For Character recognition, K-Nearest Neighbor classifier is used for recognizing input character. It is shown that the accuracy of 98.86% is procured for segmentation and 99.84% for recognition process.

Sharma et al. [2] put forward an algorithm deployed on the idea of learning by itself. In this paper, numbers and English letters are taken for recognition analysis. SVM classifies the given input into different classes by using hyper planes, and an optimal hyper-plane is selected among multiple hyper planes. Both multi-class and bi-class are used to accomplish the identification of characters with 95.23% accuracy.

Jabir Ali and Joseph [3] presented a model of Convolutional Neural Networks for categorizing Malayalam handwritten characters. It involved image acquisition, greyscale transformation, word decomposition, binarization, character segmentation, and estimation. CNN is used for the classification of Malayalam characters. Accuracy of 97.26% is obtained in examining the CNN model.

Dara and Panduga [4] described a method for offline handwritten character recognition by extracting features using 2D Fast Fourier Transform and using SVM for documents containing Telugu characters. A complete class of 750, and 1500 samples are used for training, and for developing testing area, 750 trails are used. The accuracy obtained with this method is 71%.

Angadi et al. [5] suggested and evaluated a classic CNN for online Telugu character recognition. In this process, a character image is classified into 166 classes contained in 270 trails. The network includes 4 standard layers, first layer with 5×5 and remaining with 3×3 kernels and ReLU activation functions, followed by 2 impenetrable layers with function of SoftMax. The result of this method is quite imposing compared to other algorithms. This technique provided 92.4% accuracy.

Inuganti and Ramisetty [6] has given an analysis of datasets, feature extraction techniques, and classification of various Indian scripts like Assamese, Kannada, Telugu, Gurumukhi, Tamil, Bangla, and Devanagari. The steps involved in this process are data collection, pre-processing of images, extraction of features, identification, and post-processing of images. Classification is done using Structure-based Models, Neural network models, motor models, and Statistical Models. In post-processing, confusing pairs are found and Script-specific feature is used to solve the ambiguities in the characters. Accuracy of 92% is disclosed over the collection of datasets from 168 users.

Manjunathachari et al. [7] described the morphological analysis for the classification of Telugu Handwritten composite characters. A dataset is created by writing a composite character on paper and scanning it using a scanner. 250 samples of each character are taken. Segmentation of characters is done using morphological operation and attained a 98.1% accuracy rate.

Mohana Lakshmi et al. [8] presented a coherent algorithm to recognize handwritten Telugu characters based on Histogram of Oriented Gradients (HOG) features and classification using the Bayesian Classifier. Binarization and smoothing are the two techniques used to improve the resultant picture. Based on the number of negative and positive testing and training dataset, the recognition rate was calculated. This technique provided 87.5% accuracy.

Chakradhar et al. [9] suggested diverse methods to identify Telugu handwritten characters. The authors have analyzed some of the existing systems for recognition of handwritten characters of Telugu scripts. It shows that the accuracy rate obtained using Hybrid Models is 93.1% and using SVM is 90.55%.

Kaur and Kaur [10] analyzed and reviewed many technologies to seek out characters from input images containing text characters. The process contains the following phases: Image Scanning, Pre-processing, Extraction of features, Recognition, and post-processing. In pre-processing, the following steps are processed: noise removal, thresholding, and skeletonization. For classification, the techniques used are Thomas Bayes classifier, Support Vector Machines, Neural Networks, and Nearest Neighbor classifier. In post-processing, symbols are grouped.

Fig. 2 Vowels**Fig. 3** Consonants

4 Proposed Work

The proposed system utilizes CNN framework to classify the characters. The system mainly comprises five steps:

- Data acquisition
- Defining CNN architecture
- Training the network
- Deploy the model
- Testing the network.

4.1 Data Acquisition

Creating a dataset requires lot of time and needs lot of efforts. There is no accessible dataset for Telugu characters. We have collected the dataset from particular school students and modified it. This dataset contains images of 52 Telugu characters in which 18 characters are vowels and remaining consonants. Among them, only 35 consonants and 13 vowels are used commonly. For training the model, enormous dataset is required. Each character is written on a paper in a particular order, in various designs and sizes by 60 individual writers. Then, these documents were scanned by scanner. Each character is kept in separate folder. Each folder is labeled with different names. Some of the vowels and consonants are shown in Figs. 2 and 3.

4.2 Defining CNN Architecture

The architecture mostly consists of three layers such as convolutional layers, output layer, and pooling layer. There are various CNN models which are utilized in numerous classification tasks. Some of the architectures are LeNet, VGGNet, AlexNet, Inception, etc. These architectures vary from each other because of contrast in the quantity of hyper parameters of the system. The architecture used in this paper is LeNet. The architecture of LeNet is shown in Fig. 4.

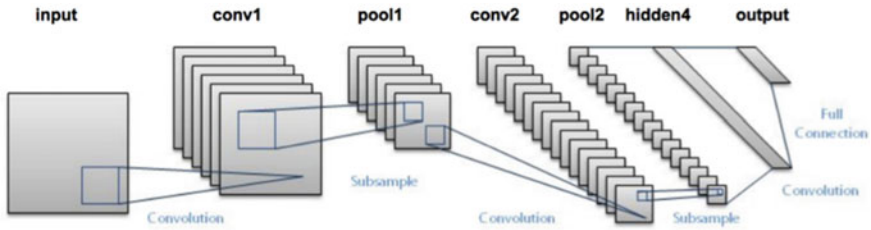


Fig. 4 LeNet architecture

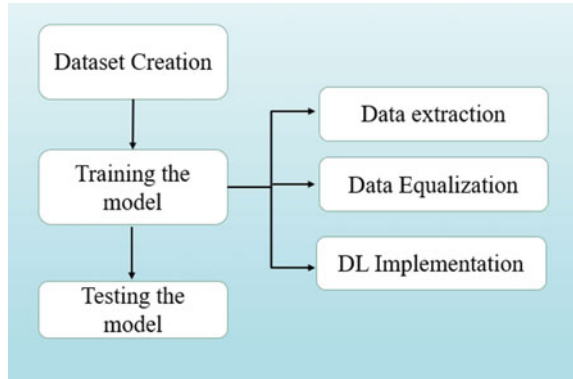
The architecture contains six layers. It contains two groups of activation, convolutional layer, and pooling layers followed by fully connected layer, at last SoftMax classifier. It has convolutional filters with each filter size of 5×5 . Then, ReLu activation function is applied and followed by 2×2 max-pooling. Adam Optimizer is used for gradient descent algorithm which accomplishes in training. An individual image of Telugu character with label is passed throughout the layers and, to gradient descent algorithm, then weights get updated. Output layer consists of 44 classes for each character in the dataset.

4.3 Training the Network

Firstly, the network needs to be prepared by training with the created dataset. Then, it is supposed to label each character in the dataset and labels format is prepared such that the network can understand them. The created dataset is prepared, and each character is labeled. This dataset is separated into training set and testing dataset in the ratio of 75:25. The training dataset is again separated into the training and validating sets. This validation set is utilized to test during training time that we can see whether our model overfits or not during every epoch. An epoch means complete iteration of training over the whole dataset. After the model is well trained, the weights are stored. Algorithm for training the network is as follows:

1. Input image.
2. Resize the image to 28×28 pixels.
3. The image is converted into an array of 2-D as CNN takes input as 2-D array.
4. Extract the label of the class from the image path and update it to the list.
5. Change the intensities of raw pixels to the range of $[0, 1]$.
6. Split the data into training dataset and testing dataset as 75% and 25%, respectively.
7. Convert the labels of integers into vectors.
8. Construct a generator for the images used during data augmentation.
9. Save the network.

Fig. 5 Flowchart of proposed method



4.4 Deploy the Model

After effective training, the model must be deployed so that the user can input an image of Telugu handwritten scripts and is predicted using the model which is saved. Telugu manually written character image has to be read to predict the output. For every predicted class, each character is mapped to equivalent Telugu character.

4.5 Testing the Network

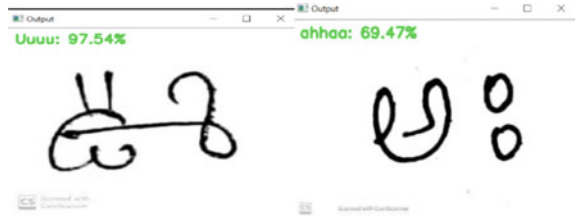
After model deployment, the model is ready for testing. 20% of the total dataset is considered for testing. First the image is loaded from the dataset. If the image does not load, then it displays error message and get exits. After loading the image, it is resized to 28×28 pixels. The image is converted to binary image and then converted into arrays. The network then finds the matching character and gives the accuracy of each character. The algorithm for testing is given below:

1. Load the image.
2. If there is error in loading the image, it exits and displays the error message.
3. Pre-process the image for classification, i.e., resized to 28×28 and converting image into binary image and converting into arrays.
4. Load the trained convolutional neural network.
5. Classify the input image (Fig. 5).

5 Experimental Results

This section addresses results that are obtained after performing convolutional neural network technique for the dataset on handwritten Telugu characters. By this algorithm, we have achieved the accuracy of 90%. The accuracy can be increased by

Fig. 6 Results of characters with accuracy rates

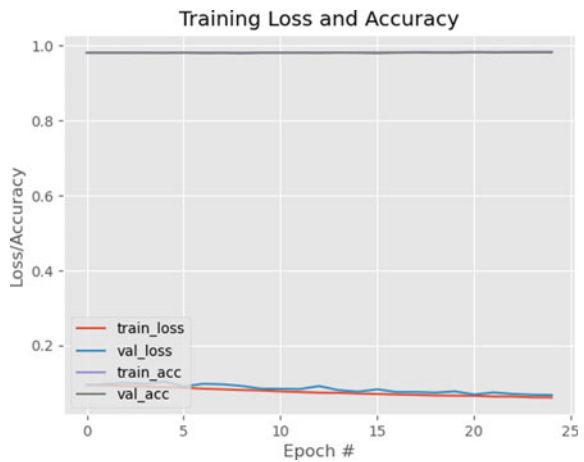


increasing the dataset. It also plots a graph for each character that is recognized. The graph tells accuracy of the character that is trained (Figs. 6 and 7). The graph also tells about the loss of information in each character (Fig. 8).

Fig. 7 Results of characters with their accuracies

CHARACTER	ACCURACY
Sha	99.65%
Ga	99.57%
Ra	99.44%
Tha(Tha)	98.75%
Ini	98.71%
uuuu	97.54%
Va	92.16%
Rhuuuu	80.39%
Ana	77.20%
Cha	70.43%

Fig. 8 Epoch versus loss/accuracy



6 Conclusion

In OCR, there is a need to recognize the character more accurately. In this paper, we have used deep learning technique, i.e., CNN that suits best for recognizing the images. The model is trained using the dataset that has over 1000 images of 52 handwritten Telugu characters. The accuracy and effectiveness of the CNN method can be increased by increasing the dataset of the characters. The accuracy obtained using this model is 90% approximately. The future enhancement of this work can lie in effective identification of characters from the connected sentences or words.

References

1. Sahara P, Dhok SB (2018) Multilingual character segmentation and recognition schemes for Indian document images, Jan 2018. IEEE
2. Sharma S, Cheeran AN, Sasi A (2017) An SVM based character recognition system, May 2017. IEEE
3. Jabir Ali V, Joseph JT (2018) A convolutional neural network based approach for recognizing malayalam handwritten characters. Int J Sci Eng Res
4. Dara R, Panduga U (2015) Telugu handwritten isolated character recognition using two dimensional fast Fourier transform and support vector machines. IJCA
5. Angadi A, Vatsavayi VK, Gorripati SK (2018) A deep learning approach to recognize handwritten Telugu character using convolutional neural networks. In: Proceedings of 4th ICCM2018
6. Inuganti SL, Ramisetty RR (2019) Online handwritten Indian character and its extension to Telugu character recognition. IJRE
7. Vishwanath NV, Manjunathachari K, Satya Prasad K (2018) Handwritten Telugu composite characters recognition using morphological analysis. IJPAM
8. Mohana Lakshmi K, Venkatesh K, Sunaina G, Sravani D, Dayakar P (2017) Handwritten Telugu character recognition using Bayesian classifier. IJET
9. Chakradhar CV, Rajesh B, Raghavendra Reddy M (2016) A study on online handwritten Telugu character recognition. IJSETR
10. Kaur J, Kaur R (2017) Review of the character recognition system process and optical character recognition approach. IJCSMC

Implementation of Python in the Optimization of Process Parameters of Product Laryngoscope Manufactured in the Injection Mold Machine



Balachandra P. Shetty, J. Sudheer Reddy , B. A. Praveena ,
and A. Madhusudhan

1 Introduction

Many researchers apply theoretical and experimental techniques to develop complex-shaped injection molded parts. However, their functional performances are dependent on mold thermal parameters and process parameters [1, 2]. In literatures, authors have focused on mold design by performing computer-aided design software analysis [3]. Rapid progress in plastic material led to great increase in new materials every year [4]. The injection molding industry has shown significant attention to develop cost-effective production methods after World War II for manufacturing medical products. Manufacturing laryngoscope (medical device) used for airway management is an essential element to attain quick and positive tracheal intubation [5].

The surface roughness of the laryngoscope plays a vital role, as anesthesiologists use it to lift the tongue for fixing the air pipe in the larynx of patients. Injection-molded parts quality is influenced because of many operating parameters [6, 7]. Inappropriate choice of any parameter affects the near-net-shape manufacturing capability (parts require post-processing operations), increases manufacture lead time, energy consumption, and cost [8, 9]. Therefore, influencing parameters to fabricate high-quality products require process optimization.

Trial and error method in optimizing parameters results in increased material usage, cost, labor, energy, and time. Industry personnel or practice engineers require a simple mathematical tool and code that offers quick solutions for solving optimization problems [10, 11].

There are many publications that focused on injection molding process parameters optimization, some of them are more academic and difficult to practice. In the

B. P. Shetty · J. Sudheer Reddy · B. A. Praveena (✉) · A. Madhusudhan
Department of Mechanical Engineering, Nitte Meenakshi Institute of Technology, Bengaluru,
Karnataka 560064, India
e-mail: praveen.ba@nmit.ac.in

present paper, Taguchi method is applied practically to optimize process parameters to determine suitable parametric conditions to get the best product [12–14]. The statistical implementation of the Taguchi method is carried out using Python codes.

2 Methodology

2.1 Product Laryngoscope and the Process

To reduce the transmitted exerted force on the organs such as tooth and soft tissue lesions and for positive tracheal intubation, there is a requirement to develop an alternative device. The innovative product laryngoscope is as shown in Fig. 1 is designed and manufactured to obtain video feature of larynx.

The present work is focused on manufacturing the double channel laryngoscope through the process injection molding as it is opined as the best choice [15]. In injection molding, the raw material is heated followed by melting in injection unit and later transferred to mold subjected to higher pressure by clamping the molds. The design features of the product need to be carefully modified such that the part ensures better surface finish. Polymer chains will deform, when the melt transfer takes place from sprue to runner, followed by gates, ingates, and cavities. During the process of deformation, high shear stress and rapid cooling take place at the forefront of mold surfaces. Thereby, complex crystalline morphology can be observed clearly in the microstructure of semi-crystalline polymers of injection molded parts [16].

Fig. 1 Fabricated laryngoscope samples



2.2 Optimization of Surface Roughness Using Taguchi

Using orthogonal arrays (OA), Taguchi is a strategy to minimize the number of experiments. Engineers at industries apply the Taguchi method for various applications involving developing new products, monitoring, controlling, and improving the existing product quality and processes [17]. Design of experiments (DOE) is an efficient technique that requires lesser experimental trials and estimates individual and interaction factors [18–20]. The product quality of injection molded parts is influenced by many parameters. The appropriate parameters and levels are decided after conducting experiments and consulting literature [7, 8]. The parameters and levels set for the present work are given in Table 1.

The present work investigated four critical parameters of operation at three levels, and accordingly Taguchi L₉, orthogonal array experimental plan was employed. Taguchi L₉ matrix used for experimentation is presented in Table 2.

Each experimental condition is repeated thrice to minimize variation and performs accurate analysis. The injection molded parts (i.e., laryngoscope) obtained for each molding condition are subjected to quality evaluation (i.e., surface roughness) using Mitutoyo surface roughness tester. Total nine roughness (3 measurements × 3 replicate) values are recorded for each experimental condition, and the average values are

Table 1 Input parameters and levels of plastic injection molding process

Variables	Units	Levels
Injection pressure (<i>A</i>)	kg/cm ²	100, 130, 160
Injection velocity (<i>B</i>)	m/s	20, 30, 40
Melt temperature (<i>C</i>)	°C	220, 230, 240
Mold temperature (<i>D</i>)	°C	60, 80, 100

Table 2 Input–output data of plastic injection molding process

Exp. No.	Input parameters				Output variables	
	<i>A</i> (kg/cm ²)	<i>B</i> (mm/s)	<i>C</i> (°C)	<i>D</i> (°C)	Ra (μm)	S/N ratio (dB) $-10 \log \left[\frac{1}{n} \sum_{i=1}^n y_i^2 \right]$
L ₁	100	20	220	60	0.589	4.60
L ₂	100	30	230	80	0.495	6.11
L ₃	100	40	240	100	0.433	7.27
L ₄	130	20	240	100	0.314	10.06
L ₅	130	30	230	60	0.354	9.02
L ₆	130	40	220	80	0.341	9.34
L ₇	160	20	240	80	0.373	8.56
L ₈	160	30	220	100	0.214	13.39
L ₉	160	40	230	60	0.321	9.87

presented in Table 2. The objective function is to minimize the surface roughness, and therefore, signal-to-noise (S/N) ratio equation is identified. S/N ratio values correspond to the surface roughness values which are computed for every experimental condition presented in Table 2.

2.3 Study of Injection Molding Factors

The effect of injection molded factors on the quality of parts presented in Table 1 is experimentally studied. Higher injection pressure forces the hot-melt close to die surface walls, to create the replica of the mold surface, an increase in injection pressure tends to reduce the surface roughness of injection molded parts. The desired minimal surface roughness was obtained at the mid-values of injection speed as shown in Fig. 2. This might be due to filling defects at low injection speed and burrs and jetting at higher injection speed [8]. A combination of low melt temperature and high mold temperature resulted in reduced surface roughness values (refer Fig. 2). Too low mold and melt temperature results in low melt viscosity, promoting shrinkage, warpage, and flow lines [21, 22]. In contrast, too high a temperature results in an excessive flash and burning [8]. The obtained results strongly justify the published literature.

The summary of results of the Pareto analysis of variance were constructed for performing the factor analysis (sum at factor levels, percent contribution) and to determine the optimal conditions for surface roughness. The sum at factor levels

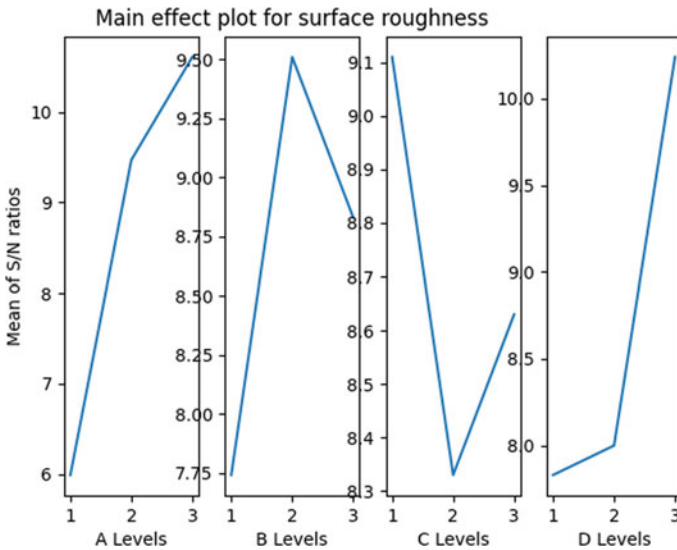


Fig. 2 Main effect plot for surface roughness from Matplotlib

Table 3 Pareto analysis of variance for surface roughness

Variables	Levels	A	B	C	D	Total
Sum at variable levels	1	17.98	23.22	27.33	23.49	78.22
	2	28.42	28.52	25	24.01	
	3	31.82	26.48	25.89	30.72	
Sum of squares of differences		312.10	42.88	8.29	97.57	460.84
Variable contribution (%)		67.72	9.30	1.80	21.17	100.00
Optimal condition		$A_3B_2C_1D_3$				

was chosen to correspond to the signal-to-noise ratio values of surface roughness of experimental trials. Note that, injection pressure followed by mold temperature and injection velocity showed a significant impact on surface roughness (refer to Table 3). Higher values of *S/N* ratio correspond to sum at each level of variables and were treated as optimal conditions for minimized surface roughness. The optimal conditions ($A_3B_2C_1D_3$) determined for minimum surface roughness was obtained correspond to the factor *A* level 3, factor *B* level 2, factor *C* level 1, and factor *D* level 3 set at 160 kg/cm², 30 m/s, 220 °C, and 100 °C, respectively. The optimal conditions for minimum surface roughness (0.214 μm) are obtained to correspond to the 8th experimental trial presented in Table 2. This strongly justifies that with limited experimental trials, computation time, and efforts, the Taguchi method determines the optimal conditions for the plastic injection molding process.

2.4 Surface Roughness Examinations

The surface roughness evaluations are made on the fabricated parts correspond to experimental conditions given in Table 4. The optimal conditions are compared with the initial conditions of the L_9 orthogonal array for surface roughness quality assessments. The laryngoscope parts (optimal and initial conditions) are subjected to quality assessment to graphically visualize the surface textures using a high magnification objective lens equipped in the non-contact-type confocal microscope. Note that optimal conditions were demonstrated with smooth surface peaks compared to initial conditions of Taguchi L_9 orthogonal array experiments (refer Table 4).

3 Implementation of Python

The implementation of Taguchi optimization is done through Python code for surface roughness optimization. To increase computational efficiency and to take care the nonlinearities, the Python code is used. The Python code provides efficient platform from which further extensions are easily possible. As the program code is in modular

Table 4 Initial and optimal conditions correspond to the Taguchi L_9 method

Experimental condition	Input variable	Output variable
Optimal condition (Exp. No. L_8 from Table 2)	$A = 160 \text{ kg/cm}^2$ $B = 30 \text{ mm/s}$ $C = 220 \text{ }^\circ\text{C}$ $D = 100 \text{ }^\circ\text{C}$	Surface roughness: $0.214 \text{ } \mu\text{m}$
Initial condition (Exp. No. L_1 from Table 2)	$A = 100 \text{ kg/cm}^2$ $B = 20 \text{ mm/s}$ $C = 220 \text{ }^\circ\text{C}$ $D = 60 \text{ }^\circ\text{C}$	Surface roughness: $0.589 \text{ } \mu\text{m}$

construction, new functions and variables can be built in easily. Through this, various other algorithms can also be implemented. The code is structured to compute signal-to-noise ratio, to build up orthogonal array, and from that to develop sum at factor level (SFL) table and finally to obtain parameter-wise optimum levels. The SFL table is further utilized to get the graph of Pareto analysis of variance of surface roughness as shown using mat plot library. The important modules of Python implementation are as shown below.

```
# To find surface roughness sn ratio (smaller the better)
sr_sn = []
def sn_ratio():
    for n in sr_response:
        res1 = (n * n)
        res2 = math.log10(res1)
        res3 = -10 * res2
    sr_sn.append(round(res3,2))
sr_sn_response = [0.589,0.495,0.433,0.314,0.354,0.341,0.373,0.214,0.321]
sn_ratio()
# To find surface roughness sum at variable (factor) levels and OA list.
sr_a = np.array([[1,1,1],[4.60, 6.11,7.27],[2,2,2],[10.06,9.02,9.34],
                 [3,3,3],[8.56,13.39,9.87]])
sr_b = np.array([[1,1,1],[4.60, 10.06,8.56],[2,2,2],[6.11,9.02,13.39],
                 [3,3,3],[7.27,9.348,9.87]])
sr_c = np.array([[1,1,1],[4.60, 9.34,13.39],[2,2,2],[6.11,9.02,9.87],
                 [3,3,3],[7.27,10.06,8.56]])

sr_d = np.array([[1,1,1],[4.60, 9.02,9.87],[2,2,2],[6.11,9.34,8.56],
                 [3,3,3],[7.27,10.06,13.39]])
sfl_abcd = []
def sfl(* sr_snr):
    total_1 = total_2 = total_3 = 0
    for i in range(0,6):
        for j in range(0,3):
            if sr_snr[i][j] == 1:
```



```

        total_1 += sr_snr[i + 1][j]
elifsr_snr[i][j] == 2:
        total_2 += sr_snr[i + 1][j]
elifsr_snr[i][j] == 3:
        total_3 += sr_snr[i + 1][j]
sfl_abcd.append(round(total_1,2))
sfl_abcd.append(round(total_2,2))
sfl_abcd.append(round(total_3,2))
def divide_chunks(l, n):
    # looping till length l
    for i in range(0, len(l), n):
        yield l[i:i + 3]
sfl_abcd = list(divide_chunks(sfl_abcd, 3))
print(sfl_abcd)

# find max in list of lists factor wise
optimum_f_1 = list(map(max,sfl_abcd))
print(optimum_f_1)

# Sum of squares of deviation
m = sfl_abcd
t_ssd = []

def ssdiff(i):
    for list in m:
        ssd1 = (m[i][0] - m[i][1]) ** 2
        ssd2 = (m[i][0] - m[i][2]) ** 2
        ssd3 = (m[i][1] - m[i][2]) ** 2
    ssd = ssd1 + ssd2 + ssd3
    t_ssd.append(round(ssd,2))

ssdiff(0)
ssdiff(1)
ssdiff(2)
ssdiff(3)

# To find percentage contribution ratio
sum_t_ssd = sum(t_ssd)
print(sum_t_ssd)
p_c_ratio = []

```

```

def percent_cont_ratio():
    for i in range(len(t_ssd)):
        p_c_ratio.append(round((t_ssd[i]/sum_t_ssd) * 100,2))

percent_cont_ratio()

```

4 Conclusion

Optimum process parameters in injection molding process are of high priority as they are not controlled by equations and depend more on in equations. In general, setting the process parameters is left to the experience of the plastic engineer. Since the plastic exhibits a complex thermo-viscoelastic property, selecting proper parameters from varying values is a challenge. Plastic engineers otherwise select the parameters from handbooks and then adjust by the trial-and-error method. The purpose of this variance analysis is to investigate which factor primarily affect the performance characteristic of the injection molding process. The implementation of this optimization process is carried out through Python codes.

The conclusions drawn from the present work are as follows:

- (a) Taguchi method was applied to study the influencing parameters (injection pressure, injection velocity, and mold and melt temperature) that could affect the surface roughness of laryngoscope parts. Injection pressure showed a significant impact on surface roughness, followed by mold temperature and injection velocity.
- (b) Taguchi method determined optimal conditions (injection pressure of 160 kg/cm², injection velocity of 30 m/s, mold temperature of 100 °C, and melt temperature 220 °C) which could reduce the surface roughness of laryngoscope part to 0.214–0.589 μm compared to initial setting conditions (injection pressure of 100 kg/cm², injection velocity of 20 mm/s, mold temperature of 60 °C, and melt temperature 220 °C). The smooth surface textures are obtained for optimal conditions compared to the initial setting of the Taguchi L₉ method.

Acknowledgements We appreciatively acknowledge the support from TIDE programme of DST, Government of India, for funding this project.

References

1. Vojnova E (2016) The benefits of a conforming cooling systems the molds in injection molding process. *Procedia Eng* 149:535–543
2. Bianchi MF, Gameros AA, Axinte DA, Lowth S, Cendrowicz AM, Welch ST (2021) Regional temperature control in ceramic injection molding: an approach based on cooling rate optimization. *J Manuf Process* 68:1767–1783

3. Lakkanna M, Kumar GCM, Kadoli R (2016) Computation design of mould sprue for injection moulding thermoplastics. *J Comput Des Eng* 3:37–52
4. Bryce DM (1996) Plastic injection molding: manufacturing process fundamentals. Society of Manufacturing Engineers, Dearborn, MI, p 253
5. Mihara R, Komazawa N, Matsunami S, Minami T (2015) Comparison of direct and indirect laryngoscopes in vomitus and hematemeses settings: a randomized simulation trial. *Biomed Res Int*. <https://doi.org/10.1155/2015/806243>
6. Llewelyn G, Rees A, Griffiths C, Jacobi M (2020) A design of experiment approach for surface roughness comparisons of foam injection-moulding methods. *Materials* 13(10):2358
7. Mohan M, Ansari MNM, Shanks RA (2017) Review on the effects of process parameters on strength, shrinkage, and warpage of injection molding plastic component. *Polym Plast Technol Eng* 56(1):1–12
8. Kashyap S, Datta D (2015) Process parameter optimization of plastic injection molding: a review. *Int J Plast Technol* 19(1):1–18
9. Fernandes C, Pontes AJ, Viana JC, Gaspar-Cunha A (2018) Modeling and optimization of the injection-molding process: a review. *Adv Polym Technol* 37(2):429–449
10. Davis R, John P (2018) Application of Taguchi-based design of experiments for industrial chemical processes. In: Silva V (ed) *Statistical approaches with emphasis on design of experiments applied to chemical processes*, p 137. <https://doi.org/10.5772/65616>
11. Antony J, Antony FJ (2001) Teaching the Taguchi method to industrial engineers. *Work Study* 50(4):141–149
12. Li K, Yan S, Zhong Y, Pan W, Zhao G (2019) Multi-objective optimization of the fiber-reinforced composite injection molding process using Taguchi method, RSM, and NSGA-II. *Simul Model Pract Theory* 91:69–82
13. Kiatcharoenpol T, Vichiraprasert T (2018) Optimizing and modeling for plastic injection molding process using Taguchi method. *Int J Phys Conf Ser* 1026(1):012018
14. Martowibowo SY, Khlooun R (2019) Minimum warpage prediction in plastic injection process using Taguchi method and simulation. *Manuf Technol* 19(3):469–476
15. Serban D, Lamanna G, Opran CG (2019) Mixing, conveying and injection molding hybrid system for conductive polymer composites. *Procedia CIRP* 81:677–682
16. Praveena BA, Shetty BP, Lokesh N, Santhosh N, Buradi A, Jalapur R (2023) Design of injection mold for manufacturing of Cup. In: Pradhan P, Pattanayak B, Das HC, Mahanta P (eds) *Recent advances in mechanical engineering. Lecture notes in mechanical engineering*. Springer, Singapore. https://doi.org/10.1007/978-981-16-9057-0_8
17. Dehnad K (2012) *Quality control, robust design, and the Taguchi method*. Springer Science & Business Media
18. Roy RK (2010) *A primer on the Taguchi method*. Society of Manufacturing Engineers, Dearborn, MI
19. Jeevamalar J, Kumar SB, Ramu P, Suresh G, Senthilnathan K (2021) Investigating the effects of copper cadmium electrode on Inconel 718 during EDM drilling. *Mater Today Proc* 45:1451–1455
20. Suresh G, Srinivasan T, Rajan AJ, Aruna R, Ravi R, Vignesh R, Krishnan GS (2020) A study of delamination characteristics (drilling) on carbon fiber reinforced IPN composites during drilling using design experiments. *IOP Conf Ser Mater Sci Eng* 988(1):012008
21. Azad R, Shahrajabian H (2019) Experimental study of warpage and shrinkage in injection molding of HDPE/rPET/wood composites with multiobjective optimization. *Mater Manuf Process* 34(3):274–282
22. Benedetti L, Brulé B, Decreamer N, Evans KE, Ghita O (2019) Shrinkage behaviour of semi-crystalline polymers in laser sintering: PEKK and PA12. *Mater Des* 181:107906

A Practical Approach to Software Metrics in Beehive Requirement Engineering Process Model



K. S. Swarnalatha

1 Introduction

One of the stiffest parts of building a product/software framework is choosing with respect to what to construct/build [2]. None of alternate parts of the reasonable work is as hard as building up the point by point prerequisite particular. A portion of alternate parts of the product advancement process do not injure the subsequent framework if fouled up as it could be amended later. Requirements specification says what the framework ought to do, how it must act under the given limitations, the qualities it must display and have. Requirement engineering stage manages specialized learning, hierarchical, administrative, financial and social issues. According to the IEEE standard, requirement is characterized as

- An ability or condition required by a client to accomplish a goal or take care of an issue
- An ability or condition that must be met or controlled by a framework or framework part to fulfill a standard, an agreement, particular or other formally forced record
- A documented portrayal of n ability or condition as in the definition 1 or 2.

Three essential parts are incorporated into requirement elicitation and specification phase: the requestor (alluded to as the 'end client'), the designer (one who outlines and execute the framework) and the creator (one who records the prerequisites). The nature of the product application being composed and created is specifically identified with the nature of its requirement specification. RE process includes

K. S. Swarnalatha (✉)

Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore 560064, India

e-mail: swarnalatha.ks@nmit.ac.in

requirement elicitation, requirement verification and validation, requirement specification, requirement analysis and management with some interleaving between them [3]. It also includes various measurements to effectively oversee, execute and finish the product ventures. Requirement metrics are to a great degree valuable in distinguishing the dangers of a venture by finding blunders or peculiarities in the prerequisite report. Beehive RE process model finds a middle ground between the conventional models of requirement analysis and requirements collection, viz. waterfall, spiral and agile methods. In this paper, we explore how metrics help in analyzing the beehive RE process model [4] with the current RE process models. The focal point of this exploration is to feature the significance of different execution measurements which could be taken after for investigating beehive requirement engineering.

2 Performance Measurement

Programming measurements are an estimation of programming task or item; it is a degree to give quantitative examination of the degree to which the undertaking or an item can be estimated. Measurements can be utilized to gauge the alluring characteristics like comprehend capacity, testability, practicality and unwavering quality of a product item or a task [4, 5]. The primary objective of the product measurements is to recognize and evaluate the basic certainties which specifically or by implication influences the prerequisite building process. The prerequisites that are accumulated from the customers can be connected on the accompanying measurements to distinguish the parameters in view of which we can finish up whether the necessity designing procedure can be taken further or not.

An application has been composed and created utilizing both conventional and beehive RE processes to show the accompanying information has been accumulated and execution of the model has been estimated in light of performance measurements. The beehive RE process has been actualized and approved with few organizations which are AZTEC, VIGSUN and DICS. Results are appeared underneath in examination with waterfall and spiral process models. The point by point assessment of every metric is said as beneath.

3 Metrics

Ambiguity—It is one of the important metrics to be used to identify whether the requirement is clearly mentioned without any confusion, it alludes to absence of clearness in programming setting, could prompt a more noteworthy expenses brought about in the software development process. Prerequisite examination is the rotate, on which the product being created rests and it is basic that there in insignificant unclearness in this stage, inconsistencies in verbal correspondence, absence of brief

cognizance and human mistakes amid information extraction and unavoidable issues that emerge at this phase of the improvement cycle. To this end, it is of vital significance that we recognize the territories and extent of the mistakes along these lines created. The coherent successor to this progression is constrain uncertainty and fortify lucidity to additionally improve the balance and exactness of the ultimate result. The ambiguity for any given requirement lies between 0 and 1 where 0 shows requirement has certainly no ambiguous and 1 shows requirement is more ambiguous.

$$\text{ambiguity} = 1 - \frac{(\text{clearness value})}{\text{Number of Domains}} \tag{1}$$

- **Fully Understood**—In the RE process, analysis phase is principally portrayed by the distinguishing proof, accumulation and assessment of the requirement. The requirements are said to be completely comprehended when every one of the parts and their points of interest have been accumulated, altogether reviewed and obviously contemplated out. The requirement is completely comprehended for the item or undertaking the designer requires not investing much energy in investigation of the necessities and he can proceed onward to next period of programming improvement life cycle that is configuration stage, the requirement is completely seen then the engineer can convey the item on time. The fully understood requirements range between 0 and 1 where 0 shows requirements are not understood completely and 1 show that the necessary requirements are understood completely.
- **Partially Understood**—The requirements are halfway comprehended when they are fragmented or just a small amount of prerequisites are represented, subsequently prompting equivocalness and dubiousness. It is thusly basic to completely comprehend the necessities inside and out to guarantee smooth cruising through the product improvement process, and the prerequisites are not seen totally then cause issues in every single stage, lastly, the convey item may not fulfill the client prerequisites. The scope of somewhat comprehended lies between [0, 1] where 0 (Minimum) shows just couple of prerequisites are in part comprehended and 1 (Maximum) demonstrates the greater part of the necessities are incompletely comprehended.

$$\text{partially understood} = 1 - \text{fully understood} \tag{2}$$

- **Conciseness**—The conciseness of a requirement document relies upon its clearness and length of dialect utilized for a requirement document to be compact, and it is of essential significance to smother all intricate and super streams detail. The range of the metric: size of requirement document ought to be minimum.

$$\text{Smallest range of concise} = 1 + \frac{1}{1} = 0.5 \text{ (Upper Threshold)}$$

Extreme range of concise = (file size is large i.e. theoretically infinity)

$$\text{Conciseness} = 1 + \frac{1}{1 + \text{Very Large}} = \sim \text{Lower threshold}$$

$$\text{Conciseness} = \frac{1}{1 + \text{size}} \quad (3)$$

- **Reusability**—Reusability alludes to code, process, modules and so on that can be utilized again and again to accomplish comparative objectives in various items. Reusing existing code/process accelerates the improvement procedure as well as diminishes the quantity of mistakes. An examination with the existing model demonstrates that least time is required to create a similar item utilizing the beehive RE process, considering five cycles. The software/product is developed from version 1.0 to 1.5 (5 emphases). Time taken to deliver the software/product item (in days) is demonstrated as follows. The reusability range cannot be determined since it is software/product dependent.
- **Level of Detail**—It is an important metric to understand and estimate the clear requirements. To know the level of detail of individual requirement, the following metric separates the requirements as clear and unclear (ambiguous/non-ambiguous), and this metric is a measure of the reasonable and non-ambiguous. The range of this metric is min = zero and max = number of non-ambiguous requirements/total number of requirements.

$$\text{Level of Detail} = \frac{\text{Number of Non-Ambiguous Requirement}}{\text{Total Number of Requirements}} \quad (4)$$

- **Internal Consistency**—This metric focuses on the uniqueness measure of the item to decide the internal consistency. It is the proportion of elite one of kind segments to the general special things when all is said in done. A correlation of the beehive RE process with the conventional strategy demonstrates that the beehive RE process creates a littler interior consistency proportion than the other. The scope of this metric is: Number of distinctive requirements = x

Number of unique and non-deterministic requirements = q

Max value when $q = 0$

$$= x - q/x$$

$$= x - 0/x$$

$$= 1$$

Min value when $z = 0$

$$= x - q/x$$

$$= x = q$$

$$= q - q/x$$

$$= 0$$

i.e., min = 0
 Max = 1

$$\begin{aligned} &\text{Internal consistency} \\ &= \frac{\text{no. of unique requirements} - \text{no. of unique and non deterministic requirements}}{\text{no. of unique requirements}} \end{aligned} \tag{5}$$

- **Annotation by Relative Importance**—Division of necessities that are commented on as essential. An item is described by certain one of a kind highlight. These are generally more essential than alternate highlights. Portion of prerequisites that are commented on as essential are figured and contrasted and the other two models, in a division of four classes. Here as indicated by our presumption, we considered the prerequisites that are have a place with class 4 is having most noteworthy need and correspondingly class 1 is less essential necessities, due lack of time or any deficiency of assets in building up the venture, we can overlook the necessities from class 1. The territory cannot be resolved for this metric since it relies upon the need of the prerequisite.
- **Annotation by Relative Stability**—Highlights of an item named as steady additionally are exceptional to the given item since they do not change with the make. An examination of the steadiness highlights is given as takes after. The scope of the metric is [0–1], where 0 (minimum) indicates that none of the requirements is stable and 1 (maximum) indicates relatively all requirements are stable.

$$\text{Annotation by Relative Stability} = \frac{\text{Number of stable requirements}}{\text{Total number of requirements}} \tag{6}$$

- **Preciseness**—The ‘positive’ term in ‘true positive’ alludes to the recognizable proof of a prerequisite as important to the procedure, and the ‘true’ term represents the effective distinguishing proof of the same. To put it plainly, genuine positive can be characterized as the effective recognizable proof of a necessity as basic to the prerequisites building process. The ‘negative’ term in ‘false negative’ alludes to thinking about a necessity as pointless to the procedure, while the ‘false’ term demonstrates that this thought isn’t right. In a word, false negative can be characterized as the inaccurate assurance of a prerequisite as unimportant to the process. The range of the metric is [0–1]

$$\text{Precise} = \frac{n_p}{n_p + n_f} \tag{7}$$

True → +ve → n_p , false → -ve → n_f .

- **Redundancy**—The redundant attributes are the ones that are not novel to the given item. It is estimated as an overabundance of highlights over the exceptional qualities. This proportion in a perfect world must be low. An examination between the two models demonstrates that the beehive RE process delivers lower redundancy quotient when compared to existing models like waterfall and Boehm spiral. Redundant is the demonstration of utilizing a word or an expression that

rehashes pointlessly and in this way unusable. Repetition in RE process includes a specific prerequisite being said over and again in various settings. This can be dispensed with by saying the prerequisite in gathering of settings together. It winds up important to evaluate the excess in the arrangement of prerequisites given. In the event that n_{total} speaks to the aggregate arrangement of prerequisites and just n_{unique} necessities are extraordinary or non-excess, at that point repetition can be ascertained as under. The scope of the metric is [0–1].

$$\text{Redundancy} = \frac{n_{\text{total requirements}} - n_{\text{unique requirements}}}{n_{\text{total number of requirements}}} \tag{8}$$

- **Atomicity**—The scope of atomicity ranges between [0–1], where 0 shows no atomicity, 1 demonstrates atomicity. The atomicity is computed as the out degree for every one of the hub must be figured. The aggregate check gives the atomic qualities for that domain.

$$\text{Atomicity} = \frac{\text{Number of atomic requirements}}{\text{Total number of requirements}} \tag{9}$$

- **Average Degree of Dependencies**—The scope of normal level of dependency lies between [0, Max (out level of necessity i)]. The normal level of dependency is figured utilizing the dependency chart

$$\text{Average degree of requirements} = \sum_{i=1}^n \frac{\text{Degree of requirements}}{\text{Total number of requirements}} \tag{10}$$

- **Stochasticity**—The scope of the stochasticity lies between [0–1], where 0 shows the requirements are not subject to outside elements and 1 demonstrates requirements rely upon outer components. Stochasticity is a term which demonstrates that requirements are reliant on outer factor, for every requirement, dole out qualities in the scope of 0–1. In the event that a requirement totally relies upon the outer factor relegate esteem 1. On the off chance that a requirement does not rely upon outer factor 0 is doled out. Additionally, allocate stochastic characteristics and complete all of the qualities for each and every one of the domains for all of the criteria in the region. This gives the stochasticity for the necessities of the specific task/item,

$$\text{Stochasticity} = \frac{\text{Number of stochastic requirements}}{\text{Total number of requirements}} \tag{11}$$

- **Effective Ambiguity**— For each requirement, an ambiguity value between 0 and 1 is assigned. If a particular requirement is more uncertain, assign value 1; otherwise, assign value 0. For each requirement, assign the need class, which determines whether that particular requirement belongs to the highest need class,

which is 4, or the lowest need class, which is 1, perform duplication of the ambiguity value and need class will give us the practicable uncertainty for each. Figured in every neighbourhood.

$$\text{Effective Ambiguity} = \frac{\sum \text{Ambiguity} * \text{Importance Class}}{\sum \text{Number of Requirements}} \quad (12)$$

4 Result and Analysis

The beehive process model has been compared with other 2 conventional models, namely waterfall, spiral model. The beehive RE process model has been validated in 3 companies. Table 1 gives the consolidated results for a set of 92 requirements. The beehive RE process model was design and developed to overcome most of the shortfalls of the traditional model methods like waterfall model, spiral model and agile method. This model is proved to be more effective in terms of less ambiguity, better level of understanding, reduced redundancy, reduced randomness and increased usability. Beehive RE process has proved to be an operative process.

5 Conclusion and Future Enhancements

Utilizing measurements to quantify execution of programming forms assumes a vital part in programming building. The outcomes got by these measurements go about as execution markers for various exercises, areas and antiquities of the product forms. These measurements give dynamic help to viably controlling and estimating programming ventures. In this paper, an arrangement of execution measurements is displayed that can be utilized to control and deal with the product extends in a more reasonable way. Our handy experience of utilizing the previously mentioned execution measurements in various programming process models demonstrates that these measurements are exceptionally compelling to stay informed concerning the present condition of the undertakings of the distinctive programming forms and the product created by utilizing these measurements have effectively been sent. We trust that by actualizing these execution measurements, and the product improvement firms can profit by a more profound understanding about their activities which consequently will not just better oversee the product, yet in addition contribute toward accomplishment of the business objectives in a viable way. To additionally upgrade and expand extent of our proposed metric, we plan to investigate extra measurements for execution estimation for the prerequisite building period of the product extends as an imminent future work to this.

Table 1 Beehive process model comparison with conventional models

S. No.	Metrics	Metric values		
		Waterfall model	Beehive process model	Spiral model
1	Ambiguity	0.166	0.1277	0.1313
2	Fully understood	0.83	0.87	0.8685
3	Partially understood	0.17	0.13	0.1313
4	Conciseness	0.018	Avg = 0.1268	0.02
5	Reusability	139.5 man hours	134 man hours	136 man hours
6	Level of detail	0.833	0.869	0.8686
7	Internal consistency	0.8915	0.863	0.905
8	Annotation by relative importance	Class 1—20 Class 2—20 Class 3—43 Class 4—13	Class 1—21 Class 2—12 Class 3—35 Class 4—24	Class 1—16 Class 2—39 Class 3—27 Class 4—17
9	Annotation by relative stability	0.844	0.891	0.8282
10	Preciseness	0.9146	0.9397	0.8369
11	Redundancy	0.156	0.045	0.1616
12	Atomicity	0.53	0.5245	0.56
13	Average degree of dependency	1.5	1.05	1.5
14	Stochastic	0.30	0.299	0.26
15	Effective ambiguity	0.301	0.281	0.250

References

1. Swarnalatha KS, Srinivasan GN, Bhandary PS (2014) A constructive and dynamic frame work for requirement engineering process model—bee hive model. *Int J Comput Eng Technol (IJ CET)* 5(7):48–54. ISSN 0976-6367 (Print), ISSN 0976-6375 (Online)
2. Swarnalatha KS, Srinivasan GN (2013) A survey on emerging trends in requirement engineering for a software development life cycle. *Int J Adv Res Comput Commun Eng* 2(1):950–957. ISSN (Print): 2319-5940, ISSN (Online): 2278-1021
3. Svensson RB, Höst M, Regnell B (2010) Managing quality requirements: a systematic review. In: 2010 36th EUROMICRO conference on software engineering and advanced applications, pp 261–268. ISBN 978-0-7695-4170-9/10 \$26.00 © 2010 IEEE. <https://doi.org/10.1109/SEAA.2010.55>
4. Ali MJ (2006) Metrics for requirements engineering. Master thesis submitted to Umeå University, 15 June 2006
5. Morasca S (2013) Fundamental aspects of software measurement. In: De Lucia A, Ferrucci F (eds) *Software engineering. ISSSE 2010, ISSSE 2009, ISSSE 2011. Lecture notes in computer science*, vol 7171. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-36054-1_1

Suitability of Process Models for Software Development



K. S. Swarnalatha

1 Introduction

A lot of software development process models are available today. As each of these comes with their own strengths and weaknesses, developers are often faced with the challenge of choosing the right model [1]. Choosing the right model being a very important first step, quantitative tests to evaluate models for desired characteristics can hugely simplify the task and help developers make better decisions in the choice of the model. Well-defined quantitative measures can also reduce the amount of time required to choose the right model. A defined set of metrics for understanding a model are present today [2], but a quantitative approach to measure them is lacking. Some work is done in analysing metrics and enumerating their properties [3]. In this paper, we define the method for measuring (i) redundancy in model, (ii) persistence of learning and (iii) flexibility of model. Each of the below metrics is defined, and a generic formulation is provided. The formulae are then applied directly or in a modified form to three models.

2 Evaluation Metrics

Redundancy, flexibility and persistence of learning of a software process model are elucidated below.

- a. **Redundancy:** Redundancy in a software process model is a measure of the number repeated steps in one complete cycle. When a model is characterized by

K. S. Swarnalatha (✉)

Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bengaluru 560064, India

e-mail: swarnalatha.ks@nmit.ac.in

no return to previous steps and model flows step-by-step forward till the end, it is as said to have no redundancy. Redundancy for a model is not straightforward to calculate, for the reasons that:

- i. Failure can happen at any step, thereby causing a backtrack to previous steps.
- ii. The probability of failure is not known beforehand.
- iii. Amount of backtracking is not same for failure at different steps of a model.

Here we calculate the average number of redundant steps in a model as specified below in Eq. 1.

Let R be the average number of redundant steps in a software engineering model.

$n \rightarrow$ number of steps in the model

$p \rightarrow$ probability of failure at step i ($1 \leq i \leq n$)

$$R = \sum_{i=1}^n ((\text{number of steps bracktraced from step } i) * (\text{prob of failure at step } i)) \quad (1)$$

A probabilistic weighting of number of retracted steps is done to account for the fact that the probability of failure is not same for all steps. An average/net redundancy count makes it easier to compare models on the basis of this characteristic.

An approximation of failure probabilities for above formula is required in most cases because of the reason that the probability of failure is not known beforehand as it is mentioned earlier. The simplest approximation would be to assign equal probability for failure at every step as shown in below Eq. 2.

\therefore Probability of failure at any step = $\frac{1}{n}$ and the formula gets reduced to:

$$R_{\text{simple}} = \frac{1}{n} * \sum_{i=1}^n (\text{number of steps bracktraced from step } i) \quad (2)$$

- b. **Redundancy Cost:** Once redundancy is identified in a model, and it is necessary to determine the cost of it.

Each step in a software process model incurs different costs. The problems faced while calculating redundancy cost are as follows:

- i. Measuring of cost incurred at each repeated step and
- ii. How to scale up the step cost if step is repeated multiple times.

The cost can be calculated in terms of money, time (man hours), other resources like number of people, etc. The cost in equation (at each step) is either an absolute cost or a relative one. In case of relative cost, all the costs are calculated with respect to the same benchmark cost as shown in Eq. 3

$p_i \leftarrow$ probability that failure occurs at step i

$n \leftarrow$ number of steps in model

$n_i \leftarrow$ number of steps backtracked when failure happens at step i
 $c_{ij} \leftarrow$ cost at step j when failure occurs at step i
 $RC \leftarrow$ average redundancy cost

$$RC = \sum_{i=1}^n p_i * (\text{sum of costs of all the backtracked steps from step } i)$$

$$RC = \sum_{i=1}^n \left(p_i * \sum_{j=i-n_i}^i c_{ij} \right) \tag{3}$$

Note that for any step i , n_i is the total number of repetition of it. If the model is without repetition, then $n_i = 0$, with the step performed only once. The average RC for each step (RC_{step}) can be computed by dividing RC with the average number of redundant steps in the model (R), when $R \neq 0$. $RC = 0$ when $R = 0$.

$$RC_{\text{step}} = \frac{RC}{R}$$

A simpler formulation of the RC formula can be obtained by assuming equal probability of failure at each step, similar to the redundancy formula simplification, reducing RC to RC simple as shown in below Eq. 4.

$$RC_{\text{simple}} = \frac{1}{n} * \sum_{i=1}^n \left(\sum_{j=i-n_i}^i c_{ij} \right) \tag{4}$$

One complication is the calculation of cost at each step. This problem is made worse, when the repeat costs are required to be computed. One of the problems faced with the calculation of redundancy—determining the probability of failure—resurfaces in this formulation too.

- c. **Flexibility:** Flexibility of a model is an important metric for evaluating a model. Gao and Yang have provided a preliminary definition and reasoning for it [4]. We define flexibility as the ability of a model to take corrective actions, with the minimum number of backtracking steps in case of a change like introduction of new requirements, error in design, etc.

A model is characterized by high flexibility if it performs corrective action in the same step where the problem occurred, without any backtracking. On the other end of this spectrum is a model that requires restarting the process from step 1 in case of failure. Such a model is the least flexible. The value of flexibility falls in the range [0, 1], with 1 characterizing the former type of model and 0 characterizing the latter. The equation for flexibility is shown below in Eq. 5.

Flexibility, when failure happens at step i , is

$$F_i = 1 - \frac{\text{number of steps backtracked}}{\text{step numbers at which failure occurred}} \tag{5}$$

This formulation favours shorter number of backtracked steps. Another property is that it favours short backtracks in the later steps than similar short backtracks in earlier steps.

Average flexibility for the model (F_{ave}) is calculated as shown in below Eq. 6:

$$F_{ave} = \frac{\sum_i F_i}{\text{number of steps where failure occurred}} \tag{6}$$

- d. **Persistence of Learning:** In almost all practical applications of a software process model, redundancy exists. If so, we desire a property which tries to minimize the cost of redundancy. That property of the model should ensure that every repetition of a step is easier than the former repetition, i.e. reduce the repeat cost of a step. *Persistence of learning* is precisely this property. This property is a measure of the learning at a particular step that becomes useful in carrying out the repetition of the step. The important question is how we measure the amount of learning at each step. Consider the scenario when a failure at step i takes place and backtracking to step $i - 1$ happens, with $C_i, C_{(i-1)}, C_{(i-1)1}$ as cost of performing step i , cost of performing step $i - 1$ the first time and cost of performing step $i - 1$ the second time (i.e. first repeat), respectively. Larger persistence of learning at step $i - 1$ means that $C_{(i-1)1} < C_{i-1}$. The greater the difference of $C_{(i-1)} - C_{(i-1)1}$, more persistent is the learning. So, we define persistence of learning in terms of this difference, more precisely as percentage change of C_{i-1} . The formulae for persistence are shown below in Eq. 7:

$n_b \leftarrow$ number of steps backtracked

$t \leftarrow$ time of failure

PL \leftarrow Persistence of learning for the model

$p_i \leftarrow$ probability of failure at step i

$$PL = \sum_{\text{for all steps where failure occurred}} \sum_{j=i-nb}^i (p_i) * \frac{(C_j)_{t-1} - (C_j)_t}{(C_j)_{t-1}} \tag{7}$$

These formulations help to quantitatively understand the desired properties of a model. Choosing a model for a particular application requires taking these properties into consideration. Based on the application, some properties are more important

than the others. From a performance standpoint, a good model should have minimal redundancy, and it should have high flexibility and high persistence of learning.

3 Application

The formulae can be applied to three models. Waterfall: waterfall model is one of the oldest software development models, and the model can be adopted when the project size is too large and requirements should be known in advance. V-model: it is an extension of the waterfall model, the significance of *v* is verification and validation, as the name says V&V is applied at each and every phases of software development life cycle. And iterative and incremental model: initial version of the software is developed with few specification; later, it will be iterated to incorporate the new requirements, in incremental model the requirements are divided into independent modules, and each module of the software has to undergo all the phases of software development life cycle [5, 6].

a. Redundancy

1. Waterfall [5, 6]:

Here if we assume the classic model where the requirements are fixed throughout the various steps, without backtracking, then redundancy = 0.

2. V-model [5, 6]:

In this model, from steps 1–5, failure at any step leads to restarting the process. This half is equivalent to a waterfall model. In the second half, failure at any step leads to partial backtracking, not restarting.

$$\begin{aligned}
 R &= \sum_{i=1}^n p_i * (\text{number of steps backtracked} \\
 &\quad \text{from step } i \text{ on occurrence of failure}) \\
 &= p_1(1) + p_2(2) + p_3(3) + p_4(4) + p_5(5) \\
 &\quad + p_6(3) + p_7(5) + p_8(7) + p_9(9)
 \end{aligned}$$

For simplicity sake if we consider equal probability (*p*) for all steps,

$$\begin{aligned}
 R_{\text{simple}} &= p[1 + 2 + \dots + 9] \\
 &= p * 39 = 39/9
 \end{aligned}$$

This value changes based on actual number of steps and the backtrackings employed in variations of the model.

3. Iterative and incremental [5, 6]:

As a waterfall model is used in each step, redundancy within each step = 0. But, in case of this model there is a subtle point to be noticed. Recurrence in actuality is not zero. It is implicit in the extra number of iterations performed. This can be quantified by the below formulation:

$$\max(0, \text{actual} - \text{expected number of iterations})$$

Here before the model is employed, the expected number of iterations is forecasted. If this expectation falls short of the actual number, redundancy exists.

$$R = \max(0, \text{actual} - \text{expected number of iterations})$$

b. Redundancy Cost

1. Waterfall model:
Redundancy cost is 0 as redundancy = 0 in the classic model.
2. V-model:

$$RC = \sum_{i=1}^n \left(p_i \sum_{j=i-n_i}^i c_{ij} \right)$$

$$RC_{\text{ave}} = \frac{RC}{R} = \frac{\sum_{i=1}^n \left(p_i * \sum_{j=i-n_i}^i c_{ij} \right)}{\sum_{i=1}^n p_i * i}$$

The denominator is attained from the redundancy formula of V-model.

3. Iterative and incremental:

$$C_i \leftarrow \text{cost of Step } i$$

$$RC = \sum_{\text{for each of the extra steps, } i} C_i$$

Here the probabilities are excluded as they do not apply for any step, but for the steps of mini waterfall contained in each step

$$RC_{\text{step}} = \frac{RC}{R}$$

$$= \frac{\sum_{\text{for each extra iteration, } i} C_i}{\max(0, \text{actual} - \text{expected number of iterations})}$$

If actual = expected number of iterations, then $RC = 0, RC_{\text{step}} = 0$.

c. **Flexibility**

By the definition, the waterfall model is the least flexible and the iterative and incremental model is the most, among the three models in consideration. V-model falls in between these two. In V-model, the first half steps are equivalent to that of a waterfall model, giving the model 0 flexibility. The second half is partly more flexible than the waterfall model.

1. Waterfall model:

$$F_i = 1 - \frac{\text{number of steps backtracked}}{\text{step number at which failure occurred}}$$

Though redundancy = 0 for this model, when failure at step i occurs, we cannot proceed forward, rather restart the process again. This is equivalent to backtracking to the first step.

$$\therefore F_i = 1 - \frac{i}{i} = 0$$

$$F_{\text{ave}} = \frac{\sum_i F_i}{\text{number of steps where failure occurred}} = 0$$

2. V-model:

$$F_1 = 1 - \frac{1}{1} = 0$$

$$F_2 = 1 - \frac{2}{2} = 0$$

⋮

$$F_5 = 1 - \frac{5}{5} = 0$$

$$F_1 = F_2 = F_3 = F_4 = F_5 = 0$$

This is because as pointed out earlier, the first half of the model is similar to the waterfall model.

$$F_6 = 1 - \frac{2}{6} = \frac{2}{3}$$

$$F_7 = 1 - \frac{4}{7} = \frac{3}{7}$$

$$F_8 = 1 - \frac{6}{8} = \frac{1}{4}$$

$$F_9 = 1 - \frac{9}{9} = 0$$

$$F_{ave} = \frac{\sum F_i}{n} = \frac{\frac{2}{3} + \frac{3}{7} + \frac{1}{4}}{9}$$

In actual practice, the value of F will vary from F_{ave} shown here as it is not required for failure to happen at every step of the model. In such cases, flexibility of one implementation of the model can be used as an approximate measure of flexibility for a future, similar implementation.

3. Iterative and incremental:

The formula for flexibility is not directly applicable for this model.

For one method of applying the formula, assume that the expected number of iterations is m and the actual number of iterations is n with $n > m$.

Let $m < i \leq n$.

For each step $< i$, $F_{step} = 1$.

For every step i , if we think that i might be the last iteration of model, then $i - m$ is the number of backtracked steps.

$$\therefore F_i = 1 - \frac{(i - m)}{i} = \left(\frac{m}{i}\right)$$

$$\therefore F_{ave} = \frac{\sum_{j=1}^m 1 + \sum_{j=m+1}^n \left(\frac{m}{j}\right)}{n}$$

d. Persistence of Learning

1. Waterfall model:

As redundancy = 0, persistence of learning property is not applicable.

2. V-model:

$$\therefore PL = \sum_i \sum_{j=i-nb}^i p_i \left(\frac{(C_j)_{t-1} - (C_j)_t}{(C_j)_{t-1}} \right)$$

For this model, the formula can be directly applied, without any modification.

3. Iterative and incremental:

For this model, we apply the formula using a different interpretation of the extra iterations than for flexibility. The former interpretation can lead to repeated values in each term of the outer summation.

Here we consider that for each i (i , n , and m having the same meaning as explained in PL formulation), the value $\left(\frac{C_{(i-1)} - C_i}{C_{i-1}}\right)$ in place of

$$\sum_{j=i-nb}^i P_i \left(\frac{(C_j)_{t-1} - (C_j)_t}{(C_j)_{t-1}} \right)$$

$$\therefore PL = \sum_{i=m+1}^n \left(\frac{C_{i-1} - C_i}{C_{i-1}} \right)$$

4 Conclusion

The formulation of evaluation metrics needs suitable modifications when applied to different models, as shown in previous section for three different models. Also, these formulations do not cover the gamut of evaluations that are done in the selection of a model. That said, our work can act as a stepping stone for future work in this direction.

References

1. Bokhari MU, Siddiqui ST (2011) Metrics for requirements engineering and automated requirements tools. In: Proceedings of the 5th national conference; INDIACom-2011
2. Ali MJ (2006) Metrics for requirements engineering. Master's thesis, 15 June 2006
3. Srinivasan KP, Devi T (2014) Software metrics validation methodologies in software engineering. *Int J Softw Eng Appl (IJSEA)* 5(6)
4. Gao Y, Yang Y. "Flexibility" of software development method
5. Pressman R. *Software engineering—a practitioner's approach*, 7th edn. McGraw-Hill. Accessed Nov 2016
6. Somerville I. *Software engineering*, 9th edn. Pearson Education Ltd. Accessed Nov 2016

Solving Problems of Large Codebases: Uber's Approach Using Microservice Architecture



K. S. Swarnalatha, Adithya Mallya, G. Mukund, and R. Ujwal Bharadwaj

1 Introduction

Back in 2016, Uber chose a drastic plan to change from monolithic to microservices architecture to increase their dominion over the pay per travel industry which revolutionized the doorstep pickup and drop for people.

Monolithic Model: Monolithic architecture has a single codebase with different modules. Modules are isolated as either for business highlights or specialized highlights. It has a single construct framework which constructs the whole application and/or reliance on a system. It too has single executable or deployable solutions.

Architecture: Monolithic apps were designed keeping in mind the ability to handle a plethora of tasks simultaneously. In simple terms, they are complex apps which house several tightly held functions. Monolithic tools also are used specifically for huge codebases. This requires compiling, testing even for a small change in the function within the platform which time is consuming [1]. It is also called multi-layer architecture as monolithic usually has more than three layers which comprises of

UI layer: This is the layer which the user uses to interact with the service which is responsible for all the actions, requests and the retrieval of information.

Business layer: This is responsible for the company's specific business logic or the data logic which changes according to the need of the application.

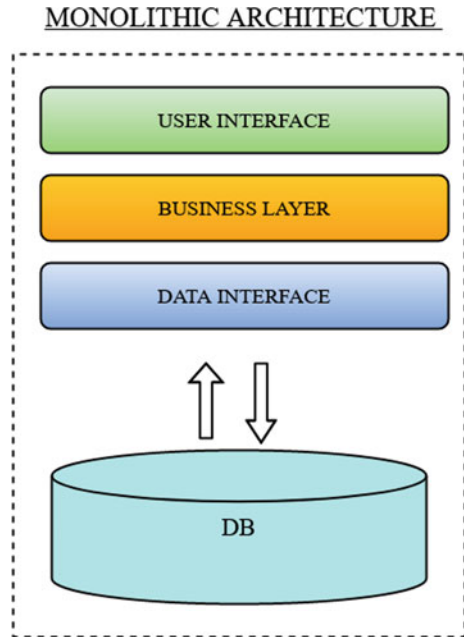
K. S. Swarnalatha (✉)

Professor, Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore 560064, India
e-mail: swarnalatha.ks@nmit.ac.in

A. Mallya · G. Mukund · R. Ujwal Bharadwaj

Student, Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore 560064, India

Fig. 1 Monolithic architecture



Data interface: Data interface performs internal DB actions called in response to queries.

Databases: A storage point for all the data which needs to be stored as shown in Fig. 1.

Characteristics

Platform Crossing: These are the ones that affect the whole application such as handling users, monitoring the spikes and drops in performances. In this architecture, the functioning concerns are specific to one app so it is easier to perform tests on for further development. Applications which are monolithic are easier to test, debug and deploy as the app is a single unit, which enables you to run total testing from end to end [2].

Deployment Efficacy: As it is a single file or folder which can be deployed without any kind of hassle.

Modular Development: If the development team is comfortable with the similar codebase, they can easily manage the process of finishing a project.

Microservices Model

While a formal definition of microservices does not exist, we can broadly define it as a framework made up of numerous services which can be deployed individually, and no service is dependent on another service to work.

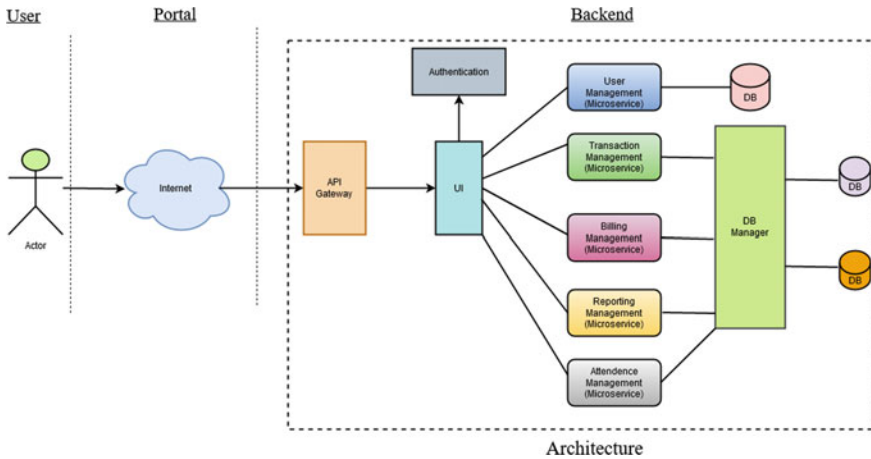


Fig. 2 Microservice model

Microservices architecture dissolve into a smaller set of independent entities. The entities carry out all the processes as a separate entity of the required service [3]. Therefore, the services in the architecture have their own unique logic to work on and a db of its own as well as a specific function to perform as shown in Fig. 2.

Architecture

In this type of architecture, the entire flow of work is separated into groups of individual modules which can interact with each other through predetermined methods called application programming interfaces (APIs). Each specific service has the capability of being independently micromanaged and scaled to specific needs of the users. Microservices offer foolproof fault segregation which means that in the case of a fault in one of the service the whole system need not stop running. Whenever the fix for the error is ready, it can be ready for deployment separately only for the designated entity of the service rather than redeployment of the entirety of the app. This architecture enables us to have a wide array of the technology stack which can be effectively suited for the required process instead of being forced into choosing an existing approach.

Characteristics

Component Independence: The required services can be redeployed any time needed and updated at will, which gives more freedom to developers. If a faulty section of the build is found in a specific microservice its impact only is felt within that particular service and this will not be recurring issue throughout the entire application which can reduce the chance of the whole architecture failing.

Scalability: It is more easier to scale the application as components are modular and can be added or removed based on the necessity.

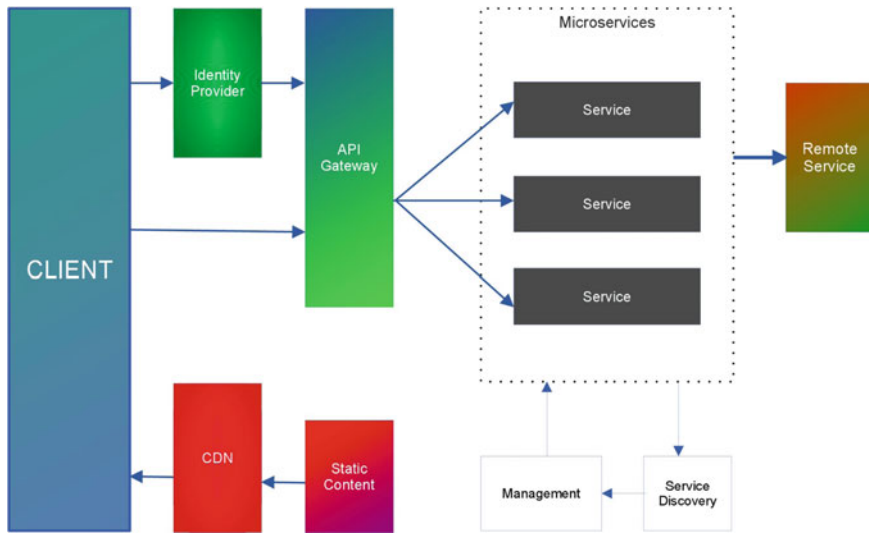


Fig. 3 Components of microservice

Fault Tolerance Is Heightened: Any error found in the microservices affects only that specific service and not the whole working system. So all the alterations and tests are executed with fewer errors and risk free.

Microservices Working Principle: The microservice working principle as shown in Fig. 3.

A general microservice architecture model is composed of

- Clients
- Identity providers
- API gateway
- Messaging formats
- Databases
- Static content
- Management
- Service discovery.

Clients

Clients are different entities sending requests and waiting for responses. Different applications from various devices can be considered as clients, which try to perform various functions like search, configure, manage, build, etc.

Identity Providers

All the requests from the clients are sent to these identity providers, which authenticate them and map those requests to the API gateway. The API gateway then takes over to communicate those requests.

API Gateway

API gateway serves as the entry point for all requests coming from clients, which process them and forward those requests to the appropriate microservice.

Advantages of using API gateway

Microservices can be modified/updated without any communication from the clients. Since API gateway is independent of clients, services can also use messaging protocols that do not adhere to web standards. API gateway can also perform certain important functions such as authentication, security and load balancing.

Messaging Formats

Services communicate through two types of messages:

Asynchronous messages

These messages are used by clients who do not wait for a response from a service. Type of the message is usually defined, and these messages have to be interoperable between implementations.

Synchronous messages

In this messaging model, a response is awaited from the service, after a request is sent by a client. Microservices commonly use the representational state transfer (REST) framework, because it is based on a stateless, client-server model and the HTTP protocol. This protocol offers a major advantage by providing a distributed environment where each functionality is provided with separate resources to carry out its operations.

Data Handling

Each microservice owns a database that is independent of all other databases they use it to capture data and operate on them using their independent functionality. Data bases of each microservices are handled by its service API only as shown in Fig. 4.

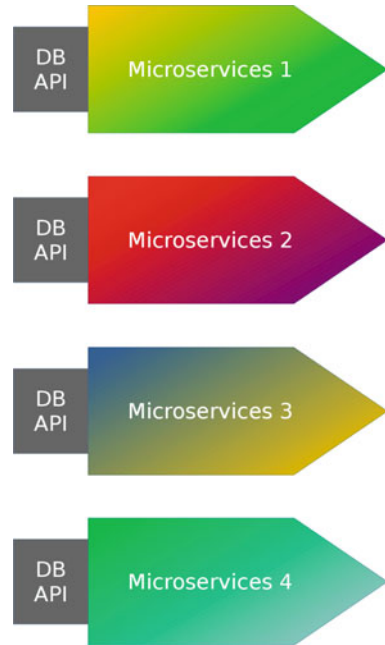
Static Content

After the services processed among the microservices and they are done communicating they send the static data response to a storage device based on cloud technology which in turn delivers them to the clients using the content delivery networks (CDNs). Apart from the core components of the architecture, there are a few other significant components they are

Management

This component deals with non-functional requirements such as load balancing, distribution of work on separate nodes and identification of failed modules.

Service Discovery

Fig. 4 DB of microservice

This component guides microservices to find routes of communication between them. It maintains a list of services on which nodes are placed.

Evolution of Uber's Architecture

Before discussing what the engineering team at Uber did and if it really solved their problems it is important to understand when does one make the shift? Microservices come with their own engineering limits and investments, so do not make the shift just because you are scaling the product to a larger cloud of users. When core domains grow and new features are introduced, separation of concerns becomes a critical point to look at the engineering team at *Uber* started experiencing a rapid growth when they were trying to bring out more services and this saw a significant increase in the developer activity. Adding more features might lead to a need to fix more bugs and resolve other technical debts which makes it extremely difficult when there is only a single repo to work on. Organizations try to expand their team with each service taking the onus of running a specific feature, independent of the other [4]. This means now a team of developers working on the payments feature do not have to depend on the status of a team working on the passenger-driver management (e.g.). Smaller teams benefit from monolithic architecture where code is consolidated, so the above point should be kept in mind when one is thinking of transitioning into microservices architecture (Fig. 5).

So what do we make of it? Microservices can be viewed as an *operational* benefit that organizations adopt at the expense of *performance*.

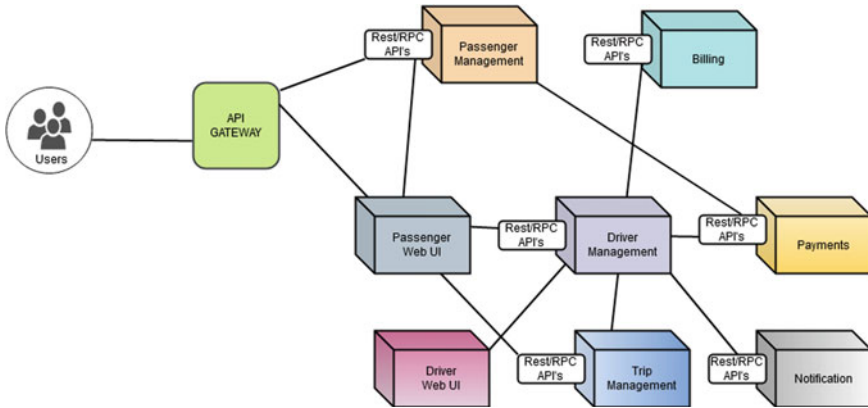


Fig. 5 Evolution of Uber’s architecture

Uber’s Transition: Uber started to grow into a larger team of engineers, with multiple teams owning pieces of the stack and they wanted to break free. Microservices would allow teams to be more autonomous and make their systems more flexible. So, Uber transitioned into a Microservice architecture that would now increase the reliability of systems, provide clear separation of concerns, Define ownerships and Simplify deployments. This was the “**service-oriented microservices architecture**” (SOA). However, as the team grew larger they began to notice some issues associated with the *complexity* of this system. Root cause analysis became extremely difficult as understanding interdependencies between services became difficult. Now, the codebase was these black boxes that could show unexpected behaviours and getting visibility into it needed the right tools, making debugging hard. When you transition from a monolithic codebase to microservices it is important to make some critical infrastructure changes. Two of them are defining sensible *contracts* and *communications*. With newer features added as individual microservices, it is important to set up endpoints between each of them for communication and have well-defined contracts for response. Uber observed that services providing REST or RPC endpoints offered weak contracts, impacting the overall **resilience** of the system. They needed a more standard way of communication that could provide *type safety*, *validation* and *fault tolerance*. Uber found that *Apache Thrift* was one such tool that met their needs best. It helps in building cross-language services, which means with the data types and interfaces defined in language-agnostic files services that are written in different languages (Python, Node, etc.) could now communicate with each other. Thrift binds services to strict contracts guaranteeing type safety. A **contract** is basically a set of rules that a service must adhere to while trying to interact with that service. It describes how to call service procedures, what inputs to provide and what response to expect. A strict contract reduces time spent on figuring out how to communicate with a particular service. Deployment of services got simple even as a microservice evolved since Thrift solved problems of safety. To handle latency and fault tolerance, Uber wrote

libraries from the ground up taking inspiration from libraries used in other companies like Netflix's *Hystrix*. But their monolithic API turned into a distributed monolithic API now. To build a simple feature, one has to work across multiple services, collaborating with teams that own them. Lines of service ownership blurred, services that appeared independent had to be deployed together on changes. Once you adopt a microservice architecture there is no turning back, you need to adjust and adapt for the larger scheme of things.

Introduction to DOMA: Over the years as Uber grew to provide 2000+ microservices they started experiencing the downsides of its complexity. Its operational benefits were too good to be rejected or replaced, with lack of alternatives in the market. Uber came up with a more generalized approach that could find a fine balance between overall system complexity and flexibility associated with microservices architecture—"domain-oriented microservices architecture" (DOMA).

Principles of DOMA

- Orientation towards domains
- Layer designs
- Well-defined gateways
- Mechanism to extend domains.

Uber's Architecture

Domains: A logical grouping of one or more microservices representing a functionality forms a domain. In Uber, there are domains of map search services, fare services and matching platforms with different gateways for each.

Layer Design: A layered design helps at separation of concerns and dependency management at scale.

They designed it keeping failure blast radius and product specificity in mind, which means the bottom layers are the ones that have more dependencies, tending to have larger blast radius while also representing more general functionalities. A layer is only dependent on the layers below it, functionality moves down the pyramid, i.e. specific to generic.

Uber's layer stack looks like this as shown in Fig. 6.

An API gateway is a way to decouple client interface from the backend implementation. Microservices communicate through API calls (RPC OR REST), so when a client makes a request the gateway routes it to the right service and produces expected response, keeping track of everything (Fig. 7).

Extensions

Extensions help when you might want to make use of the functionality of an underlying service, without affecting its implementation or reliability. Uber uses extensions for both **logic** and **data**.

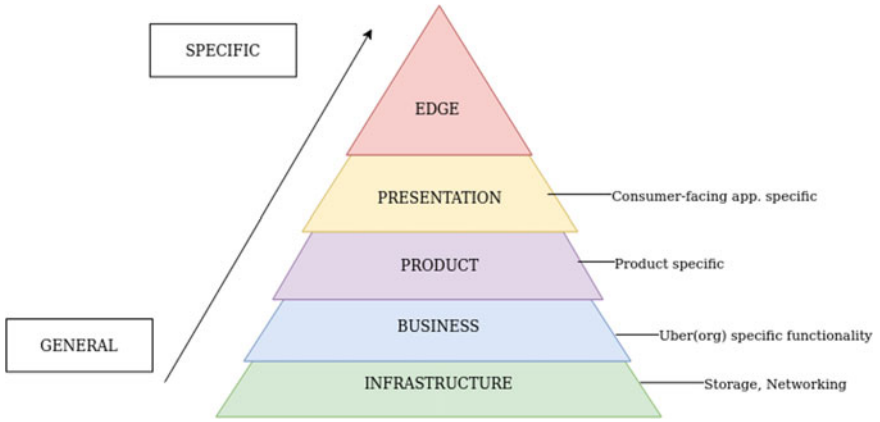


Fig. 6 Uber's layers stack

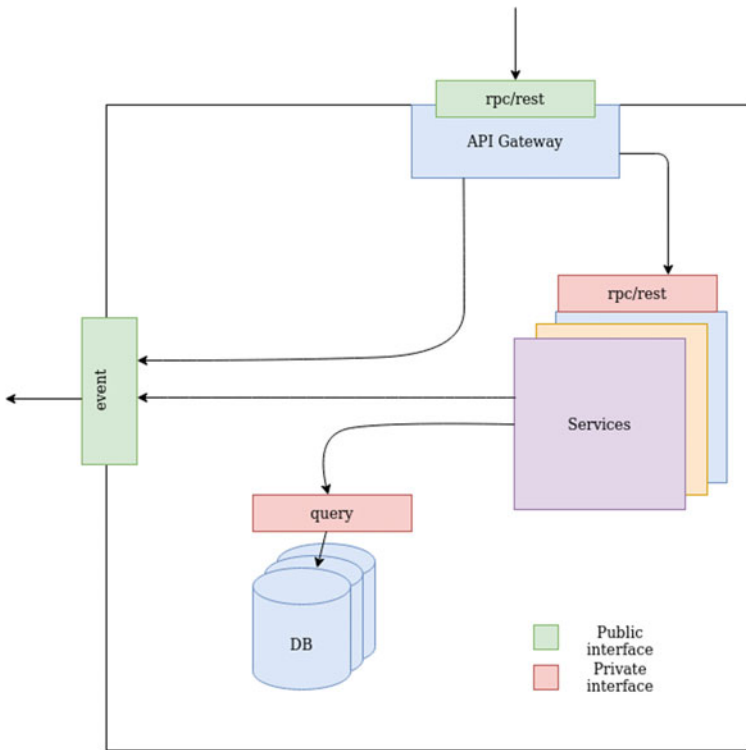


Fig. 7 Microservice API gateways

Logic extensions //talk about “go online” as example

- Uses plugin pattern.
- Interface-driven approach, i.e. teams can implement extended logic without modifying the underlying code.

Data Extensions

Data extensions for arbitrary data that can help in avoiding bloat in core data models.

Uber uses Protobuf(`google.protobuf.Any`) for arbitrary data.

DOMA tries to reduce complexities of the microservice model of development, which increases in complexity as clusters of services increase and makes the entire ecosystem inefficient. Even now, DOMA is constantly evolving and improving at Uber to suit their ecosystem, which grows each time they expand into a new domain.

2 Conclusion

Microservices architecture like DOMA has shown positive signs at Uber, simplifying their developer experience and system complexity. Product teams at Uber saw an accelerated development with reduction in time taken for code reviews, tests and planning. The onboarding time for a new feature saw a reduction by 25–50%. As Uber was stepping into offering newer services, adopting microservices architecture helped them maintain platforms easier with less expense. We think organizations that are trying to scale their team and venture into offering more services to their customers can benefit the most from this design pattern.

References

1. Di Francesco P (2017) Architecting microservices. In: 2017 IEEE international conference on software architecture workshops (ICSAW), pp 224–229. <https://doi.org/10.1109/ICSAW.2017.65>
2. <https://medium.com/nerd-for-tech/uber-architecture-and-system-design-e8ac26690dfc>
3. <https://medium.com/@narengowda/uber-system-design-8b2bc95e2cfe>
4. <http://highscalability.com/blog/2015/9/14/how-uber-scales-their-real-time-market-platform.html>

Circular Economy with Special Reference to Electrical and Electronic Waste Management in India



S. Veena , H. R. Sridevi , and T. C. Balachandra 

1 Introduction

The current population of India is 1.3 billion as of November 2021 based on Worldometer elaboration of the latest United Nations data. India population is equivalent to 17.7% of the total world population. On an average, the yearly growth rate is estimated to be 1%. At present, the Urban population in India is found to be 35.0% of the total population. The rural to urban migration or urbanization trend is expected to sustain in the next 50 years for better living. It is forecasted that by 2050, the Urban population will grow to 55%. Figure 1 shows the statistics of Urban versus Rural population in India [1].

Urbanization leads to increased usage of products and services. Hence, the modern living demands for more resources resulting in increased pollution and waste generation. The gap between the increase in the demand and limited resources calls for an alarming situation which needs to be addressed to ensure that even the future generations will have sufficient resources. This demands for a paradigm shift that results in building a new modern society based on an economic system which aims at eliminating the waste while repetitively reusing the limited resources [2].

The Circular Economy is a smart and innovative approach toward increasing the economic benefits, thereby reducing the environmental damage and assuring the efficient management of resources [3]. In this way, the life cycle of products is extended. The goal of this Circular Economy concept is to decouple economic

S. Veena (✉) · H. R. Sridevi · T. C. Balachandra
Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: veena.s@nmit.ac.in

H. R. Sridevi
e-mail: sridevi.hr@nmit.ac.in

T. C. Balachandra
e-mail: balachandra.tc@nmit.ac.in

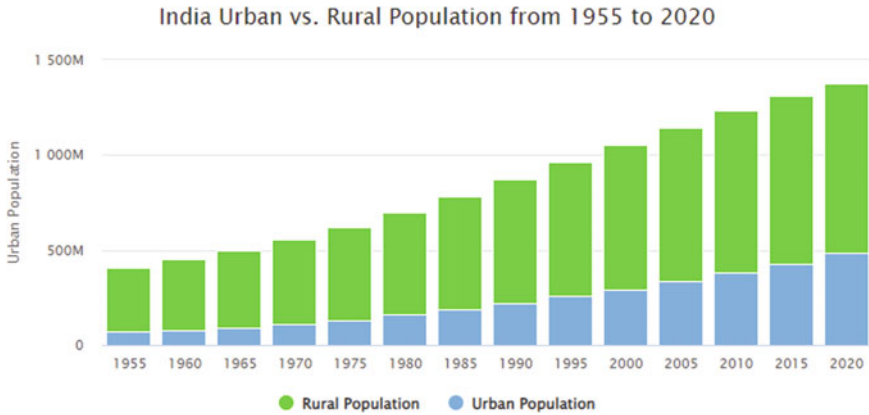


Fig. 1 Urban versus rural population from 1955 to 2020. *Source* <https://www.worldometers.info/demographics/india-demographics/#urb>

growth from resource consumption [4]. The Circular Economy attempts to increase resource productivity, reduce energy consumption, and reduce greenhouse gas emissions by using the 3R (Reduce, Reuse, Recycle) waste management method [5]. In recent times, the concept of Circular Economy is gaining popularity in the field of sustainability due to the evident increase of demand–supply imbalance [6]. Besides, roof-top solar power generation has been the order of the day and regulatory bodies and the Government has encouraged it in a big way. Many office spaces with relatively a smaller number of floors have installed Solar Photovoltaic Systems (SPV) [7]. Earlier, the waste generated by discarded electrical machinery and huge metallic debris was scrapped and allowed to go as land use waste. This was nothing but following the “Linear Economy Model” where no thought was given to recycling or upcycling. Yet another upcoming area that is contributing to Circular Economy is making scaled models from industry scrap. This involves converting used spare parts into highly creative models/artifacts that can be displayed. Such models can be either working models or non-working models. This activity that has picked up recently contributes in a big way toward “land use waste minimization” [8]. The information reported in the present work focuses on these aspects.

2 Concept of Circular Economy

Circular Economy is a model that has brought a change in the Linear economy. It is a transition toward the utilization of renewable energy (non-conventional sources of energy), elimination of harmful chemicals, and waste by implementing better design of products. Its prime emphasis is on reusing, recovering, remanufacturing, and regeneration of products and materials after the lifetime of the product. India is rich in Iron Ore and Bauxite, yet it is import dependent for some of the rare earth metals

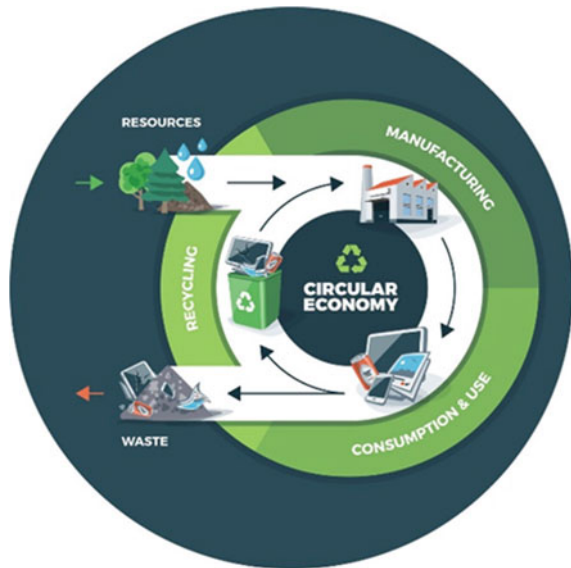
which are required for manufacturing Electrical and Electronics Equipment's (EEE). Further, electrical and electronic equipment are vastly used in automobiles that can be salvaged after its useful life and can be converted into artifacts of high aesthetic value. It is known that the cost incurred in extracting the raw materials is more than the rate of their occurrence in its crude form. So instead of investing in mining raw materials, it is recommended to use secondary raw materials which can be derived from the e-waste. This alternate to mining raw materials leads to resource security and environmental sustainability. There is a need for modern tools/techniques/practices which focuses on limiting the use of nonrenewable resources and explores the usage of secondary resources which can be recovered from the waste and hence resulting in transparency, socio-economic, and environmental benefits. This Circular Economy approach builds connection among various stake holders across the value chain.

Thus, in short, Circular Economy can be defined as a system which comprises of 3 aspects (3R) namely:

1. To Reduce the usage of nonrenewable and toxic materials thereby transiting toward utilizing the renewable energy.
2. To Reuse the products which still are in working condition by adopting better designs.
3. To Recycle the e-waste into new resources for further utilization [9].

In this paper, Circular Economy concept in managing e-waste for Electrical and Electronic Equipment is highlighted. Now, in recent years, there is a paradigm shift in the thinking process of individuals, organizations, and leaders. This new shift has been toward the "Circular Economy" where at least a part of the discarded material is reused in some way (Fig. 2).

Fig. 2 Circular Economy model *Source* solarimpulse.com



3 Circular Economy Benefits

- Environment protection, waste reduction, and planned recycling are the major benefits of a Circular Economy.
- CE creates employment opportunities and enhances the economic growth.
- Consumers are benefitted by adapting Circular Economy in their purchases.
- Consumption of nonrenewable resources is reduced.
- Land use waste is minimized.

4 Circular Economy Challenges and Solutions

Our Planet Earth is very delicate with limited natural resources and a balanced ecosystem. But due to urbanization, there is a huge demand for creation of new products for which naturally available finite resources are exploited and the used products are disposed unsystematically. This is “cradle-to-grave” approach which is mainly consumption driven. However, with the increase in the population, the gap between the demand and the supply grows thereby increasing the demand for the finite resources. Hence, an awareness should be brought in to replace the linear economic model which is oriented toward take-make-dispose (cradle-to-grave) approach to a Circular Economy model which adopts cradle-to-cradle approach [4].

There is a transition in the socioeconomic, ecological balance, advancement in the technology, rapid growth in population, global awareness regarding the finite nature, and rapid depletion of resources. Linear to Circular Economy transformation is expensive. To achieve Circular Economy, financial incentives should be provided.

The production process should be designed to obtain a better product with high performance. Reduction in per unit production cost will not only increase the production but also increase the consumption levels and thereby override the environmental benefits and eco-effectiveness. Usage of the product must be made efficient throughout its life cycle. A proper technique should be devised for reusing the product before disposal. Few more solutions that can be thought of to achieve Circular Economy include sharing the resources, adapting to a new purpose, reprocessing, and categorizing the waste into organic and inorganic materials and making use of Renewable energy sources. Also, the fundamental approach to Circular Economy is to stop producing the products in surplus and redundant products. However, it is observed that recycling the products is the most practiced solution to attain Circular Economy.

Proper framework must be designed to promote the concept of Circular Economy both at the national and international level. Dependency on inter-sectoral, inter-organizational changes can be avoided by restructuring the societal and institutional policies. Awareness must be created about the new tradition of consumption.

4.1 Recommendations to Incorporate Circular Economy in E-Waste Management

This section presents the upcycling strategies for implementation of CE in managing the waste Electrical and Electronic equipment. Various ways to handle the challenges in attaining CE at every stage are proposed [9]. Figure 3 shows the process flow for incorporating CE in E-waste management.

Stage 1: Acquisition of raw materials for EEE—India is advancing in the field of electronics by setting up its own manufacturing facilities. In this context, critical metals such as gallium, germanium, selenium, and indium-tellurium which are very essential for production should be explored from e-waste.

- Promoting technology to track the availability of critical materials extracted from mining e-waste.
- Incentives can be awarded for producers who use certain percentage of critical materials from secondary resources.

Stage 2: Product design and manufacturing—Lot of research work has been carried out to overcome the problems caused by mounting e-waste. Many upcoming product designers and researchers are working toward converting discarded automobile spares into artifacts that transform into display material carrying very high aesthetic value [4]. Hence, product design and component manufacturing stages must work effectively to tackle the electronic waste.

- Eco-friendly products should be designed to ensure longer product life and feasible for recycling.



Fig. 3 Process flow to incorporate CE in E-waste management

- Incentives can be awarded for manufacturers who adopt Circular Economy measures in the design and production stages.
- Scholarships and fellowships to promote design thinking and creativity in the student community are essential.

Stage 3: Consumption—Environmental sustainability is one of the key aspects in achieving Circular Economy in consumption stage. Hence, awareness on segregation of waste should be brought in consumers as they play an important role in the disposition of E-Waste.

- By eco labeling the products, the consumers will gain confidence about the recycled materials used in the product. Hence, Eco labeling facilitates the consumers to make right decision in choosing the products and thereby minimizing the waste.
- A strategy/awareness on managing the e-waste, Circular Economy practices, and strategic obsolescence are required.

Stage 4: End-of-life electrical and electronics products collection systems—A system to collect and recycle the E-waste is of high importance as the E-waste is growing exponentially. Consumers should be facilitated with an efficient collection network so that they get access to dispose the E-waste at their comfort.

- Promotion of Buy back/resale schemes for products with proper guidelines.
- Establishing mechanisms to ensure the collection of waste fractions which are toxic and most hazardous.

Stage 5: Recovery stage—This is a very crucial stage wherein the used electronic products instead of being disposed, are collected, and are dismantled to find any parts of the product or metals which can be reused or recycled. Hence, the collected E-waste should be taken to professional dismantlers and specialized recyclers who can recover the precious metals from the E-waste with proper environmental safeguards. To achieve Circular Economy and increase resource efficiency in exploiting the critical raw materials, the existent technologies should be improved post end-of-life of products.

- To ensure effective use of E-waste, some standards should be imposed for dismantling and recycling of the E-waste.
- To promote green approach. As lot of energy is consumed in collecting the E-Waste and reusing them, utilization of renewable energy should be encouraged thereby promoting Circular Economy.

The government has been working hard to develop policies and projects that would help the country transition to a Circular Economy. Committees for various emphasis areas have been constituted to hasten the country's transition from a linear to a Circular Economy. The committees will be led by the concerned line ministries and will include domain experts, academics, and industry representatives as presented in Table 1. In their particular emphasis areas, the committees will produce thorough action plans for shifting from a linear to a Circular Economy. They will also take out

Table 1 Emphasis areas of end-of-life goods and their in-charge ministry (Indian context)

End-of-life products	In-charge government bodies
Municipal solid waste and liquid waste	Ministry of Housing and Urban Affairs
Scrap metal (ferrous and non-ferrous)	Ministry of Steel
Electronic waste	Ministry of Electronics and Information Technology
Lithium ion (Li-ion) batteries	NITI Aayog
Solar panels	MNRE
Gypsum	Department for Promotion of Industry and Internal Trade
Toxic and hazardous industrial waste	Department of Chemicals and Petrochemicals
Used oil waste	Ministry of Petroleum and Natural Gas
Agriculture waste	Ministry of Agriculture and Farmers' Welfare
Tyre and rubber recycling	Department for Promotion of Industry and Internal Trade
End-of-life vehicles (ELVs)	Ministry of Road Transport and Highways

Source pib.gov.in

the appropriate processes to guarantee that their findings and recommendations are implemented effectively [10].

Even end-of-life goods, recyclable materials, and wastes are among the priority areas, which either continue to represent significant issues or are emerging as new difficulty areas that must be addressed holistically.

4.2 Roadmap for Recycling Solar Panels—A CE Approach

In the coming years, India's vision should be in the direction of a steady shift toward renewable energy sources and reduction of carbon emissions. However, this transition may not be easy as lot of challenges such as optimization of energy saving and reduction in energy demand must be met. Modern solar energy systems and materials are a key factor in addressing these challenges. Circular Economy concept can be used to tackle the critical situation of recycling the large mass of PV waste [7].

Recycling processes of crystalline silicon (c-Si) modules leads to cost saving, thereby resulting in sustainability of the supply chain in the long run. This is expected to influence recovery of energy and embedded materials, while lowering CO₂ emission and energy payback time for the PV industry.

Stage 1—Extracting the raw materials is expensive as compared to manufacturing and availability of raw materials is limited. Hence, materials used for manufacturing PV panels are selected such that they are not only efficiently used but can also be reclaimed. Cadmium, molybdenum, gallium, selenium, indium, tellurium, silver, and silicon are majorly used in production of PV cells. So, to minimize the use of these raw materials, an effective method should be used to extract these semiconductor materials which are found in thin film PV production waste. One of the technologies employed to recycle used solar panels is based on “Dielectric Heating.”

Stage 2—An effective solar cell design or a new freedom of design should be encouraged for development of solar-powered products to ensure longer product life with good performance. The process devised in manufacturing the solar products should take care of all the needs of the clients and take essential steps to design the modules on the fly. The manufacturers should adopt Circular Economy in the design and manufacturing of the PV panels. Also, the manufacturing companies are recommended to upgrade their products and their productions by full integration of PV materials, enhancing the receiving surface of the PV module and hence improving the efficiency of the PV module.

Stage 3—To achieve sustainable future, consumer awareness regarding the Green India and make in India should be brought in and should address the main target stakeholders namely PV systems suppliers, PV installers and service providers for energy-efficient buildings, training providers, solar-powered consumer users, construction companies’ associations, real estate promoters, energy management agencies, local authorities, and national/regional public bodies. People should be encouraged to install PVs on building roofs and thereby utilize solar energy to electrify the houses and to heat water.

Stage 4—Resale of the solar products should be encouraged to minimize production of new products. The companies should promote buy back schemes to ensure the collection of the waste solar products. Similarly, a proper system is to be set up to obtain a collection of around 90% of the PV waste throughout the country. PV waste also includes the accessories such as the cables, junction box, and frame, creating awareness among people on clever segregation of the PV panel waste and Circular Economy practices.

Stage 5—Instead of disposing the end-of-life PV panels, broken solar cells, production scrap, etc., a simple and appropriate mechanism should be devised to dismantle them and recover the elements such as CIS, CIGS, GaAs, or CdTe to produce marketable products. However, it is mostly the glass which is obtained from the thin film PV scrap. Hence by applying mechanical stress (laser or vibration) on the waste mixture of the brittle PV cells, silicon can be easily separated. Some of the materials which cannot be separated physically can be chemically treated. Similarly, the expensive materials such as In or Ag obtained from recycling can be re-entered into recycling process or can be sold to market. Hence, the new thin film PV modules can be produced from the materials obtained from recycling. The rate at which solar panels are installed is phenomenally increasing, and the rate at which



Fig. 4 PV cycle recycling process for c-Si modules—summary

they are reaching their end of life is also increasing. By 2050, this figure is expected to touch 5.5–6 million tons. A typical generic process for recycling crystalline silicon solar module is shown in Fig. 4.

5 Conclusion

This paper presents an overview of possible adaptation of Circular Economy through activities and practices and initiatives involving recycling and upcycling. The work reported is expected to influence the current mindset of individuals and corporates to move from a linear economy model toward a Circular Economy model. Some components of the PV recycling process for solar modules are discussed. The recycling of c-Si modules saves money, ensuring the supply chain’s long-term viability. This projected “birds eye view” with a motto of taking a paradigm shift toward Circular Economy is expected to have an impact on energy and integrated material recovery, as well as reduction in PV installation payback time. PV waste management has the potential to create new avenues for industry improvement and provide employment opportunities for both public and private sector partners, in addition to having a good impact on the environment. PV unit recycling can remove and keep potentially dangerous compounds (such as lead, cadmium, and selenium), as well as recover rare materials (such as silver, tellurium, and indium) for future use.

References

1. <https://www.worldometers.info/demographics/india-demographics/#urb>
2. D’Amato D, Korhonen J (2021) Integrating the green economy, circular economy and bioeconomy in a strategic sustainability framework. *J Ecol Econ* 188. <https://doi.org/10.1016/j.ecolecon.2021.107143>
3. Barrie J, Schröder P (2021) Circular economy and international trade: a systematic literature review. *Circ Econ Sustain*. <https://doi.org/10.1007/s43615-021-00126-w>
4. Goyal S, Kapoor A, Esposito M (2016) Circular economy business models in developing economies—lessons from India on reduce, recycle, and reuse paradigms. *Thunderbird Int Bus Rev* 60(5):729–740. <https://doi.org/10.1002/tie.21883>
5. Ramakrishna S (2020) Circular economy and sustainability pathways to build a new-modern society. *Dry Technol*. <https://doi.org/10.1080/07373937.2020.1758492>
6. Circular economy in electronics and electrical sector. Ministry of Electronics and Information Technology, Government of India, New Delhi

7. Circular economy: recent trends in global perspective (2021) Springer Science and Business Media LLC
8. Yi S, Lee H, Lee J, Kim W (2019) Upcycling strategies for waste electronic and electrical equipment based on material flow analysis. *Environ Eng Res.* <https://doi.org/10.4491/EER.2018.092>
9. Brenner W, Adamovic N (2020) Creating sustainable solutions for photovoltaics. In: 2020 43rd international convention on information, communication and electronic technology (MIPRO), pp 1777–1782. <https://doi.org/10.23919/MIPRO48935.2020.9245369>
10. Brenner W, Adamovic N (2016) The European project solar design illustrating the role of standardization in the innovation system. In: 2016 39th international convention on information and communication technology electronics and microelectronics (MIPRO)

Analysis and Evaluation of Pre-processing Techniques for Fault Detection in Thermal Images of Solar Panels



Sujata P. Pathak and Sonali A. Patil

1 Introduction

Thermal imaging or infrared imaging is the process of converting infrared radiation, i.e., heat into visible images. Thermal images represent the spatial distribution of temperature differences in an image viewed by a thermal camera. Using an infrared camera, temperature differences in a photovoltaic module are detected and are visualized in a thermal image. During thermal image acquisition, there are chances that lot of noise may get introduced in thermal images. To process these images further in the tasks such as segmentation, classification, and detection, it is necessary to filter them using suitable filters. Pre-processing of thermal images is first step consisting of filtering, image enhancement and segmentation.

Rest of the paper is organized as follows: Sects. 2–4 covers previous work done in this area along with details about filters and histogram equalization techniques. Various performance measures used for evaluation are mentioned in Sect. 5. Section 6 discusses the analysis of experiments and results after filtering and histogram equalization techniques. Finally, Sect. 7 presents conclusion.

2 Previous Work

Researchers have compared performance of various filters on different sets of data by adding various types of noise such as salt and pepper, Gaussian, and speckle

S. P. Pathak (✉) · S. A. Patil
K J Somaiya College of Engineering, Vidyavihar, Mumbai, India
e-mail: sujatapathak@somaiya.edu

S. A. Patil
e-mail: sonalipatil@somaiya.edu

[1–11]. Wahab et al. [12] have done comparative evaluation of image filtering and contrast stretching of individual channels of color thermal images. The effectiveness of individual filters is evaluated using MSE and PSNR. Three image filtering techniques as median filter, Gaussian filter and Wiener filter are used along with contrast stretching. Performance of four filters such as median, mean, Wiener, and adaptive filter is compared in the paper [13] on the basis of the quantitative parameters such as PSNR and RMSE on MR images. In the paper [14], existing denoising algorithms such as wavelets based approach and filtering approach are reviewed and performance is compared against bilateral filters. Various noise models are used to describe multiplicative and additive noise in an image. Results of the degraded noisy images are analyzed in terms of MSE, PSNR, and universal quality identifier (UQI).

3 Noise Removal Using Filters

3.1 Bilateral Filter

Bilateral filter is one type of nonlinear filter proposed by Tomasi and Manduchi [15]. This technique reduces additive noise from image and smoothens the images along with preserving edges. A weighted sum of local neighborhood pixels is considered which depend on both intensity difference and spatial distance. Hence, noise is averaged out and edges also are well preserved. Bilateral filter's output at a specific pixel location x is computed mathematically as shown in Eq. 1.

$$\tilde{I}(x) = \frac{1}{C} \sum_{y \in N(x)} e^{-\frac{(y-x)^2}{2\sigma_s^2}} e^{-\frac{(I(y)-I(x))^2}{2\sigma_r^2}} I(y) \quad (1)$$

3.2 Wiener Filter

Wiener filter is linear type of filter, which is applied to an image adaptively, customizing itself to the regional image variance. Wiener performs slight smoothing when variance value is large and more smoothing when variance value is less. This method usually produces improved results compared to linear filtering. The adaptive filter is more selective compared to linear filter, which preserves edges along with other high-frequency components of an image. This filter gives good results when the noise is "white" noise, such as Gaussian. It is not suitable for image containing more edges. For images affected by blurring and additive noise, Wiener filter is considered as the best MSE-optimal stationary linear filter. More computation time is required than linear filtering since they are applied in frequency domain [11].

3.3 Mean Filter

Mean filtering is an easy and simple method frequently used for smoothing and noise reduction in images. It does this by decreasing amount of intensity difference between two neighboring pixels. This method replaces each pixel value with the average or mean value of its neighbors, including itself. This will replace all eliminating pixel values which are not representative of their neighborhood. This filter is based on a kernel like a convolution filter. During mean calculation, it uses the shape and size of the neighborhood. For an average smoothing 3×3 square mask is more common, but for more severe smoothing, larger masks such as 5×5 squares can be used [3].

3.4 Gaussian Filter

Many of the graphics softwares use Gaussian filter for reduction of image noise by blurring an image. The smoothing 2D convolution operator, i.e., Gaussian operator is used to blur images and remove noise. The probability distribution for noise is defined by the Gaussian function. The image structures can be enhanced by Gaussian smoothing [16]. In 2D, circularly symmetric Gaussian has the form:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2)$$

3.5 Median Filter

Median filter preserves useful detail in the image compared to mean filter by considering its nearby neighbors. Each pixel is replaced with the median of neighboring pixels. The midpoint is calculated by arranging all the neighborhood pixel values into numerical order. The pixel under consideration is replaced with the midpoint pixel value. High spatial frequency details are passed while remaining highly effective at removing noise on images. For the images corrupted with Gaussian noise, median filtering may not be much effective at removing noise. Also, it is relatively costly and complicated to calculate. All the values in the neighborhood are sorted which makes processing very slow [3].

4 Contrast Enhancement Using Histogram Processing

Fundamental technique for image enhancement is histogram processing. Histograms are easy to compute, thus making this technique a common tool for all image processing applications [15].

4.1 Histogram Equalization

Histogram equalization is pre-processing technique used to improve contrast in images. Contrast enhancement is achieved by efficiently distributing most repeated intensity values, i.e., by extending the intensity span of the image. The probability of occurrence of intensity level r_k in a digital image is approximated by

$$P_{r(r_k)} = \frac{n_k}{MN} \quad (3)$$

where MN are total number of pixels in the image, and n_k denotes number of pixels with intensity r_k . The discrete form of the transformation for histogram equalization is

$$S_k = T(r_k) = (L - 1) \sum_{j=0}^k p_r(r_j) \quad (4)$$

$$k = 0, 1, 2, \dots, L - 1$$

where L is the number of gray levels in the image (256 for an 8-bit image). Equalized image is obtained by using Eq. 2 for mapping of every pixel in the input image with intensity r_k to a subsequent pixel with level s_k in the output image. This process is called as histogram equalization or histogram linearization transformation [17–22].

4.2 Contrastive Limited Adaptive Equalization (CLAHE)

CLAHE is different than adaptive histogram equalization with respect to contrast limiting. In this technique, transformation function is developed by applying contrast regulating procedure to each neighboring pixel. It prevents over amplification of noise which happens in adaptive histogram equalization. In CLAHE, the output value for a pixel is its rank in a histogram of pixel intensity values in the contextual region. This is like counting the number of pixels with intensities less than the affected pixel in the contextual region. The contrast effect is limited by clipping the histogram at a predefined value before computing the CDF. This clip limit varies depending on the size of the neighborhood region or normalization of the histogram. Commonly

used clip limit value is between 3 and 4. Obtained histogram is the actual histogram of recorded intensities centered at the pixel in consideration. But, it is clipped at a particular height with the clipped pixels redistributed uniformly across all intensities in the range of the recorded image. Due to clipping, improvement of noise is reduced in comparatively similar areas of the image by changing the highest possible level of contrast enhancement [21, 23–25].

4.3 Mean Preserving Bi-Histogram Equalization (BBHE)

In this method, initially an input image is decomposed into two parts based on the mean of the input image. First image part consists of samples less than or equal to the mean whereas the other one consists of samples greater than the mean. Then, this method equalizes the two parts of an image autonomously based on their respective histograms with the restriction that the samples in the first set are mapped between minimum gray level and the mean and the samples in the second set are mapped between mean and maximum gray level. So, one of the sub-sections of image is equalized up to the mean value and the other sub-section is equalized from the mean based on the respective histograms. The resulting equalized sub-images are bounded by each other around the input mean, due to which mean brightness is preserved [26].

4.4 Equal Area Dualistic Sub-Image Histogram Equalization (DSIHE)

The basic concept behind DSIHE and BBHE is similar. In DSIHE histogram is separated based on gray level with cumulative probability density equal to 0.5 which is mean in case of BBHE. In DSIHE, original image is separated into two equal area sub-sections based on its gray level probability density function. Two sub-sections are equalized individually and then the equalized sub-sections are composed into one image. This algorithm enhances visual information in an image. It also prevents a big shift of average luminance of original image [27].

4.5 Brightness Preserving Dynamic Fuzzy Histogram Equalization (BPDFHE)

This algorithm uses the notion of smoothing a global image histogram using Gaussian kernel. After this valley regions are segmented for dynamic equalization. BPDFHE [28, 29] handles the image histogram in such a way that no remapping of the histogram

peaks is required. Only redistribution of the gray-level values in the valley portions between two successive peaks takes place. It consists of following working steps:

- (A) Fuzzy Histogram Computation.
- (B) Partitioning of the Histogram.
- (C) Dynamic Histogram Equalization of the Partitions.
- (D) Normalization of the Image Brightness.

5 Performance Measures

Consider an image of size $A \times B$. If $x(i, j)$ is the original image and $y(i, j)$ is the noisy image. Based on this, different measurement parameters can be defined as below:

5.1 Peak Signal-to-Noise Ratio

PSNR is represented as the ratio of maximum power in the image to the corrupted noise in the image [16]. The unit of PSNR is dB (decibels). If the PSNR value is higher, then the quality of the filtered image will be good. The formula to calculate PSNR is

$$\text{PSNR} = 10 \log_{10} \left(\frac{R^2}{\text{MSE}} \right) \quad (5)$$

5.2 Mean Square Error

It is defined as:

$$\text{MSE} = \frac{1}{A * B} \sum_{i=1}^A \sum_{j=1}^B [x(i, j) - y(i, j)]^2 \quad (6)$$

5.3 Signal-to-Noise Ratio

SNR or S/R is the ratio of signal power and noise power. It compares signal level with noise level. It is given by

$$\text{SNR} = S/N \quad (7)$$

5.4 Structural Similarity Index

SSIM metric indicates image quality degradation affected by operations such as losses in data transmission or data compression. Two images as reference and processed image from same image capture are required. SSIM measures the perceptual difference between two similar images [30]. A specific form of the SSIM index is

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (8)$$

5.5 Absolute Mean Brightness Error (AMBE)

It is described as the absolute difference between the input image mean and the output image mean [22]. It can be calculated as:

$$\text{AMBE} = |E(X) - E(Y)| \quad (9)$$

6 Comparative Analysis and Results

In this research, total 1487 thermal images of solar panel containing hotspots of various categories are considered. These images are captured under different environmental conditions and at different sites. Some sample images are shown in Fig. 1. Sample thermal color image and images after application of various filters are shown in Fig. 2. Image quality is affected by noise during thermal image capturing. Various filters are applied on thermal images with different faults. Performance analysis of different filters is done using statistical parameters such as PSNR, SNR, MSE, and SSIM.

6.1 Filter Performance

- A. **MSE:** Table 1 shows MSE values for various images after application of five filters on five thermal images belonging to five different faults. Graphical representation of the same is shown in Fig. 3. Bilateral filter gives minimum values for mean square error for the sample images tested.

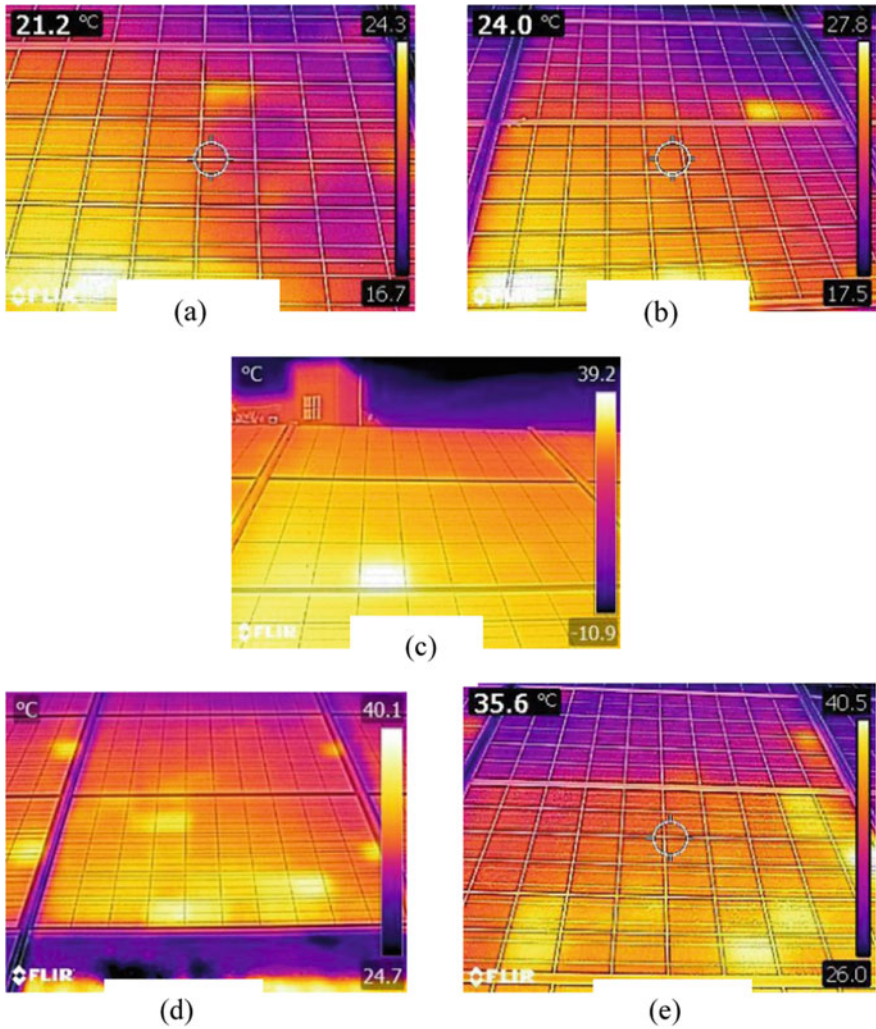


Fig. 1 Thermal images with different types of hotspots

- B. **PSNR**: For good filtering effect, the values of PSNR should be high. The PSNR values along with graphical representation are as shown in Table 2 and Fig. 4, respectively. Bilateral filter gives highest PSNR values.
- C. **SNR**: Signal-to-noise ratio should be high after application of filters. The SNR values are shown in Table 3 along with graphical representation in Fig. 5. From the values of SNR, bilateral filter gives good results.
- D. **SSIM**: SSIM is used for measuring the resemblance between two images. SSIM values range between 0 and 1; one indicating highest similarity. SSIM values

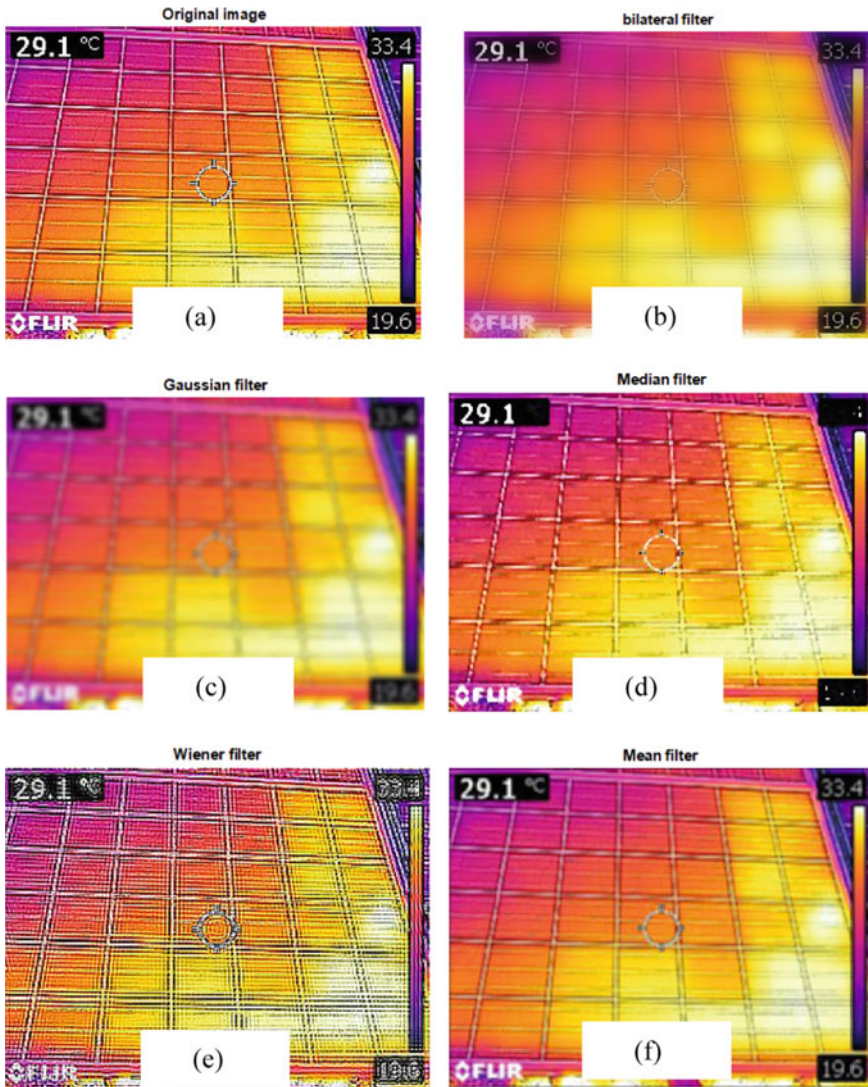


Fig. 2 Thermal images **a** original image, **b** bilateral filtered image, **c** Gaussian filtered image, **d** median filtered image, **e** Wiener filtered image, **f** mean filtered image

after application of five different filters are shown in Table 4 along with graphical representation in Fig. 6. Bilateral filter gives highest values for SSIM.

Comparing the MSE, PSNR, SNR and SSIM values, it can be proved that bilateral filter performs well in all the cases for noise removal. It gives higher values of PSNR, minimum value for MSE and highest value for SSIM. SSIM value closer to 1 indicates that image quality is very good after filtering.

Table 1 MSE values of filtered images

Images	Bilateral filter	Wiener filter	Mean filter	Gaussian filter	Median filter
IMG1	148.964	1856.90	348.629	650.2182	259.857
IMG2	97.9311	1479.40	299.784	482.8619	227.798
IMG3	164.18	2801.30	489.034	875.5435	392.717
IMG4	121.2	1953.40	362.346	631.6767	291.632
IMG5	137.042	2651.40	440.666	788.2662	337.248

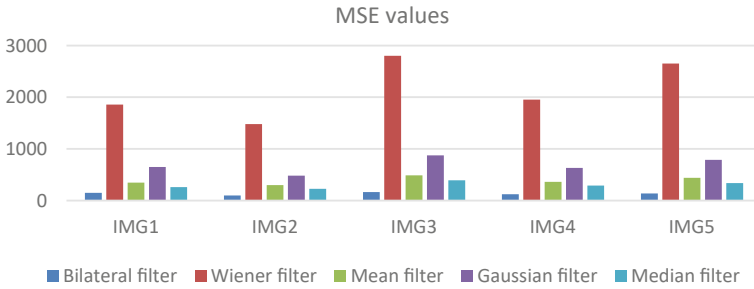


Fig. 3 Graph of MSE values for five images

Table 2 PSNR values of filtered images

Images	Bilateral filter	Wiener filter	Mean filter	Gaussian filter	Median filter
IMG1	26.4	15.4428	22.7072	20.0002	23.9835
IMG2	28.2216	16.4301	23.3627	21.2926	24.5553
IMG3	25.9776	13.6573	21.2374	18.708	22.19
IMG4	27.2958	15.2229	22.5396	20.1259	23.4825
IMG5	26.7623	13.8961	21.6897	19.1641	22.8513

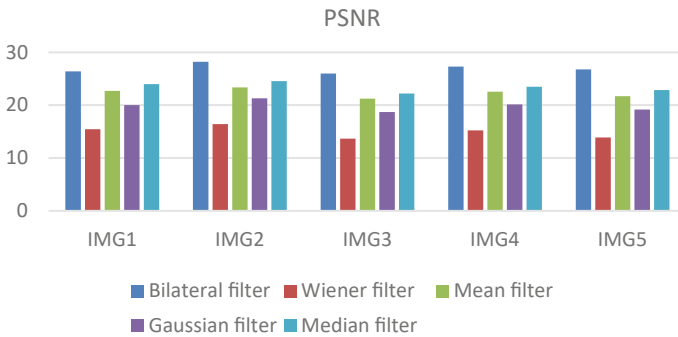


Fig. 4 Graph of PSNR values for five images

Table 3 SNR values of filtered images

Images	Bilateral filter	Wiener filter	Mean filter	Gaussian filter	Median filter
IMG1	21.9261	10.9606	18.1755	15.4332	19.5837
IMG2	24.7548	12.7865	19.8423	17.7676	21.1596
IMG3	22.0338	9.6457	17.221	14.6609	18.3258
IMG4	23.421	11.1544	18.5994	16.1694	19.6912
IMG5	21.8332	9.0205	16.673	14.1058	17.9813

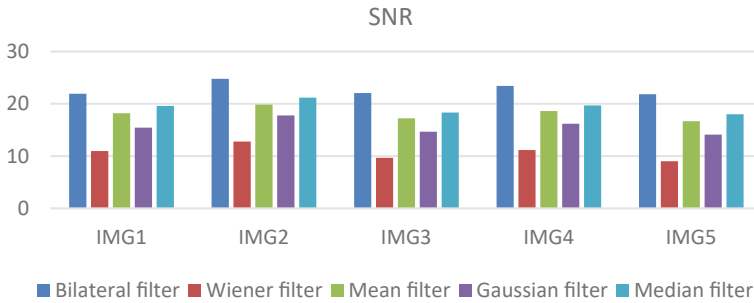


Fig. 5 Graph of SNR values for five images

Table 4 SSIM values of filtered images

Images	Bilateral filter	Wiener filter	Mean filter	Gaussian filter	Median filter
IMG1	0.9833	0.8343	0.9586	0.9218	0.9722
IMG2	0.9882	0.8624	0.9609	0.9343	0.9722
IMG3	0.9777	0.7411	0.937	0.881	0.9527
IMG4	0.9847	0.8174	0.9515	0.9118	0.9627
IMG5	0.9827	0.7599	0.9451	0.9002	0.962

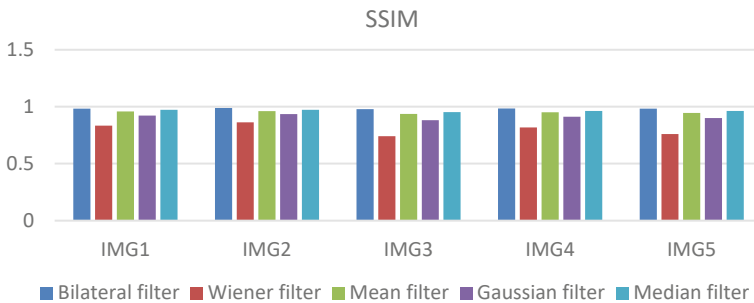


Fig. 6 Graph of SSIM values for five images

6.2 Histogram Equalization

After filtering the images with bilateral filter, five different histogram equalization techniques are applied on these images.

The sample original thermal color and bilateral filtered image along with gray scale image and histogram are shown in Fig. 7. On the bilateral filtered image various histogram equalization techniques are applied as shown in Fig. 8. These histogram techniques are applied on all the available thermal images and the HE technique providing good quality is selected as one of the pre-processing techniques.

The performance of various histogram equalization techniques is evaluated using PSNR, AMBE, and SSIM values. As shown in Table 5 and Fig. 9, BPDFHE gives highest values of PSNR. Higher PSNR values indicate good image quality.

As per the results shown in Table 6, it may be observed that BPDFHE gives least AMBE values. After looking at average AMBE values shown in last row of table, BPDFHE gives minimum value compared to other methods. Graphical representation is shown in Fig. 10.

SSIM values for ten different types of thermal images are shown in Table 7 along with graph in Fig. 11. SSIM assesses image quality and ranges from 0 to +1. +1

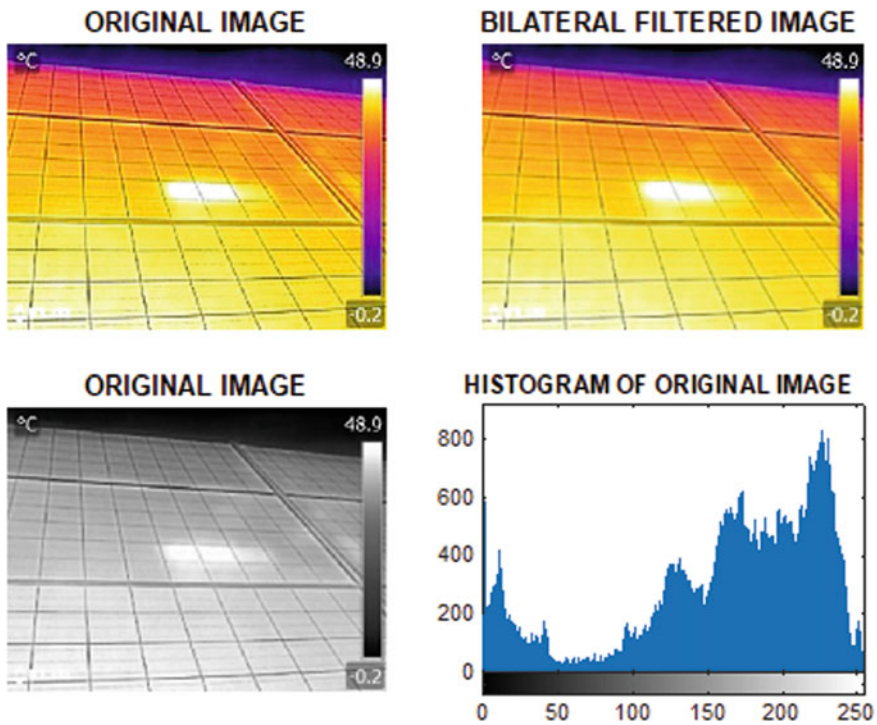


Fig. 7 Original thermal and gray image for one type of fault

Fig. 8 Results of various histogram equalization techniques **a** HE, **b** CLAHE, **c** BBHE, **d** DSIHE, **e** BPDFHE

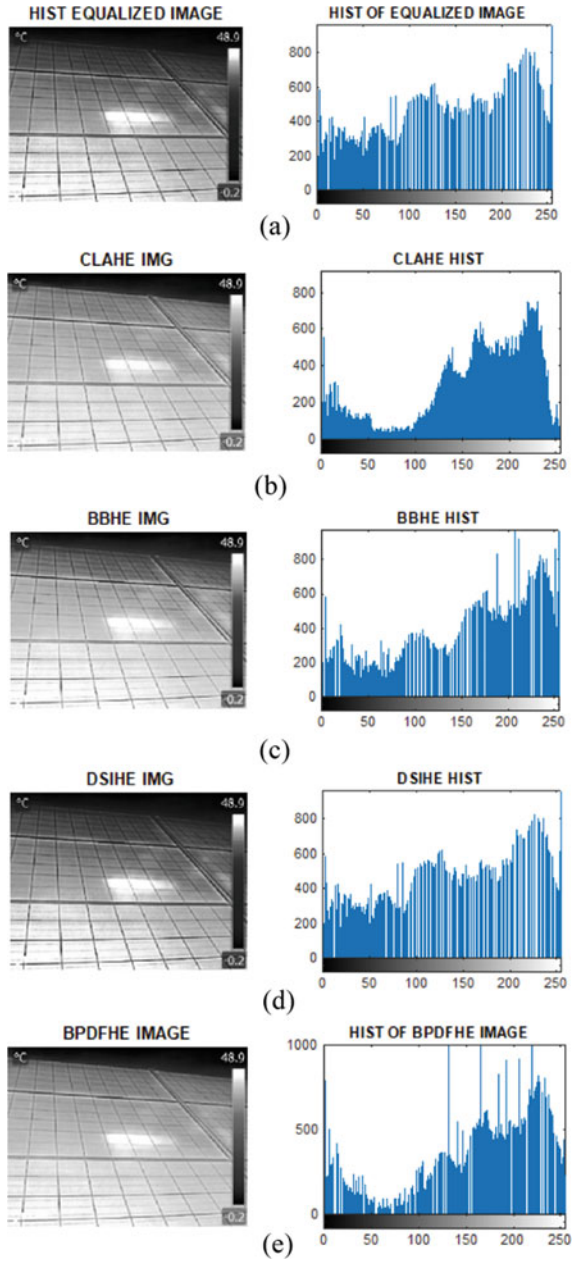


Table 5 Comparison of PSNR values

S. No.	Images	HE	CLAHE	BBHE	DSIHE	BPDFHE
1	IMG1	21.211	30.636	27.019	21.925	43.865
2	IMG2	23.235	30	23.61	23.854	37.153
3	IMG3	19.669	33.256	26.26	19.409	37.616
4	IMG4	24.634	33.712	23.231	24.653	35.716
5	IMG5	21.253	30.136	22.6226	20.06	44.655
6	IMG6	25.28	30.264	25.3	26.18	31.74
7	IMG7	20.5	30.913	20.97	20.58	42.322
8	IMG8	21.993	32.496	22.05	21.99	35.333
9	IMG9	15.552	33.01	25.651	16.387	38.939
10	IMG10	19.652	33.543	29.161	19.772	37.081
	Mean	21.2979	31.7966	24.58746	21.481	38.442



Fig. 9 Graphical representation of PSNR values for 10 sample images

Table 6 Comparison of AMBE values

S. No.	Images	HE	CLAHE	BBHE	DSIHE	BPDFHE
1	IMG1	17.661	1.183	2.4873	15.6708	0.017
2	IMG2	2.708	0.123	0.841	11.1299	0.0029
3	IMG3	17.68	0.0934	6.405	19.03	0.073
4	IMG4	0.6037	0.444	13.371	1.643	0.1045
5	IMG5	12.123	2.257	5.894	19.616	0.01755
6	IMG6	2.114	0.4723	3.999	4.318	0.0033
7	IMG7	4.406	3.281	4.13	6.411	0.02
8	IMG8	1.601	0.787	4.186	5.405	0.027
9	IMG9	33.94	0.124	1.145	28.78	0.0073
10	IMG10	21.261	0.293	4.6751	20.736	0.04136
	Mean	11.4098	0.9057	4.7133	13.2739	0.0313

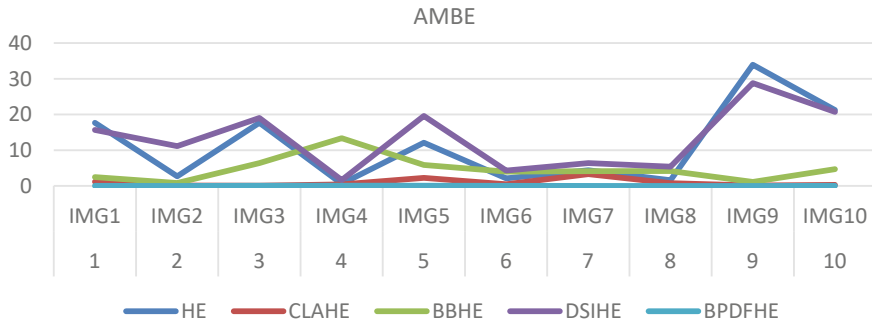


Fig. 10 Graphical representation of AMBE values for 10 sample images

Table 7 Comparison of SSIM values

S. No.	Images	HE	CLAHE	BBHE	DSIHE	BPDFHE
1	IMG1	0.906	0.98	0.9577	0.924	0.994
2	IMG2	0.9355	0.9817	0.9464	0.97	0.992
3	IMG3	0.885	0.9766	0.9	0.913	0.982
4	IMG4	0.9296	0.982	0.919	0.9422	0.9842
5	IMG5	0.9205	0.9813	0.929	0.945	0.994
6	IMG6	0.9579	0.9814	0.965	0.9629	0.9859
7	IMG7	0.86	0.981	0.8947	0.869	0.992
8	IMG8	0.8938	0.983	0.914	0.916	0.9852
9	IMG9	0.836	0.981	0.942	0.855	0.988
10	IMG10	0.899	0.985	0.9811	0.923	0.9849
	Mean	0.90233	0.9813	0.93489	0.92201	0.98822

indicates two images are very similar or same. From the calculated SSIM values, it can be stated that for the given sample images, BPDFHE gives maximum value indicating that original and histogram equalized images are more similar.

7 Conclusion

Noise removal techniques are very essential in any image processing applications. Several filtering techniques have been analyzed and its performance is assessed for solar panel thermal images using parameters such as SNR, PSNR, MSE, and SSIM. From the obtained results, it can be noted that bilateral filter gives excellent results for noise removal with highest values of SSIM and lowest value for MSE for the solar panel thermal images. Once the filter is finalized, next step will be histogram equalization of images to increase the intensity variation. Analysis of five

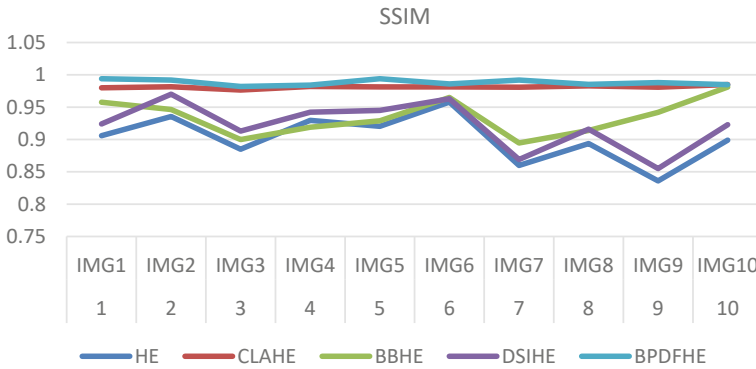


Fig. 11 Graphical representation of SSIM values for 10 sample images

different histogram equalization techniques such as HE, CLAHE, BBHE, DSIHE, and BPDFHE is included. Qualitative analysis is done using parameters such as PSNR, SSIM, and AMMBE. Experimental results indicate that BPDFHE is the most efficient histogram equalization technique for the given data providing lowest value of AMBE and highest value of SSIM.

Acknowledgements Authors would like to thank PV Diagnostics Ltd. Mumbai for providing thermal images of solar panel.

References

1. Prasad V, Gopal R (2016) LHM filter for removal salt and pepper with random noise in images. *Int J Comput Appl* 139:9–15. <https://doi.org/10.5120/ijca2016908962>
2. Yadav RB, Srivastava S, Srivastava R (2017) Identification and removal of different noise patterns by measuring SNR value in magnetic resonance images. In: 2016 9th international conference on contemporary computing, IC3 2016, pp 9–13. <https://doi.org/10.1109/IC3.2016.7880212>
3. Tania S, Rowaida R (2016) A comparative study of various image filtering techniques for removing various noisy pixels in aerial image. *Int J Signal Process Image Process Pattern Recognit* 9:113–124. <https://doi.org/10.14257/ijcip.2016.9.3.10>
4. Khetkeeree S, Thanakitvivil P (2020) Hybrid filtering for image sharpening and smoothing simultaneously. In: ITC-CSCC 2020—35th international technical conference on circuits/systems, computers and communications, pp 367–371
5. Isa IS, Sulaiman SN, Mustapha M, Darus S (2015) Evaluating denoising performances of fundamental filters for T2-weighted MRI images. *Procedia Comput Sci* 60:760–768. <https://doi.org/10.1016/j.procs.2015.08.231>
6. Hoshyar AN, Al-Jumaily A, Hoshyar AN (2014) Comparing the performance of various filters on skin cancer images. *Procedia Comput Sci* 42:32–37. <https://doi.org/10.1016/j.procs.2014.11.030>
7. Srivastava C et al (2013) Performance comparison of various filters and wavelet transform for image de-noising. *IOSR J Comput Eng* 10:55–63. <https://doi.org/10.9790/0661-01015563>

8. Janaki K, Madheswaran M (n.d.) Performance analysis of different filters with various noises in preprocessing of images. *Int J Adv Netw Appl* 372–376
9. Kumar MP, Murthy PHST, Kumar PR (2011) Performance evaluation of different image filtering algorithms using image quality assessment. *Int J Comput Appl* 18:20–22. <https://doi.org/10.5120/2289-2972>
10. Dwivedy P, Potnis A, Soofi S, Giri P (2018) Performance comparison of various filters for removing different image noises. In: International conference on recent innovations in signal processing and embedded systems, RISE 2017, Jan 2018, pp 181–186. <https://doi.org/10.1109/RISE.2017.8378150>
11. Varghese J (2013) Literature survey on image filtering techniques. *Int J Comput Appl Technol Res* 2:286–288. <https://doi.org/10.7753/ijcatr0203.1014>
12. Wahab AA, Salim MIM, Yunus J, Ramlee MH (2018) Comparative evaluation of medical thermal image enhancement techniques for breast cancer detection. *J Eng Technol Sci* 50:40–52
13. Garg S, Vijay R, Urooj S (2019) Statistical approach to compare image denoising techniques in medical MR images. *Procedia Comput Sci* 152:367–374. <https://doi.org/10.1016/j.procs.2019.05.004>
14. Paudel S, Rijal R (2015) Performance analysis of spatial and transform filters for efficient image noise reduction
15. Tomasi C, Manduchi R (1998) Bilateral filtering for gray and color images. In: IEEE international conference on computer vision. <https://doi.org/10.1677/joe.0.0930177>
16. Umamaheswari D, Karthikeyan E (2019) Comparative analysis of various filtering techniques in image processing. *Int J Sci Technol Res* 8:109–114
17. Zeng M, Li Y, Meng Q, Yang T, Liu J (2012) Improving histogram-based image contrast enhancement using gray-level information histogram with application to X-ray images. *Optik (Stuttg)* 123:511–520. <https://doi.org/10.1016/j.ijleo.2011.05.017>
18. Akila K, Jayashree LS, Vasuki A (2015) Mammographic image enhancement using indirect contrast enhancement techniques—a comparative study. *Procedia Comput Sci* 47:255–261. <https://doi.org/10.1016/j.procs.2015.03.205>
19. Cheng HD, Shi XJ (2004) A simple and effective histogram equalization approach to image enhancement. *Digit Signal Process* 14:158–170. <https://doi.org/10.1016/j.dsp.2003.07.002>
20. Lu L, Zhou Y, Panetta K, Agaian S (2010) Comparative study of histogram equalization algorithms for image enhancement. In: Mobile multimedia/image processing, security, and applications 2010, vol 7708, pp 770811–770811-11. <https://doi.org/10.1117/12.853502>
21. Suryavamsi RV, Reddy LST, Saladi S, Karuna Y (2018) Comparative analysis of various enhancement methods for astrocytoma MRI images. In: Proceedings of the 2018 IEEE international conference on communication and signal processing ICCSP 2018, vol 1, pp 812–816. <https://doi.org/10.1109/ICCSP.2018.8524441>
22. Senthilkumaran N, Thimmiraja J (2014) Histogram equalization for image enhancement using MRI brain images. In: Proceedings of 2014 world congress on computing and communication technologies WCCCT 2014, pp 80–83. <https://doi.org/10.1109/WCCCT.2014.45>
23. Pizer SM, Johnston RE, Erickson JP, Yankaskas BC, Muller KE (1990) Contrast-limited adaptive histogram equalization: speed and effectiveness. In: Proceedings of the first conference on visualization in biomedical computing, pp 337–345. <https://doi.org/10.1109/vbc.1990.109340>
24. Gupta S, Gupta R, Singla C (2017) Analysis of image enhancement techniques for astrocytoma MRI images. *Int J Inf Technol* 9:311–319. <https://doi.org/10.1007/s41870-017-0033-8>
25. Raj D, Mamoria P (2016) Comparative analysis of contrast enhancement techniques on different images. In: Proceedings of 2015 international conference on green computing and internet of things, ICGCIoT 2015, pp 27–31. <https://doi.org/10.1109/ICGCIoT.2015.7380422>
26. Kim YT (1997) Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Trans Consum Electron* 43:1–8. <https://doi.org/10.1109/30.580378>
27. Wang Y, Chen Q, Zhang B (1999) Image enhancement based on equal area dualistic sub-image and non-parametric modified histogram equalization method. *IEEE Trans Consum Electron* 45:68–75. <https://doi.org/10.1109/30.754419>

28. Sheet D, Garud H, Suveer A, Mahadevappa M, Chatterjee J (2010) Brightness preserving dynamic fuzzy histogram equalization. *IEEE Trans Consum Electron* 56:2475–2480. <https://doi.org/10.1109/TCE.2010.5681130>
29. Garud H, Sheet D, Suveer A, Krishna Karri P, Ray AK, Mahadevappa M, Chatterjee J (2011) Brightness preserving contrast enhancement in digital pathology. In: *ICIIP 2011—proceedings of 2011 international conference on image information processing*. <https://doi.org/10.1109/ICIIP.2011.6108964>
30. Bovik A, Wang Z, Sheikh H (2005) Structural similarity based image quality assessment, pp 225–241. <https://doi.org/10.1201/9781420027822.ch7>

Comparative Analysis of Medical Imaging Techniques Used for the Detection of Thyroid Gland with an Emphasis on Thermogram



G. Drakshaveni and Prasad Naik Hamsavath

1 Introduction

The secretory organ (thyroid) endocrine gland endocrine ductless gland may be a gland of the system. Set within the front of the neck between the collar Bones and below Adam's apple (the larynx). Endocrine glands are glands of the system that secrete their merchandise, hormones, directly into the blood rather through a duct.

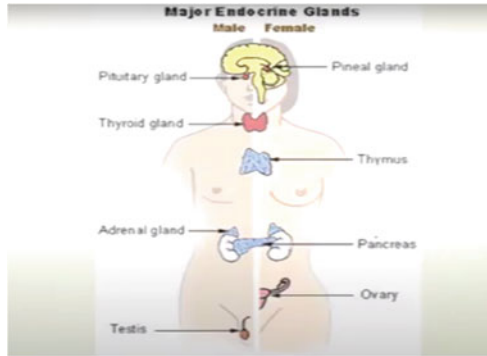


The major glands of the system embody the ductless gland, neural structure, endocrine, endocrine, pancreas, testis, the ductless gland, ovaries, and adrenal glands.

G. Drakshaveni (✉)
Department of MCA, BMSIT and Management, Bengaluru, India
e-mail: gdrakshaveni23@gmail.com

P. N. Hamsavath
Department of MCA and Advisor—Foreign Students, NMIT, Bengaluru, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_60

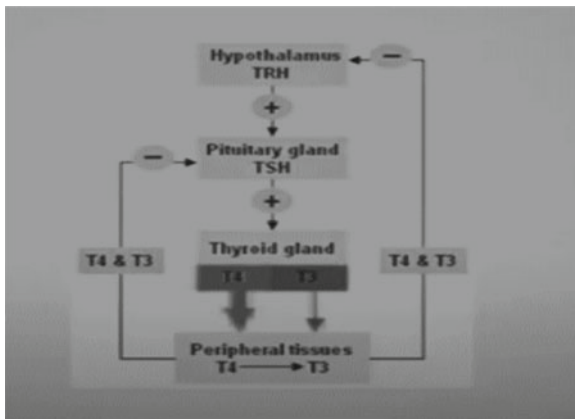


The thyroid gland produces hormones T4 and T3 which affect different organs and functions of human body, sleep, metabolic rate, bone, hair, brain, heart, skin and nails, intestines, muscles, temperature regulation.

Thyroid gland disorders are hyperthyroidism (high levels of thyroid hormone), hypothyroidism (low levels of thyroid hormone), thyroid nodules (growth/enlargement of the thyroid—goiter), thyroid cancer.

Test done for thyroid gland are: Blood test is done to identify thyroid stimulating hormone (TSH), Free T3, Free T4, Total T4 protein-bound, and Total T4 protein-bound, antibody tests, thyroid peroxidase, thyroid-stimulating immunoglobulin, thyroglobulin, ultrasound of thyroid, radioactive scan (the activity of thyroid gland), also for treatment hypothyroidism, and also a thyroid cancer.

Procedure for hormones generation in the human body: Hypothalamic–pituitary–thyroid (HPT)-axis gland sends positive signals to pituitary gland to produce thyroid stimulation hormones and pituitary gland, in turn, send positive signals to the thyroid gland to produce T4 and T3 hormones in to directly into bloodstreams. In turn, peripheral tissues send negative signals to the hypothalamic-pituitary gland to stop more hormone generations.



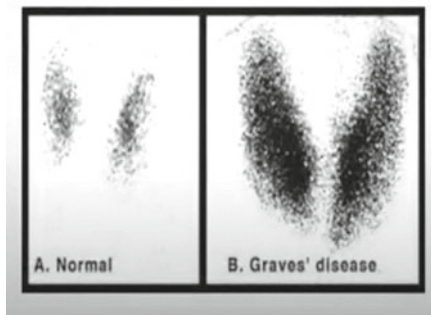
Hyperthyroidism: Hyperthyroidism is a condition where the body exposed to excessive thyroid hormones, it is 5–10 times more common in women than men. If it is the mildest form, then there are no symptoms. More often patients have discomforting, disabling, or even life-threatening symptoms, few more symptoms are trembling hands, fast heart rate, anxiety, irritable and quarrelsome feeling, weight loss, nervousness, even when eating more, increased sweating, intolerance to warm temperature, loose and frequent bowel movement, muscle weakness especially upper arms and thighs, thin and delicate skin, irregular heart rhythm, change in menstrual pattern, and prominent store of eyes in old patients. Accelerated loss of calcium from the bones, the risk for fractures.

Blood tests in hyperthyroidism: Blood tests in hyperthyroidism are done to test TSH level-thyroid stimulating hormone is the screen test usually LOW. If any abnormal in TSH, then it is followed by other tests like Free T4 and Free T3, Total T4, Total T3. Usually HIGH. Thyroid antibodies like TSI and TPO are also done.

Blood tests in hyperthyroidism:

Blood tests	Normal range	Hyperthyroidism
TSH	0.4–4.5	Less than 0.4
Free T4	0.6–1.8 ng/dl	Greater than 1.8
Free T3	1.4–4.2 ng/dl	Greater than 4.2

Radioactive iodine scan hyperthyroidism are done with I24 or iodine I31, the process to do this scan is normally after I24 or I31 injection; after 5 h, a scan is done to check the thyroid gland.



Treatment of hyperthyroidism: Medications, RAI treatments, surgical removal of the thyroid.

Hyperthyroidism on pregnancy: Two pregnant women out of 10,000 pregnancies are affected by hyperthyroidism. Usually, hyperthyroidism pregnant women will be having worse in the 1st trimester and improve in the 2nd and 3rd trimesters.

Usually, pregnant women are diagnosed with hyperthyroidism by a blood test. Pregnant women cannot use RAI scanning during pregnancy. Pregnant patient's lab work is kept slightly in the upper limit of normal (due to reason if controlled too well can cause hypothyroidism in the baby).

Hypothyroidism: Hypothyroidism is a condition where under activity of the thyroid gland happens when it produces less than the normal amount of thyroid hormones. Hypothyroidism slows down body functions. Hypothyroidism is most of the time permanent but sometimes it can be temporary. Studies have shown 10% of women and 3% of men are hypothyroid. Initially, the gland can secrete more hormones to compensate but is not able to keep up.

Hypothyroidism symptoms: Hypothyroidism symptoms are putty face dry itchy skin, constipation, drowsiness, dry and brittle hair, forgetfulness, difficulty with learning (with respect to kids), sore muscles, increased frequency of miscarriages, heavy and/or irregular menstrual periods, and increased sensitivity to many medications.

Causes for hypothyroidism: Congenital hypothyroidism is one type of hypothyroidism in infant born with inadequate thyroid tissue or enzyme defect. If not treated adequately can lead to physical stunting and mental damage, one in every 3000–4000 babies can have a diagnosis of hypothyroidism.



Diagnosis of hypothyroidism: Blood tests are done to test hypothyroidism. Usually, TSH is the screening test done and it will be usually high and also Free T4 and free T3 tests are usually low. No scans are requested [1].

Treatment of hypothyroidism: Treated with levothyroxine (Generic), synthroid, and levoxyl (Brand Name). Usually based on weight. Some cases like the elderly or patients with cardiac issues can start at a lower dose. In most cases, treatment is needed for life. Periodic monitoring of TSH is needed for optimal treatment.

Hypothyroidism and pregnancy: Hypothyroidism condition is not very clear if all women planning for pregnancy need to be tested for thyroid disease. High-risk patients or patients with hypothyroid symptoms with family history need to be tested. If women are already hypothyroid needs to be tested as soon she knows she is pregnant and every 4 weeks in the 1st half of pregnancy and every 5–6 weeks thereafter, medication needs to adjusted based on the tests, if not treated during pregnancy, it

may lead to anemia, premature birth, and miscarriage can occur in pregnant women [2].

Thyroid nodules: Thyroid nodules occur in 5% of women and 1% of men and increase in frequency with age. When a nodule is found, cancer needs to be ruled out, 95–97% are not cancerous, usually, no symptoms found on routine physical exam, sometimes on ultrasound, MRI or CT of the spine or chest or a PET scan.

Thyroid nodules symptoms: Symptoms for thyroid nodules are most of the time no symptoms, sometimes difficulty in swallowing, difficulty in breathing, sometimes can cause harassment of voice if pressing on the nerve which supplies the voice box, even it can affect cosmetic appearance.

Thyroid nodule treatment: No medical treatment for thyroid nodules, if causing compression symptoms, then surgery is the only option.

Thyroid cancer: The chance of being diagnosed with thyroid cancer has increased three-fold in the last 30 years, due to the use of thyroid ultrasound, detection of smaller cancers at a younger age than most adult cancers. Three out of four cases are women. Women in the 40s and 50s are more proven to thyroid cancer, men in the 60s and 70s chance is more, death rate is steady for many years. It can be hereditary in the cause of medullary thyroid cancer. Having a first degree relative like a parent, sister, or child is with thyroid cancer increases the risk and low diet in iodine can increase the risk of follicular and papillary thyroid cancer. Radiation is proven to have a risk factor. Head and neck radiation in childhood.

Thyroid cancer signs and symptoms: A lump within the neck, generally growing quickly, swelling within the neck pain within the front of the neck, typically mounting to the ears and roughness or different voice changes that do not escape, hassle swallowing, and hassle respiration. A continuing cough that's undue to a chilly.

Thyroid cancer diagnosis and types: Thyroid cancer is diagnosed by ultrasound and biopsy.

Types of thyroid cancer: Thyroid cancer types are papillary thyroid cases which are most common, follicular thyroid cancer 2nd most common, hustle cell cancer, medullary thyroid cancer, anaplastic thyroid cancer poor outcomes.

RAI scan and treatment: Uses after surgery to diagnose any residual disease and also if need after the treatment PET scans are used if cancer does not take up iodine to find cancer has spread [3].

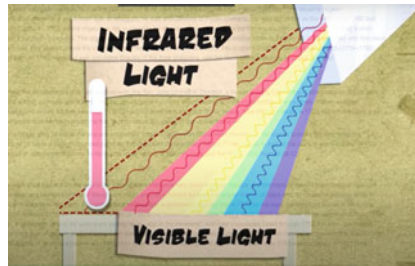
Thyroid cancer treatment: Treatments for thyroid cancer are surgery which is the main treatment for any type of cancer. Radioactive iodine can be used if patients are in the intermediate high risk, preparation for RAI diagnosis on treatment can be done, slipping the thyroid medications, by using a medication causes thyroid. All patients need to be on thyroid replacement, and depending on the risk of the patient, the dose adjusted based on TSH and also neck ultrasound is done periodically to see if there is the only recurrence [4].

2 Thermogram Images

Due to COVID-19 screening, each person to check skin temperature before they enter a building. It is an easy and efficient way of helping to reduce the risk of spreading the virus. FLIR is the best thermal imaging camera.

Procedure for the Thermal Imaging Camera

The first person who invented infrared light was Sir Frederick William Herschel, who passed sunlight through a prism and found out that seven colors were out of the prism [5].



Then, he measured the temperature of all the colors and found that low temperature was ultraviolet color, as he moved toward another color he found that there was an increase in the temperature and the highest temperature was with the color red and lowest temperature was with blue or violet color and even he checked beyond red color which was not visible light then he found that there was a temperature which was higher than red but not visible he was the first person who invented infrared region with a wavelength of 1050 which is an invisible region (invisible) to the human eye.



Infrared thermography is a technique addressed to the visualization and acquisition of thermal images [6].

Different process of images taking different way length.

Later, Max Planck gave the mathematical equation $E = h\nu$, where h is a Planck constant and ν is the frequency of the radiation and E is energy.

Wien's law formula



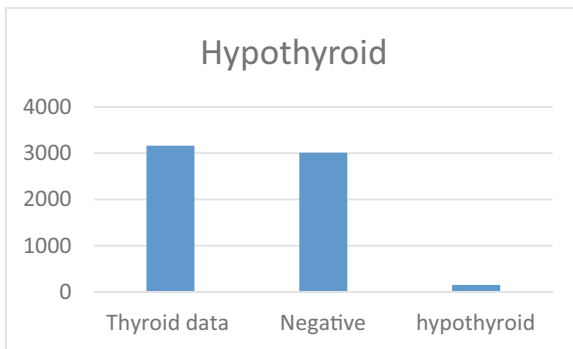
The equation describing Wien's law is very simple: $\lambda_{max} = b/T$, λ_{max} is the aforementioned peak wavelength of light. T is the absolute temperature of a black body. $b = 2.8977719 \text{ mm} \cdot K$ is the Wien's displacement constant [7].

Any camera will be having three components, they are main lens (Glass), sensor, and electronic processing unit, whereas in thermography, camera lens will be Germanium lens (0.7 eV), sensors will be indium, gallium and arsenic, and image processing.

Types of thermal imaging cameras: They are un-cooled and cooled thermal imaging camera.

3 Result

Out of 3126 dataset, the following are results

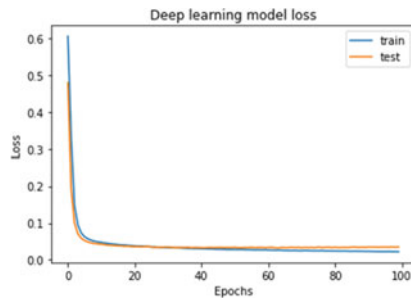


summarize the result and plot the training and test loss

```

plt.plot(result.history['loss'])
plt.plot(result.history['val_loss'])
# Set the parameters
plt.title('Deep learning model loss')
plt.ylabel('Loss')
plt.xlabel('Epochs')
plt.legend(['train', 'test'], loc='upper right')
# Display the plots
plt.show()

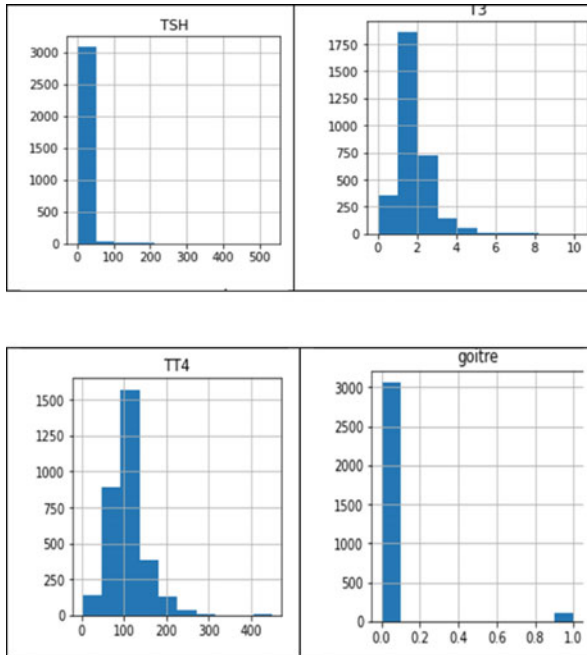
```



```

# Input
model = Sequential()
# Hidden layer
model.add(Dense(64, kernel_initializer='uniform', input_dim=24,
activation='relu'))
# Output layer
model.add(Dense(1, kernel_initializer='uniform', activation='sigmoid'))

```



4 Conclusion

Thyroid gland is very important gland for human being to be normal, thyroid gland secretion of hormones is increased then human will be hypothyroid patient. If thyroid gland secretion of hormones is less, human will be hyperthyroid patient. So, it is very important for any human being to health conscious. Designing better image enhancement technique will aid detecting and segmenting thyroid more efficiently which in the future will be considered.

References

1. Shi W, Zhuang X, Wang H, Duckett S, Luong DV, Tobon-Gomez C, Tung K, Edwards PJ, Rhode KS, Razavi RS et al (2012) A comprehensive cardiac motion estimation framework using both untagged and 3-D tagged MR images based on nonrigid registration. *IEEE Trans Med Imaging* 31(6):1263–1275
2. Fa Y, Mendis S (2014) Global status report on non communicable diseases 2014. World Health Organization reports
3. Sathish D, Kamath S, Rajagopal KV, Prasad K (2016) Medical imaging techniques and computer-aided diagnostic approaches for the detection of breast cancer with an emphasis on thermography—a review. *Int J Med Eng Inform*

4. Lustig M, Donoho DL, Santos JM, Pauly JM (2008) Compressed sensing MRI. *IEEE Signal Process Mag* 25:72–82
5. Liang D, DiBella EV, Chen RR, Ying L (2012) k-t ISD: dynamic cardiac MR imaging using compressed sensing with iterative support detection. *Magn Reson Med* 68:41–53
6. Axel L, Montillo A, Kim D (2005) Tagged magnetic resonance imaging of the heart: a survey. *Med Image Anal* 9(4):376–393; Lustig M, Donoho DL, Pauly JM (2007) Sparse MRI: the application of compressed sensing for rapid MR imaging. *Magn Reson Med* 58:1182–1195
7. www.healthand.com
8. Zhao B, Haldar JP, Christodoulou AG, Liang ZP (2012) Image reconstruction from highly undersampled (k, t)-space data with joint partial separability and sparsity constraints. *IEEE Trans Med Imaging* 31:1809–1820
9. www.patientmemoirs.com
10. www.allbreed.net
11. www.data.conferecnewworld.com
12. www.adclinic.com
13. www.cancer.org
14. www.endocrine.org
15. www.hopetbi.com
16. <https://en.wikipedia.org/wiki/Infrared>
17. https://en.wikipedia.org/wiki/Max_Planck
18. www.omnicalculator.com/physics/wiens-law
19. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
20. Ramos-Llordén G, den Dekker AJ, Sijbers J (2017) Partial discreteness: a novel prior for magnetic resonance image reconstruction. *IEEE Trans Med Imaging* 36(5):1041–1053
21. Salem N, Malik H, Shams A (2019) Medical image enhancement based on histogram algorithms. *Procedia Comput Sci* 163:300–311. In: 16th international learning & technology conference

The AgroCart Android Application to Manage Agriculture System



N. Sreenivasa, B. A. Mohan, Roshan Fernandies, H. Sarojadevi, E. G. Satish, and Abrar Ahmed

1 Introduction

India is a country, where more population is passionate about the agriculture business, directly or indirectly. The manufacture of food and raw materials in agriculture eventually is the reason for the existence of the population. The demand for food production technology is continuously increasing. In India where the population is very passionate about agriculture but still we are not able to alter the utilization of agriculture. However, there is a need to review the mechanism for improving the technology. The dearth of rain and overproduction of crops in the market is one of the foremost reasons, an extension of irrigation facilities, water management, and use of seeds. The majority of Indian farmers, including small-scale producers, are often unable to access the information that could increase the yield of the crops and

N. Sreenivasa (✉) · H. Sarojadevi · E. G. Satish · A. Ahmed
Department of CSE, Nitte Meenakshi Institute of Technology, Yelahanka, Bangalore 560064,
India

e-mail: sreenivasa.n@nmit.ac.in

H. Sarojadevi

e-mail: sarojadevi.n@nmit.ac.in

E. G. Satish

e-mail: satish.eg@nmit.ac.in

A. Ahmed

e-mail: 1nt18cs400.abrar@nmit.ac.in

B. A. Mohan

Department of ISE, BMSIT, Avlahalli, Yelahanka, Bangalore 560064, India

e-mail: mohan.ba@gmail.com

R. Fernandies

NMAM Institute of Technology, Nitte, Mangalore, India

e-mail: roshan_nmamit@nitte.edu.in

lead to a better price for their crops. Marketing and selling the crops at a proper price with average profit is that the most important problem faced by the farmer is lack of pricing. The widespread network of mobile phones could be the game-changer in this problem. The main purpose is to develop smart mobile phone-based solutions that help in agriculture farm management, which leads to agricultural revenue improvements, and also helps in the conservation of the agricultural farms. The Agro Cart is an android-based application that is developed to maximize profits for the farmers. This application is the bridge between the farmer and the customer that helps buyers and growers in such a method that none of them must compromise.

1.1 Brief History of Technology/Concept

This Agro Cart has Four Modules:

(i) Preprocessing

As we have discussed before overproduction of crops in the market, is one of the major problems for the farmers for not gaining profit. So, we have planned in such a way there will be no loss for the farmers if they follow this application. Preprocessing module guides the user to grow the crops according to the market requirements. According to the market survey, we guide farmers that this much is the requirement for next month or according to the crop cultivation period. Every crop has a certain period of time till harvesting. The farmer has to register as soon as the area is allowed to register. This module works on the basis of first come first serve. For an ex: Tomatoes can be classified, when they become to harvest: Early season: need 45–60 days to reach harvesting from transplanting. Midseason: need 65–80 days to touch harvesting from transplanting. Late season: need 85 or more days to reach harvesting from transplanting. So, according to the market requirements allowed area is provided in the application for the next period to cultivate the tomato. Then the farmer has to register and book the land according to his investment. If the allowed area is full for a certain period then he has to wait for next month to cultivate. User can see how much area is remaining and how much area is registered.

(ii) Buying and Selling

After Growing the crops farmers will not get proper prices for their crops. So, to overcome this problem we are helping farmers to gain profit. In this module, there will be two phases one is for the farmer and another is for the customer. The farmer needs to register and login into the application and there will be lists of crops with the price he has to check for his product and update the number of crops he wants to sell. The price will be based on the market price. By this farmers will get a good price for their crops. Another phase is for a customer where he has to register and login into the application. The list of crops with price will be displayed at what price the farmer wants to sell. The customer needs to add a crop to the cart and then check

out his product and have to pay according to his needs. This application is like a bridge between the farmer and the customer.

(iii) **Blogs**

Most of the farmers are unemployed they are totally dependent on the shops for fertilizer. To overcome this Agro Cart guides farmers by providing proper information to use the best fertilizers to get a better yield of crops. In the blogs module, farmers can get the proper information from experts and there will be a YouTube link so that farmers can see and cultivate in a proper method. Farmers can also post their questions regarding the fertilizer that has to be sprayed for the crops. An expert and Agriculture doctors are connected with the application to help farmers by providing proper dose for the crops.

(iv) **IoT-Based Smart Irrigation**

Irrigation is one the most important for the successful cultivation of a crop. The IoT is remodeling agriculture with top techniques. IoT technology helps in collecting information about a condition like moisture temperature and soil humidity, etc. So In Agro Cart, IoT is used to check the humidity, soil, and moisture before cultivating. Farmers can make use of this for smart irrigation like he can monitor the water for the crops by this application. And it also has pest detection. There will be sensors fixed in the land if some pest or animals are entered in the cultivated land then the alarm will be buzzed and a notification will be sent to the owner's number.

1.2 Applications

- Agro Cart is a mobile app that helps farmers to grow crops according to the market requirement.
- With help of Agro Cart, farmers will get a good price for their crops. There will be no wastage of crops or overproduction of crops.
- Agro Cart is a bridge between the customer and farmers.
- Agro Cart helps farmers to plan sprays of their crops efficiently with the help of weather conditions.
- Agro cart is a mobile app that detects and identifies disease simply by taking photos and uploading in an application.
- Agro cart helps farmer to detect and guard their crops from pest and animals.
- Agro Cart helps farmer to use fertilizers to get a better yield of crops.

2 Related Work

Online Shopping Portal is like a selected requirement of the customer that combines the shopping like buying and selling and promoting the offering in particular to their

customers. Reports may be generated at any time inside few seconds, in order that guide exertion is not required, and additional evaluation may be executed tons extra often which facilitates in taking decision. Allow a customer to get the registered form their location or area and transact for the specified product. It has always been India's maximum sizeable economic area. Farming is probably characterized as an included Association of strategies to manipulate the improvement and gathering of animals and fruit green or vegetables it as a recognizable stack Indian GDP as maximum sports are agro-based. Maharashtra is surely actively all rights considered.

A. An Android Application for Plant Leaf Disease Detection and Automated Irrigation

Due to climatic changes, the necessity for correct and bearable irrigation techniques is in excessive demand. The crops want to be irrigated in step with their water necessities primarily-based totally on climatic conditions. So, during this paper [1], a clever irrigation machine is proposed that could manage irrigation robotically the usage of a mobile utility. The snapshots of leaves are captured and are dispatched to the server, which's processed and as compared with the diseased within side the cloud database. Based on differentiation a listing of sickness suspects is dispatched to the consumer through cell utility. The machine includes (1). An embedded machine: The embedded machine has a microcontroller, a soil moisture sensor, a temperature sensor, a relay switch, and a Wi-Fi module. The consumer can log in to the cloud the usage of the consumer call and the password. A digital dig cam is supplied within side the utility for taking an image and sending it to the cloud. Then the pictograph is processed from the cloud and as compared with a database this is supplied. If the picture graph suits any of the pictures from the database, then the prediction of illnesses is dispatched to the consumer's utility (2). The android utility: After the verification of the consumer, the consumer has the option to manage and taking photos of leaves for studying the sickness. This manage statistics is then processed via way of means of the microcontroller and automated mode. Users can view statistics approximately the contemporary soil moisture sensor and temperature sensor reading. This machine is straightforward and cost-effective. The consumer can view and have interaction with the contemporary utility that's related to the cloud. The consumer also can come across sickness via way of means of taking photos of leaves and sending them to the cloud.

B. Crop-Shop: This Application is to Extremity Financial Gain for Farmer

In this paper [2], for a while, in India farmer have low amount of freedom in picking marketing and buyers for distribution of services. The nation in all States, aside from three, order includes advertising and exchanging marketing of home stead distributing should be coordinated of all-over state-claimed and regulated by market yards' showcases where go between intermediates squash, ranchers build edges, as per Goldman Sachs, middle people have become overwhelming purchasers of the agrarian market, coming about them to assume responsibility for the predicament of the ranchers and swallowing every one of the benefits. In the paper [3], the rancher's workday and late evening anticipating a proper yield. They utilize a lot of monetary

assets loaning cash and purchasing manures, seeds, and so on thus, they reserve the privilege to appreciate each rupee acquired of their corp. In this particular situation, we suggest a framework that brings ranchers close to the outlets reducing the marketers. The marketers commonly soak up to 70% of the blessings of ranchers leaving them vulnerable. Our framework contains a transportable utility so one can fill in as a level for the cultivators and outlets or customers to promote and buy their ranch gadgets. These framework objectives give a useful price to ranchers to their dwelling, House gadgets reducing the go-betweens. This lets in the outlets or the customers to buy gadgets from the ranchers at a decrease than the everyday price.

C. E-Commerce Application for Farmers

In this paper [4, 5], the digital market is the platform to integrate and bridge the gap between the farmer, markets, government, and users. It additionally lets everyone be refreshed with the changing business situation. Indian farmers are facing various challenges like not getting a good profit for their efforts as well as the investments they invest in farming. There are several reasons for this, such as a limited season, the shelf life of plants because there is not enough time to study market conditions. Studying the flowering plants and products on the current market in the agricultural sector is very important to get a good one. Since it makes no sense for the farmers to physically reach all traders since it takes a lot of time and effort, where our where farmers do not have that long time. In addition, the methods introduced by farmers traditionally create limited access to customers (traders), so there are very few options for selling the crop in the market, so with the new marketing method a farmer can sell his harvest at every level of the marketing chain (trader), Markets or directly to users along with several options. By selling their crops at a minimum price, they may not be able to fulfill their changing demands. The platform will help to sell the plants at different levels of the marketing chain, where an analysis of the current market situation can be carried out with the help of KNN algorithms and with GPS to buy or sell the plants. It provides a complaint box to launch complaints.

D. Crops Disease Finding and Anticipation by Android Applications

This technique was introduced by the students of JCEM in the Year 2015. In this paper [6, 7], it is said that according to their survey in some villages more than 80 percentile of farmers tries new kind of crops instead of their old traditional because of the lack of experiences or knowledge. This application proposed the following services for farmers:

- (1). Image Processing: In this module image are captured by camera and using image processing formulas detects leaf diseases
- (2). Online marketplace: This module facilitates third-party vendors by allowing them to sell agricultural products in one marketplace. Marketplace ecommerce will benefit for all in a number of ways Vendor: Smaller stores without the investment can establish their own ecommerce website. Consumer: Help customers from seeing advances option on this application in their mobile and find lower price and quality products
- (3). Market Rate Guide: In this module, users get information about market rates across all the available markets in geographically distributed areas. This is a web service which is

provided by government organizations to keep track of market rates. Every market has their own price as per monopoly situations (4). Weather report: In this module reports are based on a web service by www.openweathermap.org. This module help farmer to take decisions regarding water management, pesticides (5). Soil information: This module provides information about different soil types, by this user may get information about soil types. This helps farmers to decide which types of crops are suitable for this particular soil.

E. E-Commerce Approach Based on IoT and Blockchain Technology

The upsurge of the populace with a partner dramatic rate makes taking care of everybody a tremendous worry for the farming area. One little-investigated business that the trap of things can possibly reform absolutely is farming. From rancher to co-ordinations and customer, the Blockchain And the Internet of Things (IoT) and is modifying the food-producing industry. With the unfurling of IoT and blockchain, it is gotten essential to guarantee food handling for everyone brings certainty and straightforwardness inside the evolved way of life. In this paper [8], we are planning an IoT-based Farm robot and blockchain-based versatile application by contracting ecological impression, heightening customer happiness, improving straightforwardness all through the arrangement chain, and promising certifiable monetary profit to the rancher are key features. The harvest screen application makes the rancher in the event of negative conditions aware of make fitting moves. It permits the client to request for his things which will be told through the mail and afterward the co-ordinations group does the work of transportation which can be followed. The result of this undertaking is to guarantee the best quality harvest creation in view of continuous checking. The portable application is an immediate stage for online business among ranchers and clients without the association of mediators. Simple store network likewise guarantees a more attractive exchange. The rancher is furnished with deference, pay, and contribution in standard agronomics.

F. MahaFarm—An Application for Solution in Remunerative Agriculture

MahaFarm is an application which is developed by students of Information Technology Mumbai and KJS Institute of Engineering in the Year 2014. In this paper [9], they have stated that many applications are available related to agriculture in the US but no such application available in India. The researchers and interviewers are specified that farmers had an extensive variety of information needed. There are four modules in this application. General information: In this module Crops information is provided to the farmer like what is the need for growing a crop and what are precautions need to be planned before cultivations. Next is Weather Updated: Here weather conditions are shown in the application. What is the forecast for today and for the upcoming four days? So, the farmer can decide when to cultivate and spray for their crops. Third is market price: In this module Market price for a particular crop is shown in the application. So the farmers can get an idea and knowledge about the market price. And the fourth module is News Updates where agricultural news is shown in the application to motivate farmers. This application was based on SMS services like all the information which are required by the farmers is sent through

the SMS only. The drawback is that data and information are sent through SMS and there will be a delay in the SMS and have security issues.

G. Real-Time Atomization of Irrigation Systems for Social Transformation of Indian Agriculture System

The paper “Real-time atomization of the agricultural environment for social modernization of Indian agricultural system” using ARM7 and GSM’ is focused on an automatic irrigation system for the development of the Indian agricultural system and to give better irrigation in a particular area. This setup consists of an ARM7TDMI core, which is a 32-bit-microprocessor; GSM plays an important role, since it is responsible for the management and control of the irrigation in the field and sends this in the form of coded signals to the receiver. GSM operates through SMS, and it is the link between ARM processor and centralized unit. ARM7TDMI is an advanced version of microprocessors and acts as the heart of the system. The goal of the project is to realize basic applications in the field of automatic irrigation by programming the components and placing the necessary equipment. This project is used to find the field condition and use GSM for information sharing through SMS. So, in this paper [10, 11] the purpose of this project is to improve the irrigation system of the Indian agricultural system. Good environmental conditions are very important for better plant growth, higher yields, and rational use of water and other resources (such as fertilizers). This project was developed with mobile phones, based on agricultural electric pumps to remotely control irrigation, thereby reducing labor costs. The project has further applications in the field of family farming. It has a precise irrigation system through controlled irrigation.

H. A Survey of Automated GSM-Based Irrigation Systems

GSM-Based Irrigation System is proposed by students of the Engineering Institute of Engineering and Management Kolkata in the Year 2019. India has different types of soil in different areas. The Agriculturists depend on the monsoon and all different areas are not getting the same amount of rainwater in the country every year. If crops did not get water at the scheduled time, crops will not grow as expected. Since it happening every year, farmers are getting losses every year. There are many different types of solutions to solve this problem but in this paper [12], they proposed a solution with a less cost and sustainable solution. In this paper [13], they clearly mentioned, the proposed system will since the content of the soil moisture and its fertility. According to the moisture of the soil, the water will be supplied and everything will be done automatically. This saves time, wastage of water and reduces the labor cost.

3 System Requirements

(i) Software Requirements and Hardware Requirements

Cloud (parse): To Manage The backend.

Good Internet connection.

PC—Any PC/Laptop with a minimum of 500 MB hard disk, 4 GB ram, 1 GB graphic card, android phone (Mobile).

User Requirements:

The system should be user-friendly so that it is easy to use for users. The system should run 24 h a day. The system should refresh faster and take less time than possible to respond. Loading of UI must be faster. The user will search for different types of crops so the system must display the accurate result. They should handle a large amount of data. The system should handle unexpected errors.

(ii) Functional Requirements

1. The system should be user-friendly so that it is easy to use for users.
2. The system should run 24 h a day.
3. The system should refresh faster and take less time than possible to respond.
4. Loading of UI must be faster.
5. The user will search for different types of crops so the system must display the exact result. The system should handle a large amount of data.
6. The system should handle unexpected errors.

(iii) Non-functional Requirements

1. Reliability: System must able to support good relationships and trust.
2. Usability: System must be designed user-friendly so that everyone can use it. The system must able to display all information in the local language.
3. Efficiency: The system should be fast while responding to a user and able to respond to each and every user simultaneously.
4. Maintainability: The system should able to handle (adopt) or support change in the future.
5. Portability: The system should run on each and every mobile device which meets the need of the proposed system.

4 System Design

(i) Architectural Design

As we discussed proposed System has a three-module, Fig. 1 describes the data flow of the application.

The user needs to login using a login credential. The system will authenticate if the credential corrects it provides access to the application, if wrong then gives an error alert. A new user has to register first than they can log in. After login former

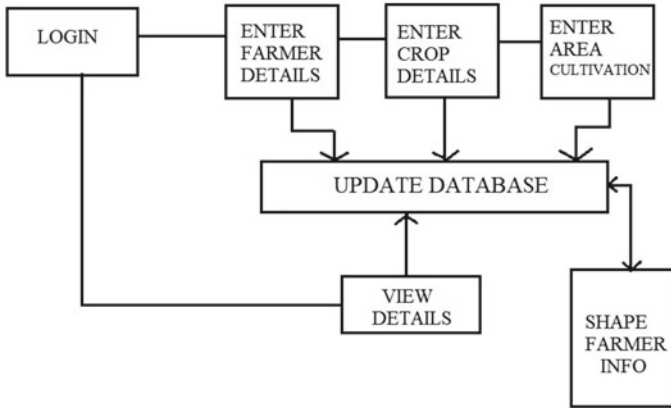


Fig. 1 Data flow diagram

are requested to enter their personal information, details of crops they cultivate, and total area of cultivation land. These all details will be stored in the system database. User can also see their information and other user name and contact numbers of who are registered for the cultivation of crops.

(ii) Use case Diagram

Pre-processing module data flow is described in Fig. 2 (Use Case Diagram), the System will be handled by the Admin and used by the user.

Users will be first requested to register and they are given access to log in. Admin has the option to update crop details, adding new crops manually according to the

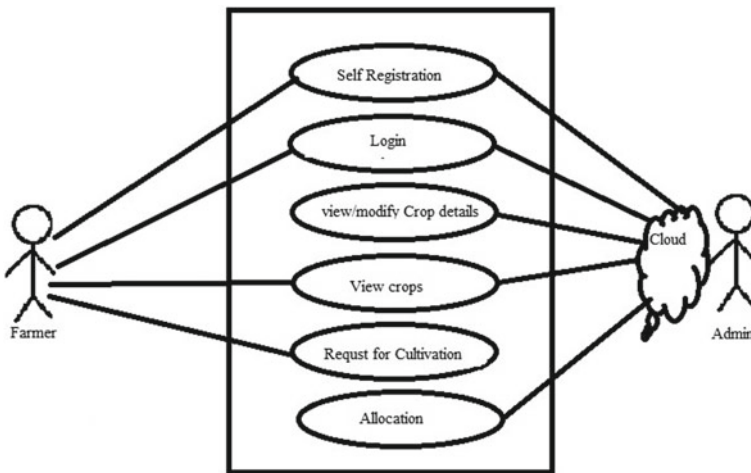


Fig. 2 Use case diagram

market needs. The user can view crop details, the details contain the name of crops, total crops needed in the market, available area, and registered area. Users are requested to register for crop cultivation. The user has to select a crop and needs to provide the total area that the user is going to cultivate the respective crops. All details will be updated in the database and the details also updated in the application. Here first come first serve will be followed if the registered area meets the marked needs for particular crop than user are requested to register for other crops available, so the user gets the minimum profit.

Buying and selling module working is described in Fig. 3 (use case diagram), Here the admin will update and add new products (crops) details like name, price, quantity, quality, and name of the seller.

Management of product is taken care of by the admin. Admin will have the option to delete and modification of the product. The user needs to log in to the application to view the product details. User has to select the products they want to buy. The selected products will be automatically stored in their cart. The cart can be managed by the user. Next user needs to provide an address and contact number. This information

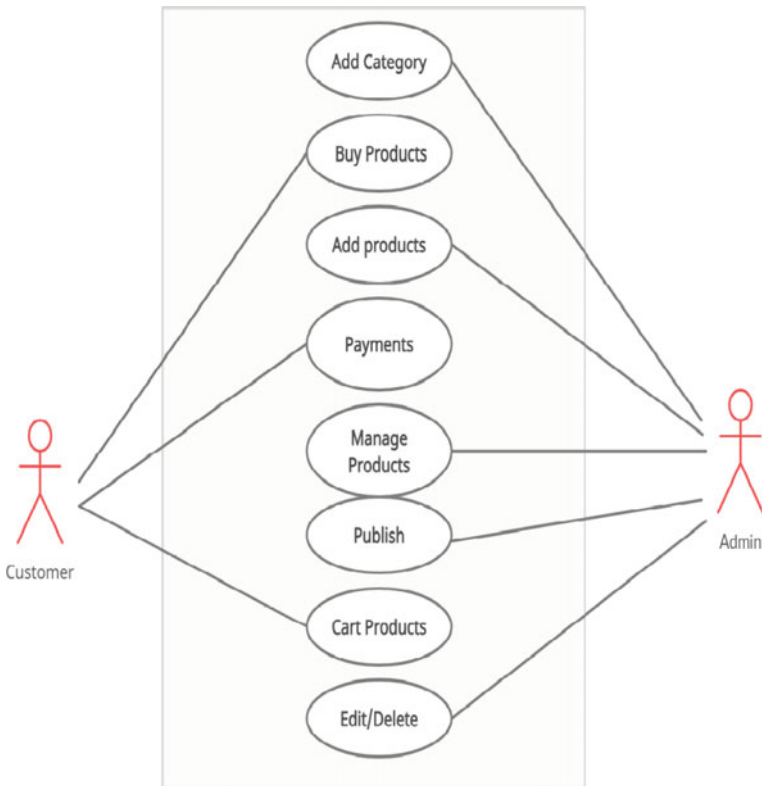


Fig. 3 Buying and selling use case diagram

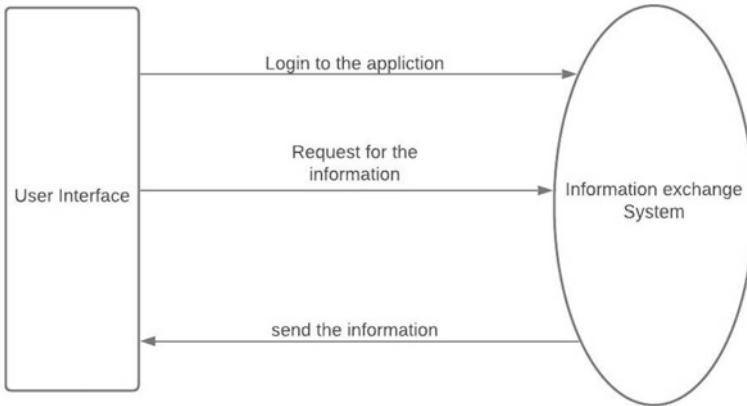


Fig. 4 Blog module

will be verified by the admin. Next users will be provided different payment options, the user needs to select one and must complete the payments. Payments done by the user will be verified and confirm the order by the admin.

Blog module is shown in Fig. 4, the user needs to log in first and user have the option to ask their query’s related to cultivation method, information about medicine any other guidelines.

These queries will be answered by the expert and if YouTube videos are available the system will provide them for the user.

5 Technical Requirments

HTML: may be a hypertext terminology that’s wont to create electronic documents on the World Wide Web. Every page that we see on the net contains HTML code that shows text and pictures in an exceedingly readable format.

Java: it is a programing language as a core technology that’s utilized in production on World Wide Web contents. It is wont to make pages interactive. Java could be a platform-independent that Write-once-run anywhere is one in every of the important key features of java language that produces the foremost powerful language. other features of java are Simple, Object-oriented, Robust, Portable, and Multithreaded.

Android: it is a software stack for mobile devices that features middleware and key applications. Some features of Andriod are Application Framework, Integrated Browser, Optimized Graphics, SQLite, Messaging, application, and Data Storage.

6 Results

- Agro Cart is a mobile app that helps farmers to grow crops according to the market requirement.
- With help of Agro Cart, farmers will get a good price for their crops. There will be no wastage of crops or overproduction of crops.
- Agro Cart helps farmers to sell their products at a good price. Farmers can check the market price and demand their products to sell at a good price.
- Agro Cart is a bridge between the customer and farmers.
- Agro Cart helps farmers to plan sprays of their crops efficiently with the help of weather conditions.
- Agro cart is a mobile app that detects and identifies disease simply by taking photos and uploading in an application.

The Stage where planning and development are done involves testing, operating, developing, and maintenance of software products. This model improves project communication and increases project manageability, cost control, and quality of products. Leading the models and business rules are exhausted in the analysis phase. Leading the software architecture is finished within the designing phase. Operations: Installation, migration, and maintenance of the total system. Sometimes this model is additionally called the Waterfall Model. The implementation phase of software development involves design specification into ASCII text file and debugging, and unit testing of the ASCII text file.

There is a need within the operations that provides the positions between the farmers, the user, and the trader where the customer username and passwords are stored in the system database resulting in the startup system session. A query that receives data and are stored in the cloud via web content to induce general data situations with the help of KNN algorithms and the use of GPS to buy or sell crops. Provide a complaint box to file complaints.

Step 1: Login Page: Users need to register before getting through the application. Complete the registration form by filling in all details like User Name, Email Id, and Phone Number is as shown in the Fig. 5.

Step 2: Home Page: After successful registration user can view the home page (Fig. 6) of the application. The home page consists of Register Crop, Market Rate, Best Practices, and Blogs.

Step 3: Register Crop: User can view allowed area and registered area for particular crop is shown in Fig. 7. User can register the crops according to allowed area.

Step 4: Buying and Selling (Fig. 8): Customers can view the agriculture products available in the application and then apply according to his requirement.

Step 4: Blogs: There are two options as shown in Fig. 9, first is the user can view the suggestions provided by the agriculture experts and the second is the user can write his views and queries so that others can make use of them.

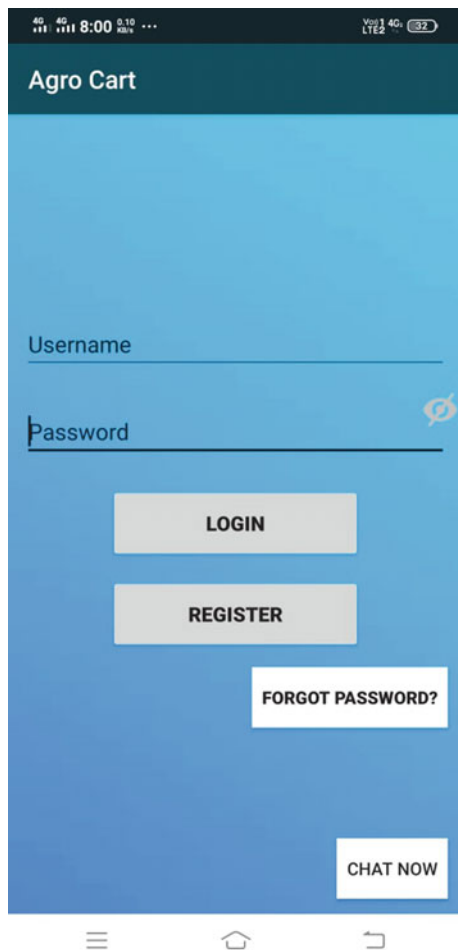


Fig. 5 Login page

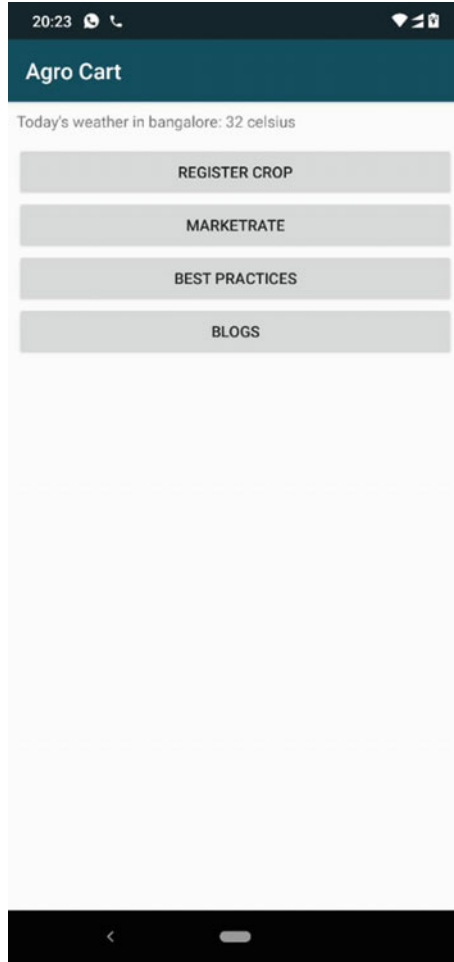


Fig. 6 Home

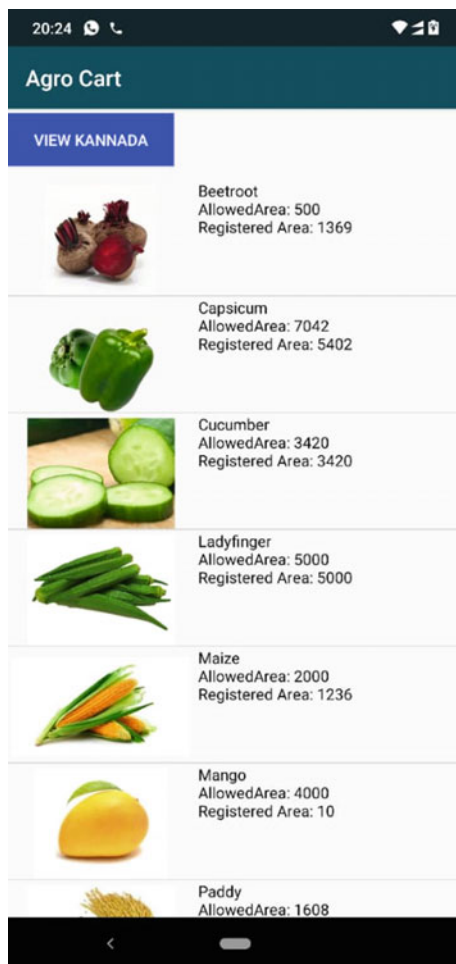


Fig. 7 Register crop

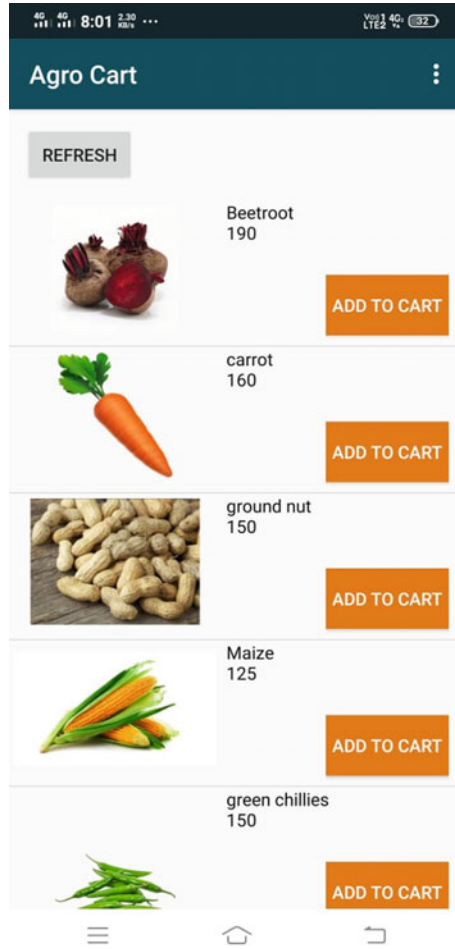


Fig. 8 Buying and selling

The screenshot shows the 'Agro Cart' application interface. At the top, there is a dark teal header with the text 'Agro Cart'. Below the header, the form consists of three input fields: 'Enter topic name', 'Enter youtube link', and 'Enter blogs content'. Each field is followed by a horizontal line. At the bottom of the form, there is a grey button labeled 'SUBMIT'. The status bar at the top shows the time '20:25' and various icons. The bottom navigation bar is visible at the very bottom of the screen.

Fig. 9 Blogs

7 Conclusion

AGROCARD is conceptually a new idea in the era of online market. While looking at the previous works, the paper aims to successfully define a new concept of farmers directly selling its stock to the customers and customers buying directly through a virtual intermediary, i.e., our system. The project aimed at providing the maximum profitability to the farmers who do not get profits due to the wholesalers who quote their own price for the stock.

With the pre-production concept, farmers can plan before cultivating the crop, by which they can get more profit. So, our system aims at providing maximum profit to the farmers through direct deals with the customers. Providing fertilizers and guiding farmers to gain good profits. According to the need of the market, the project guides the farmers that which crops to be planted to gain maximum profit.

References

1. Ranjith, Anas S, Badhusha I, Zaheema OT, Faseela K, Shelly M (2017) Cloud based automated irrigation and plant leaf disease detection system using an android application. Department of Computer Science Engineering Ernad Knowledge City Technical Campus Manjeri, Kerala, India
2. Chauhan N, Krishnakanth M, Kumar GP, Jotwani P, Tandon U, Gosh A, Garg N, Santhi V (2019) Crop shop—an application to maximize profit for farmers. School of Computer Science and Engineering, VIT Vellore
3. Grajales DFP et al (2015) Crop-planning, making smarter agriculture with climate data. In: 4th International conference on agro-geoinformatics, pp 240–244
4. Bhende M, Moheni S, Patil S, Mishra P, Prasad P (2018) Digital market: e-commerce application for farmers. Computer Engineering Department DYPIEMR, Akurdi, Pune, India
5. Li J, Zhou L (2018) Research on recommendation system of agricultural products e-commerce platform based on hadoop. School of Information Science and Engineering Guilin University of Technology Guilin, Guangxi Province, China
6. Anand VKM, Harshitha K, Chandan Kumar KN, Kumar N, Kashif Khan MK (2018) An improved agriculture monitoring system using agri-app for better crop production. Department of E&CE, SVCE, Bangalore, India
7. Reddy S, Pawar A, Rasane S, Kadam S (2015) A survey on crop disease detection and prevention using android application. Department of Computer Science Engineering, JCEM K.M. Gad
8. Madhu A, Archana K, Kulal DH, Sunitha R, Honnavalli PB (2020) Smart Bot and e-commerce approach based on internet of things and blockchain. Department of Computer Science Engineering, PES University, Bangalore, Karnataka, India
9. Bhave A, Joshi R, Fernandes R, Somaiya KJ (2014) MahaFarm—an android based solution for remunerative agriculture. Institute of Engineering & Information Technology, Mumbai, India
10. Galgalikar MM, Deshmukh GS (2013) Real-time automatization of irrigation system for social modernization of Indian agricultural system. Department of Electronics and Telecommunication Jawaharlal Darda Institute of Engineering & Technology, Yavatmal, India
11. Shiraz Pasha BR, Yogesha DB (2014) Microcontroller based automated irrigation system. Department of Mechanical Engineering, MCE, Hassan
12. Ingale HT, Kasat NN (2014) Automated irrigation system. GF's G.C.O.E, Jalgaon, C.O.E.T, Amaravati
13. Chanda C, Agarwal D, Er B, Persis UI (2013) A survey of automated GSM based irrigation systems. School of Information Technology and Engineering, VIT University, TN, India

Dielectric Recovery and Insulating Properties of Coconut Oil and Transformer Oil



T. C. Balachandra and Shreeram V. Kulkarni

1 Introduction

The intrinsic dielectric properties of some liquids have made scientists the world over to believe that they could be superior to their liquid and solid counterparts. Hence, they find their application in devices such as capacitors, cables, circuit breakers, and more importantly transformers. In transformers, apart from providing electrical insulation between windings, such liquids take away heat. The most widely studied properties are insulation strength, dielectric loss tangent, and thermal conductivity. Besides, other liquid properties like viscosity, thermal stability, specific gravity, and flash point are also studied [1]. Fine water droplets and the fibrous impurities affect the dielectric properties. However, chemical stability is often a concern. Some other factors are the cost, space, and environmental effects. Insulation characteristics of coconut oil as an alternative to the liquid insulation of power transformers have been analyzed and reported [2].

1.1 Coconut Oil

Coconut oil is abundantly available in India, Sri Lanka, and other tropical countries and is used as edible oil. The insulating properties of this oil have been a subject of study, but its insulating behavior under different conditions is yet to be understood. Generally, vegetable oils are rich in fat (nearly 90%) out of which about 65% comprise of short and medium chains and the saturated fats present in them increases the melting point. They suffer from the disadvantage of low oxidation

T. C. Balachandra (✉) · S. V. Kulkarni
Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: balachandra.tc@nmit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_62

719

resistance. The concern for future scarcity and poor biodegradability attention is focused on alternative natural esters such as vegetable oils, soya, and sunflower oil [2]. Coconut oil like most natural esters contains more free fatty acids that increase the conductivity. Some workers have experimented and proved that virgin coconut oil has more potential dielectric properties in comparison to vegetable oil, palm oil, and commercial grade coconut oil [3]. One of the main components in any power system is the power transformer, and oil/paper/pressboard insulation is considered as one of its major components. Mechanical, thermal, and electrical stresses cause deterioration of insulation during service, and insulation failure is very common. Over the last seven decades, mineral oil extracted from crude oil is a preferred candidate. Studies have proved that esters must be processed, treated, and purified to enhance its insulating properties, and this is a non-trivial task. Most distribution transformer operations are affected by insulation systems supporting them. Accidental spill and consequent environmental negative influences are a major concern in the use of mineral oils because of their poor biodegradability. Therefore, there is a strong need for a suitable substitute for mineral oil. Refined, bleached, and deodorized (RBD) coconut oil has shown superior insulating properties compared to copra coconut oil, and prototypes of transformers filled with these oils are used in Sri Lanka since 2001. It is also shown that in a relative scale, copra (direct natural extract) displays the least impressive insulating behavior [4]. Elaborate studies on aging of coconut oil and comparison with mineral oil have been reported [5]. Studies on thermal properties of vegetable are also reported [6]. Some have used insulating electrodes under a quasi-uniform AC field for their study with mineral/palm/coconut oils [7]. Some workers have experimented with blending two types of oils such as sesame oil and field-aged mineral oil [8]. Small to medium distribution transformers are currently using such insulating oils. Some other liquid dielectrics are oils extracted from rapeseed, canola, and palm. Comparison of sesame, castor, and coconut oils has also been tried using frequency domain spectroscopy by some workers [9]. The feasibility and some important insulating properties such as breakdown voltage (BDV) and dielectric recovery characteristics and a comparison with mineral oil are reported in this paper. The scope of the present study is limited to breakdown voltage studies and dielectric recovery of natural coconut oil in "as is" condition without any purification, dehydration, or neutralization processes. Standard testing procedures (ASTM D 877-02^{e1}) are followed during experimentation to obtain the BDV. The present work is focused on recovery of the dielectric property apart from studying electrical breakdown. Results are compared for mineral and coconut oil (Fig. 1).

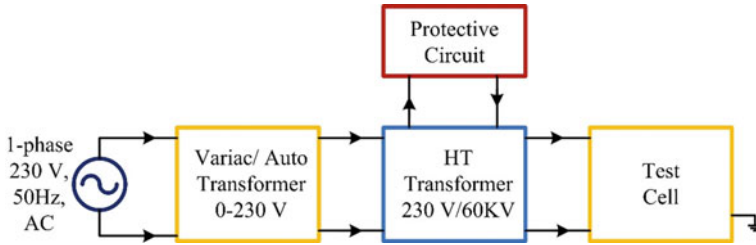


Fig. 1 Block diagram of the setup used for experimentation

2 Experimental Setup and Procedure

2.1 ASTM D 877-02^{e1} Procedures

Two test methods A and B are made available in standards documents, and the details of choices of tests are mentioned in equipment manufacturing catalogs [10].

2.1.1 Procedure A

Liquids such as askarels used in insulating/cooling liquids in transformers, cables, and similar apparatus and petroleum oils, hydrocarbons can be tested by this method. Insoluble breakdown residues are likely to settle down in dielectric liquids, especially during the time interval between tests. This is described as Procedure A. In summary, this procedure does not dictate oil sample change after each experiment. A portable BDV measuring kit conforming to IEC standards was used for measuring BDV. The rate of voltage raise is controlled at 3 kV/s. This measurement was made 5 times with a 1-min interval between each breakdown event, and average value was considered. The test cell was filled with a new oil sample, and the same steps as mentioned above were followed to obtain a new set of readings to check repeatability.

2.1.2 Procedure B

Insoluble material that does not settle during the interval between two tests is generally handled by this test. This is described as Procedure B in ASTM standards document [10]. Insulating oils used in load tap changers, circuit breakers, and equipment with heavy contaminants come under this category. After each successive breakdown, fresh oil is refilled in the test cell (Fig. 2).

Since commercial untreated samples were used, heavier particles settling down were expected, and therefore, Procedure A of ASTM standards was preferred. This test does not require refilling after each experiment (Fig. 3).

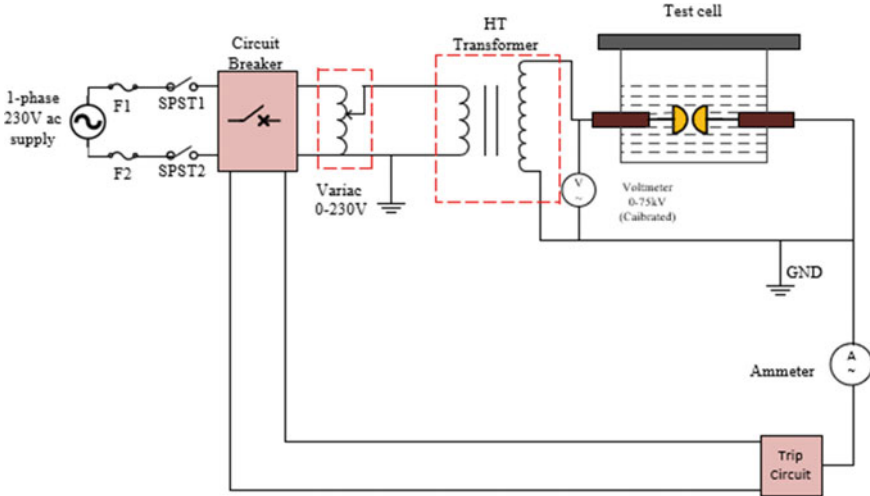


Fig. 2 Schematic diagram of experimental setup

3 Results and Discussion

The rate of rise of application of the high voltage across the gap has an influence on the observed breakdown voltage in any insulation system. However, this is considered by standards bodies like IEC and ASTM. IEC standards specify rate of rise should be 2 kV/s whereas ASTM standards specify that rate of rise should be 3 kV/s. To ascertain consistency of results, some experiments were conducted by applying both these raise times. This did not result in any significant change in the results (less than 5% variation). This is shown in Fig. 4a, b for mineral oil sample.

3.1 Continuous BDV Test on Mineral Transformer Oil

In this experiment, the voltage was raised at the rate of 3 kV/s until a voltage collapse occurred. The voltage was brought to zero, and without any change, the voltage was increased at 3 kV/s until the next breakdown occurred. Every trial was recorded and plotted as trial number as shown in Fig. 4a, b. An experiment consisting of 110 trials without changing the oil was conducted. Contrary to expectations, the BDV improved with the number of trials in case of mineral oil sample and displayed a reducing trend after 110 trials. The dotted line in Fig. 5 shows the polynomial fit to indicate the trend.

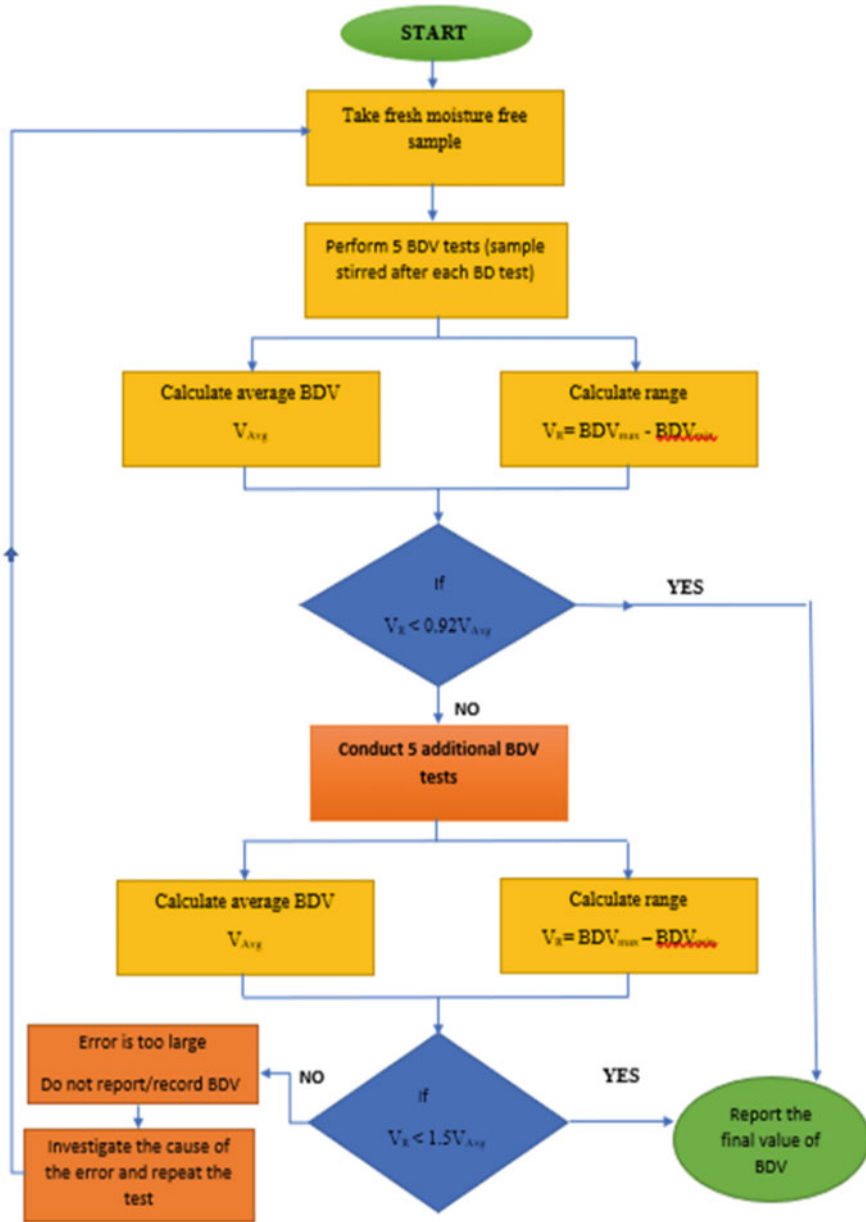


Fig. 3 Flowchart of procedure based on standard (ASTM D 877-02e1)

Fig. 4 a BDV test on mineral oil (rate of voltage rise = 3 kV/s). **b** BDV test on mineral oil (rate of voltage rise = 2 kV/s)

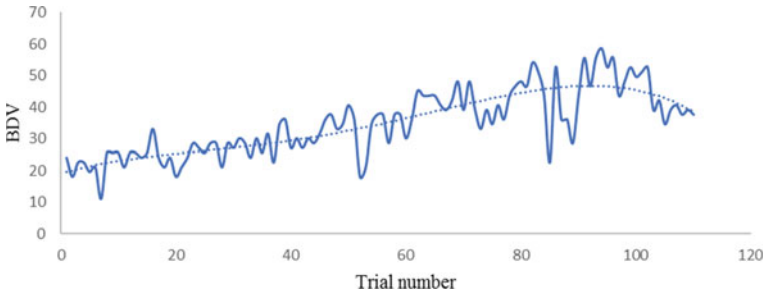
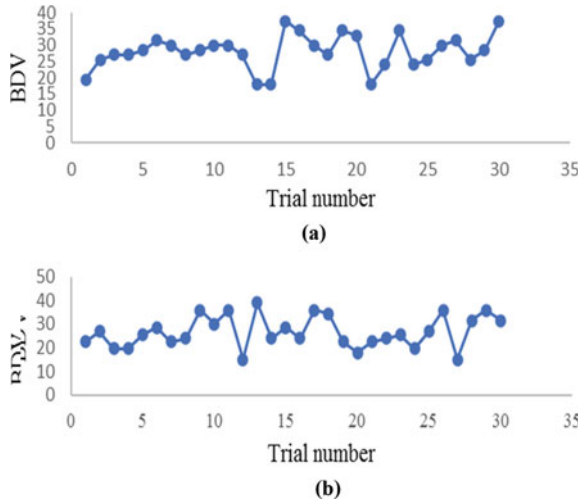


Fig. 5 BDV against number of trials for a rate of raise of 3 kV/s (Average BDV = 34.6 kV for standard test gap)

3.2 Comparative Breakdown Studies

The applied AC voltage was gradually increased maintaining a constant rate of raise as specified by the standards. This was 3 kV/s. The same procedure was adopted in all the trials. For each fresh sample, five breakdowns were allowed, and the data were recorded to obtain consistent results. Increasing the number of trials beyond five did not result in any variations. Therefore, the trials for each sample were maintained at five. The variation of average BDV (RMS value in kV) is plotted against the trial numbers as shown in Fig. 6 (for example, 1 indicates the first trial, and 6 indicates the 6th trial in Fig. 6). The BDV for mineral oil reduced from 21 to 18 kV whereas the BDV improved from 15 kV to 17.5 kV in case of coconut oil. However, the average value after 5 trials stabilized at 19.5 kV for mineral oil and 16.8 kV for coconut oil. The results demonstrate that even though the BDV reduces initially in mineral oils, after reaching a stable value, the withstand capability is not lost.

The breakdown does not significantly reduce even after 100–110 trials—see Fig. 5. This kind of dielectric recovery pattern was observable and recordable in case of mineral oil. On the other hand, it was not possible to deduce a comparison with coconut oil since commercially available coconut oil showed inconsistent results after 30 trials. Nevertheless, the present results indicate that untreated commercially available coconut oils can also be used as a dielectric, and it does not reach its flash point even after repeated application of voltage (Fig. 7 and Table 1).

In this experiment, 5 trials for each oil are conducted, and average breakdown voltage is calculated, and a graph is plotted for the same.

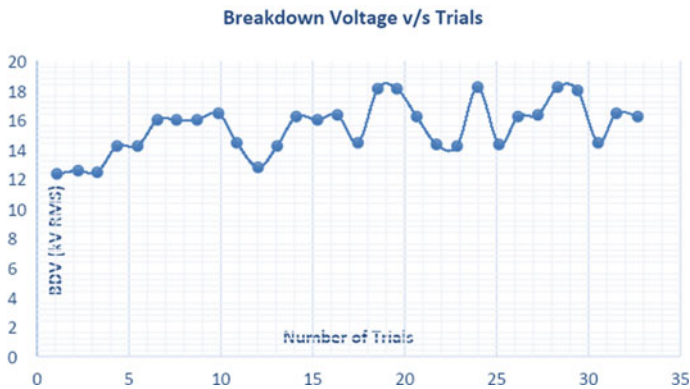


Fig. 6 Breakdown voltage against number of trials for coconut oil

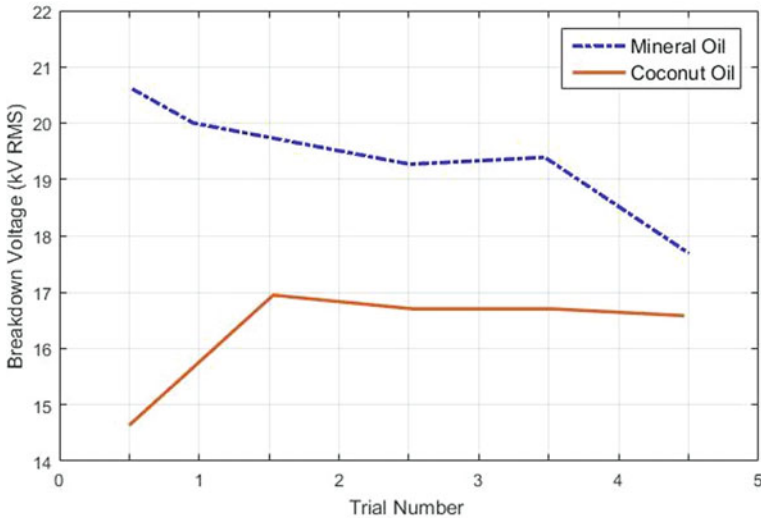


Fig. 7 Graphical comparisons between BDVs of transformer oil and coconut oil

Table 1 Table showing recorded data of BDV for transformer oil and coconut oil

Trail No.	BDV (Tr. oil) in kV	BDV (coconut oil) in kV
1	21	15
2	19.5	17.25
3	19.5	17.25
4	19.5	17.25
5	18	17.25
Avg	19.5	16.8

3.3 Repeatability Tests

Two different samples of the same type of oil were tested for BDV to check the repeatability of the results. Fig. 8a, b show the variation of the BDV with the trial number. The x -axis figures indicate the trial number (1st trial, 2nd trial, etc.), and the y -axis shows the BDV in kV (RMS). The results indicate that for 5 trials initially during the first trial and last trial coconut oil displayed larger variations—see Fig. 8a, and the variation was less for the corresponding data for transformer mineral oil. The first and last trials showed more consistency for mineral oil as seen in Fig. 8b.

4 Conclusion

The present work addresses the dielectric recovery property of both widely used mineral oil and an organic oil like commercially available coconut oil. Measurements of breakdown voltages have been customary, but the present work also throws light on the less focused “dielectric recovery.” The BDV measurement methods and the choice of appropriate method for measurement specially to test commercially available oils such as coconut oil are highlighted. The rate of raise of applied voltage (2 kV/s vs 3 kV/s) prescribed by standards bodies was found to have no significant influence on the BDV values. A manual method of tracking the total time and gradual variation of the applied voltage was found to give reasonably good repeatable results even though many automated testers are available in the market that also suffer from the disadvantage of not being periodically calibrated. The dielectric recovery of mineral oil was found to be more compared to commercially available untreated coconut oil. The experiments demonstrated that commercially available coconut oils have a potential to be used as a liquid insulator in as is condition. Further work is in progress to study the effect of different types of treatment of such oils before subjecting them to tests and the influence of suspended metallic nanoparticles.

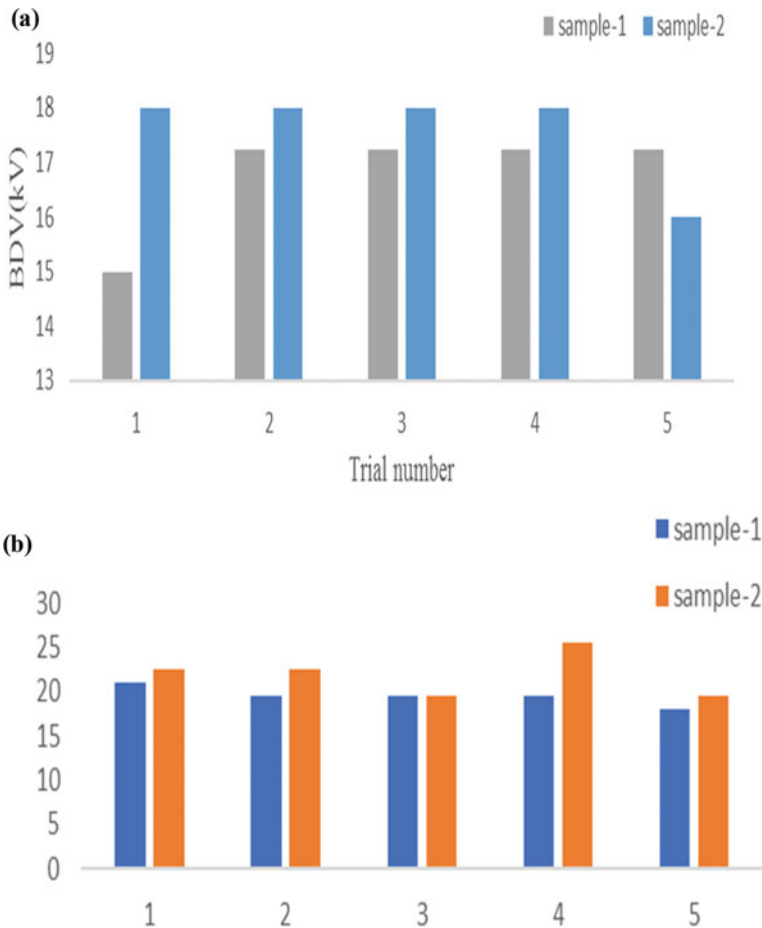


Fig. 8 (a) Test for repeatability test (coconut oil). (b) Test for repeatability test (mineral oil)

References

1. Naidu MS, Kamaraju V (2013) High voltage engineering, 4th edn. Tata Mac-graw Hill Publishing Company, New Delhi, India
2. Ranawana S, Ekanayaka CMB, Kurera NASA, Fernando MARM, Perera KAR (2008) Analysis of insulation characteristics of coconut oil as an alternative to the liquid insulation of power transformers. In: IEEE region 10 and the 3rd international conference on industrial and information systems, pp 1–5. <https://doi.org/10.1109/ICIINFS.2008.4798493>
3. Zaidi AAH, Hussin N, Jamil MKM (2015) Experimental study on vegetable oils properties for power transformer. In: IEEE conference on energy conversion (CENCON), pp 349–353. <https://doi.org/10.1109/CENCON.2015.7409567>
4. Matharage BSHMSY, Fernando MARM, Bandara MAAP, Jayantha GA, Kalpage CS (2013) Performance of coconut oil as an alternative transformer liquid insulation. IEEE Trans Dielectr Electr Insul 20(3):887–898. <https://doi.org/10.1109/TDEI.2013.6518958>

5. Matharage BSHMSY, Bandara MAAP, Fernando MARM, Jayantha GA, Kalpage CS (2012) Aging effect of coconut oil as transformer liquid insulation—comparison with mineral oil. In: IEEE 7th international conference on industrial and information systems (ICIIS), pp 1–6. <https://doi.org/10.1109/ICIInfS.2012.6304770>
6. Ahmed MR, Islm MS, Karmaker AK (2021) Experimental investigation of electrical and thermal properties of vegetable oils for use in transformer. In: International conference on automation, control and mechatronics for industry 4.0 (ACMI), pp 1–4. <https://doi.org/10.1109/ACMI53878.2021.9528278>
7. Katim NIA et al (2017) Investigation on AC breakdown of vegetable oils with insulated electrodes. In: International conference on high voltage engineering and power systems (ICHVEPS) 2017:312–316. <https://doi.org/10.1109/ICHVEPS.2017.8225963>
8. Bandara DU, Kumara JRSS, Fernando MARM, Kalpage CS (2017) Possibility of blending sesame oil with field aged mineral oil for transformer applications. In: IEEE international conference on industrial and information systems (ICIIS) 2017:1–4. <https://doi.org/10.1109/ICIINFS.2017.8300411>
9. Kumara JRSS, Fernando MARM, Kalpage CS (2017) Comparison of coconut/sesame/castor oils and their blends for transformer insulation. In: IEEE international conference on industrial and information systems (ICIIS) 2017:1–6. <https://doi.org/10.1109/ICIINFS.2017.8300410>
10. <https://hvtechnologies.com/which-oil-testing-standard-should-you-choose-to-determine-dielectric-breakdown-voltage/>

Predictive Maintenance of Lead-Acid Batteries Using Machine Learning Algorithms



H. R. Sridevi  and Shrey Bothra

1 Introduction

Quite simply, predictive maintenance (PdM) is a technique for predicting faults or system breakdowns in a failing system in order to save maintenance costs by assessing the system's present condition and/or, in a more general sense, historical data. Identifying early symptoms of a malfunction or breakdown and then initiating the necessary maintenance procedures at the proper time is how a predictive maintenance programme works. Information derived from PdM data is used in diagnostics and prognostics, and it may be used to determine what is wrong, where the fault is located, why it is happening, if it is a failure or just a defect, and when the failure will occur, if at all.

PdM has proved to be more effective than other maintenance policies, corrective and preventive. An overview of the maintenance policies is as shown in Fig. 1.

Corrective maintenance (known as unplanned or run to failure or reactive maintenance) is simply the process of letting a system to operate until it fails and then restoring it.

Preventive maintenance (also known as periodic or scheduled or regular maintenance) is maintenance work that is scheduled in advance at regular intervals [1, 2].

Using batteries to ensure the operation of critical electrical equipment is a common occurrence in a wide range of situations and applications. There are so many places where batteries are used that it would be hard to list every one of them here. It is difficult to determine the state of a battery's health, for example by a technician in a workshop without doing a comprehensive battery study, which may not be possible

H. R. Sridevi (✉) · S. Bothra
Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: sridevi.hr@nmit.ac.in

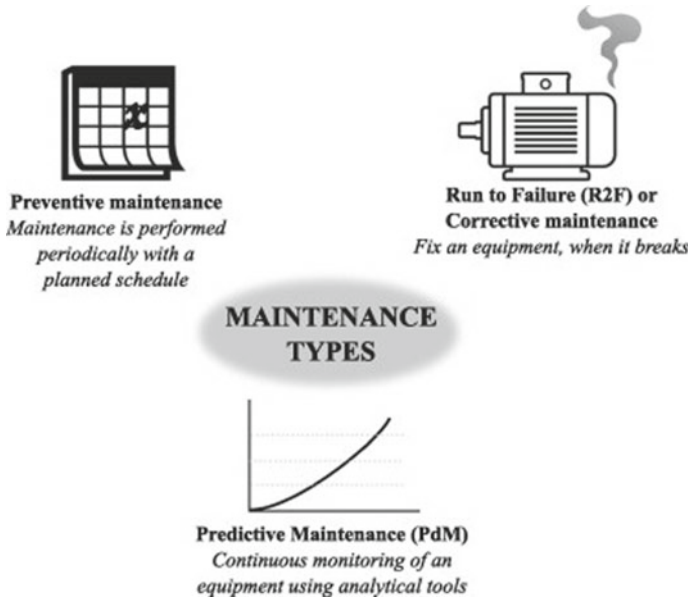


Fig. 1 Overview of the maintenance policies

due to time constraints or the cost of the operation [3]. An internal short, sulphation, corrosion, or other forms of degradation can cause a battery to fail [4].

Battery degradation can be identified by:

- (1) Reduced battery capacity.
- (2) Reduced discharge voltage.
- (3) Increase in battery internal resistance.

The formation of lead sulphate crystals on the surface of electrodes happens over a long period of time and causes the electrodes to become passive to additional electrochemical activity since it is non-conductive. Corrosion of the Pb/Pb alloy substrates in the anode is another significant source of failure. During battery recharging, the continual corrosion of the Pb/Pb alloy grid takes place. Natural oxidation of the exposed Pb grid will result in the formation of PbO_2 . Corrosion due to stress and electrochemical attack all occur as a result of the grid's ageing [5].

It is possible that unexpected battery failures will result in equipment becoming unavailable, which can be quite costly [6]. It is the goal of this study to develop prediction models for flexible maintenance of lead-acid batteries in order to extend the battery life to its maximum potential.

By adopting data-based predictive maintenance procedures, it is possible to avert unexpected battery failure.

2 Predictive Maintenance Techniques

2.1 Data Collection

The proposed battery maintenance model is based on measuring the internal resistance of battery modules to evaluate how well they are working, and it was originally created for lead-acid batteries [7].

The internal resistance of:

- (1) New/healthy batteries were discovered to be in the range of 0.1–0.3 through experiments.
- (2) Old/replaced batteries were discovered to be greater than 5.

The data collected should be accurate and comprehensive. It must include.

- (1) Date and time stamp,
- (2) The battery’s serial number and
- (3) The battery pack serial number.

The fundamental battery characteristics are listed in Table 1.

When an event happens that necessitates the replacement of a battery, the battery is labelled as failed. This type of occurrence is referred to as an event of interest (EoI) [8].

An EoI is typically classified into one of the categories listed below:

- (1) Normal/spontaneous Ageing—the resistance of a battery gradually increases as it ages, resulting in a reduction in battery capacity.
- (2) Internal Fault—a battery’s condition could deteriorate dramatically as a result of an internal fault.

Battery replacement, however, takes place only during a manual inspection. We must check historical data manually to determine the exact timing of EoI. We aim to design algorithms that automatically detect when a battery begins to degrade in order to remove the need for recurrent and significant manual labour.

Table 1 Battery attributes

Attribute	Notation
Current	I_t
Voltage	V_t
Internal Resistance	R_t
Temperature	T_t

2.2 Feature Design

Although the amount of data gathered should be significant, it is of low dimensionality. The application of feature design approaches allows us to increase the data dimension, allowing us to fully leverage the potential of big data [8].

- (1) Basic Features—current, voltage, internal resistance, and temperature are some of the basic attributes of the data acquired from the battery monitoring device. Because the current in an open circuit is always zero, in spite of the battery's health, we use the other three attributes, V_t , r_t , T_t , as our basic features.
- (2) Features related to Battery Pack—every serial battery pack comprises a specific number of discrete battery cells, which is determined by the manufacturer. We anticipate that battery pack attributes will reflect the possibility of cell failure, as damaged cells have a significant impact on a battery pack's performance.

As a result, we create several characteristics that reflect statistics of the battery pack, which include the mean voltage of battery cells μ_t^V and its empirical standard deviation σ_t^V , as shown in Eqs. (1) and (2). The mean ohmic resistance of battery units μ_t^r and its empirical standard deviation σ_t^r .

Mathematically,

$$\mu_t^V = \frac{1}{N} \sum_{i=1}^N V_t^{(i)} \quad (1)$$

$$\sigma_t^V = \sqrt{\frac{1}{N} \sum_{i=1}^N (V_t^{(i)} - \mu_t^V)^2} \quad (2)$$

where the number of batteries in a single battery pack is denoted by the letter N , and $V_t^{(i)}$ is the voltage of each individual battery cell i , $i \in \{1, \dots, N\}$ at instant of time t μ_t^r and σ_t^r . The same method was used to calculate both.

Additionally, comparing relative performance to the intra-pack average could be helpful in identifying a battery cell's health. As a result, we use two indicators in our prediction model: relative resistance and relative voltage, Rr_t and RV_t respectively, as shown in Eqs. (3) and (4).

Mathematically,

$$RV_t = V_t - \mu_t^V \quad (3)$$

$$Rr_t = r_t - \mu_t^r \quad (4)$$

- (3) Time series Features—the surveillance data are accumulated over a period of time. A number of observations highlight the importance of building features of time series, including the following:

- (4) Few attributes are very dependent on the passage of time.
- (5) By using the battery’s attributes as observations for the failing class during the final few minutes before replacement of the battery, the model will be unable to predict when the battery will need to be replaced.

As a result, we compute the rate of change as well as the gradient of specific properties across time. Mathematically, the rate of change of voltage at instant of time t , shown in Eq. (5), is defined as

$$VC_t = V_t - \text{Mean}[V_{(t-T_C):(t-T_C+D)}] \tag{5}$$

The time period utilised to calculate the rate of change is denoted by T_C , and the number of time occurrences included in a single day is D .

VG_t is the voltage gradient at time t , which is calculated by solving the least squares regression problem in Eq. (6) as follows:

$$\min_{a_0, a_1} = \sum_{i=t-T_g}^t ||V_i - (a_0 + a_1 \cdot i)||^2 \tag{6}$$

T_g is the time period during which the gradient was calculated. After finding the optimal answer, we set,

$$VG_t = a_1$$

Rate of change of resistance and resistance gradient, RC_t and RG_t , are both formulated in a similar way.

- (4) Combined features—to introduce nonlinearity into our model, the feature combination approach is employed. This generates a new feature, shown in Eq. (7), which is combined with other existing features.

$$VDR_t = V_t / R_t \tag{7}$$

Accordingly, the feature space has been expanded to include 14 attributes in total, as shown in Table 2.

2.3 Model Training

We train classification models for battery replacement based on the data that were obtained in the previous step. We anticipate that the model will produce high-quality predictions on both the training data and the testing data that have not yet been observed.

Table 2 Expanded features

Type	Feature Name	Notation
Basic feature	Voltage	V_t
	Resistance	R_t
	Temperature	T_t
Battery pack-related feature	Pack voltage mean	μ_t^V
	Pack voltage std	σ_t^V
	Relative voltage	RV_t
	Pack resistance mean	μ_t^r
	Relative resistance	Rr_t
Time series feature	Voltage change rate	VC_t
	Voltage gradient	VG_t
	Resistance change rate	RC_t
	Resistance gradient	RG_t
Combined feature	Attribute ratio	VDR_t

3 Prediction Methods

3.1 Machine Learning Algorithms

Within artificial intelligence, machine learning (ML) has emerged as a strong method for constructing sophisticated predictive algorithms in a variety of applications. In challenging environments, machine learning algorithms can manage high-dimensional and multimodal data and discover hidden patterns within data [9]. As a result, machine learning offers powerful prediction methodologies for PdM applications.

PdM based on ML can be classified into two primary categories, which are as follows.

- (1) Supervised—the information about the incidence of failures is included in the modelling data set.
- (2) Unsupervised—the data related to maintenance do not exist.

The nature of the existing maintenance policy has a significant impact on the availability of maintenance information. Because the data associated with a single maintenance cycle (the work that occurs between two consecutive breakdown incidents) is readily available in the case of corrective maintenance rules, supervised techniques can be easily deployed. However, when preventive maintenance standards are in place, the complete maintenance cycle might not be observable because maintenance is often performed far before any potential failure, making only unsupervised learning approaches viable. When feasible, supervised solutions are clearly

preferable: in this study, we investigate a supervised approach to PdM, given the overwhelming adoption of preventive maintenance practises in industries, and therefore the availability of acceptable data sets [10].

For PdM problems, regression algorithms are generally used when predicting the remaining useful life of a process/equipment, whilst classification algorithms are used when we aim to differentiate between healthy and unhealthy conditions of the system being monitored, in our case a battery. Because of their accuracy, efficiency, and ease of implementation, we chose the random forest and gradient boosting decision tree classification algorithms.

3.1.1 Random Forest

The random forest (RF) is a machine learning technique used for solving problems related to classification and regression. It employs ensemble learning, a method used for resolving complex problems by integrating many classifiers. The RF algorithm uses numerous decision trees to determine the outcome, which is dependent on the predictions of the decision trees. It anticipates by averaging the results of different trees. As the number of trees increases, the precision of the output improves.

There are two tactics in RF that makes it superior to other typical classification and regression trees: random variable selection at the split steps and bootstrap aggregation, often known as bagging. Bootstrap aggregation builds a series of new data sets by sampling evenly with replacement from the original data set and then fitting the models with them. A predictor with a variance lower than other typical classification and regression trees can be obtained by averaging over all models.

3.1.2 Gradient Boosting Decision Tree

Gradient boosting is a machine learning technique that uses an ensemble of weak learners to improve the performance of a ML model. Decision trees are generally weak learners. With the help of their combined outputs, more accurate models can be produced.

The final result of regression is derived by averaging the scores of all weak learners. The final classification result can be determined as the class with the most votes from weak students.

Weak learners work sequentially in gradient boosting. Every model strives to improve on the preceding model's.

In accuracy. This differs from the bagging strategy, in which numerous models are fitted in simultaneously on portions of the data.

3.2 *Survival Analysis*

Survival analysis is a set of statistical methods that answers questions like “how long before a specific event occurs.” It is also known as “time to event” analysis. This method was mainly established by medical researchers who were more interested in determining the predicted longevity of patients in different cohorts; however, it can also be applied in this case for predictive maintenance.

With techniques such as Kaplan Meier Estimate or The Cox Proportional Hazard Model, a survival curve estimate is obtained; predictions are drawn by calculating the probability of survival beyond a specific time t .

A flowchart showing the deployment of the algorithm to an electrical system is shown in Fig. 2.

4 Conclusion

The purpose of this study is to address the problem of anticipating the breakdown of lead-acid battery systems. ML Algorithms: random forest and gradient boosting decision tree, and survival analysis are used to solve the challenge of determining a battery maintenance policy based on historical data. The data consist primarily of sensor readings accumulated throughout the course of the battery’s life. To effectively utilise the power of big data, we used a feature expansion technique on our collected data. More frequent and regular readouts will improve model prediction performance, which is expected given the increased amount of data provided to the predictive model. This work has a lot of practical applications. The model is self-contained and requires no additional effort. As long as the fundamental characteristics of the battery are identified, the model can be easily applied to batteries manufactured by any manufacturer. The results of our method should outperform the present maintenance policies considerably.

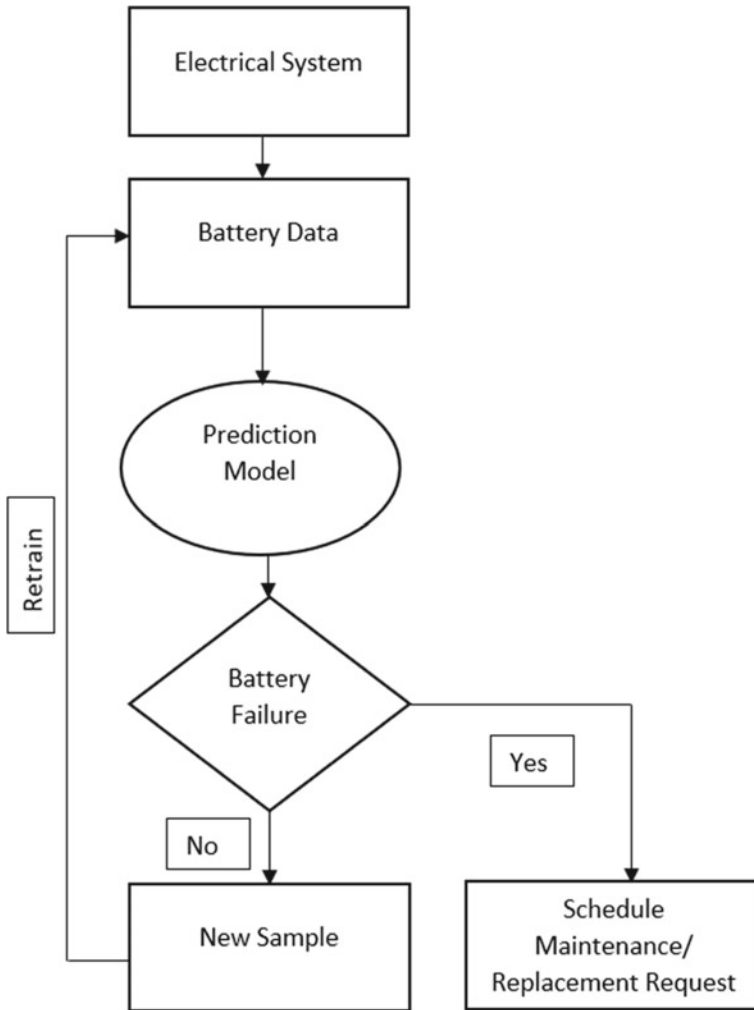


Fig. 2 Flowchart of deployment of algorithm

References

1. Selcuk S (2021) Predictive maintenance, its implementation and latest trends
2. Motaghare O, Pillai AS, Ramachandran KI (2018) Predictive maintenance architecture. In: IEEE international conference on computational intelligence and computing research (ICCIC) 2018:1–4. <https://doi.org/10.1109/ICCIC.2018.8782406>
3. Voronov S, Frisk E, Krysanter M (2018) Lead-acid battery maintenance using multilayer perceptron models. In: IEEE international conference on prognostics and health management (ICPHM) 2018:1–8. <https://doi.org/10.1109/ICPHM.2018.8448472>
4. Catherino HA, Feres FF, Trinidad F (2004) Sulfation in lead–acid batteries

5. Yang J, Hu C, Wang H, Yang K, Liu JB, Yan H (2017) Review on the research of failure mode and mechanism for lead–acid batteries. *Int J Energy Res* 41:336–352. <https://doi.org/10.1002/er.3613>
6. Voronov S, Krysander M, Frisk E (2020) Predictive maintenance of lead-acid batteries with sparse vehicle operational data
7. Gomez-Parra M et al (2009) Implementation of a new predictive maintenance methodology for batteries. application to railway operations. In: *IEEE vehicle power and propulsion conference*, pp 1236–1243. <https://doi.org/10.1109/VPPC.2009.5289709>.
8. Tang JX, Du JH, Lin Y, Jia QS (2020) Predictive maintenance of VRLA batteries in UPS towards reliable data centers.
9. Wuest T, Weimer D, Irgens C, Thoben KD (2016) Machine learning in manufacturing: advantages, challenges, and applications. *Prod Manuf Res* 4
10. Susto GA, Schirru A, Pampuri S, McLoone S, Beghi A (2015) Machine learning for predictive maintenance: a multiple classifier approach. *IEEE Trans Industr Inf* 11(3):812–820. <https://doi.org/10.1109/TII.2014.2349359>
11. Breiman L, Friedman J, Olshen R, Stone C (1984) In: *Chollet F (ed) Classification and regression trees*. Taylor and Francis

Cloud-Aided IoT for Monitoring Health Care



Aparna Manikonda  and N. Nalini

1 Introduction

The Internet of Things (IoT) is without a doubt perhaps the most reviving subject to the industry, private and public sector, and research communal. While customary web encourages correspondence between various restricted gadgets and people, IoT interfaces a wide range of associated “things” into a far-reaching organization of interrelated figuring insight without the mediation of a human. The reception of IoT and the advancement of remote correspondence advances permit patient’s ailments to be spilled to guardians continuously [1, 2]. Besides, numerous accessible sensors and versatile gadgets can quantify explicit human physiological boundaries, for example, pulse (HR), breath rate (RR), and circulatory strain (BP) through a solitary touch. Although it is as yet in the early advancement stage, organizations and enterprises have immediately embraced the force of IoT in their current frameworks, and they have seen upgrades underway just as client encounters [3].

The ascent of versatile gadgets, artificial intelligence [6], and cloud computing guarantees a firm establishment for the development of IoT in the medical services area to change each part of living souls. Because of the combination of IoT and cloud computing in the medical care area, wellbeing experts can give quicker, more productive, and better medical care administrations, which hence lead to better patient experience. Subsequently, it brings better medical care administrations, better

A. Manikonda · N. Nalini (✉)
Research Scholar, Dept. of CSE, NMIT, Bangalore, India
e-mail: nalini.n@nmit.ac.in

A. Manikonda
e-mail: aparna.subhadra@gmail.com

N. Nalini
Professor, Dept. of CSE, NMIT, Bangalore, India

patient experience, and less desk work for wellbeing experts. Specifically, IoT-based advances have as of late become famous for making non-intrusive patient wellbeing status observing where different clinical gadgets, sensors, and demonstrative gadgets can be seen as shrewd gadgets or items comprising a centerpiece of the IoT. Even though IoT can be applied in numerous clinical applications, the proper asset of the executives of a lot of observed information is been put away in cloud workers to dispense with paper-based works.

Indeed, there are a few limitations identified with the center IoT gadgets, for example, restricted memory, power supply, and preparing capacities that adversely impact the exhibition of the organizations. Also, they are exclusively centered around a single application based on clients. This powers every medical care supplier to send a customized checking network, which confines sharing of the actual sensors with different associations especially those that are not fundamentally associated with similar security perspectives [4]. As these customized networks require free administrations regarding their assets, for example, correspondence and organizations, the expense is expanded relatively. Nonetheless, the mix of IoT innovation in the medical services brings a few difficulties, including information stockpiling, information the board, trade of information between gadgets, security, and protection, and brought together and omnipresent access. One potential solution that can address these difficulties is cloud computing innovation. Figure 1 shows an ordinary medical care framework that coordinates both IoT and cloud computing to give the capacity to get to shared clinical information and normal foundation pervasively and straightforwardly, offering on-request benefits, over the organization, and performing tasks that address developing issues.

IoT-based advances interfacing clinical checking gadgets through cell phones to cloud stages have as of late become prominent for making non-intrusive patient wellbeing status observing [5, 6]. The stage can decrease the expense of medical care as administrations can be shared by various end clients. Additionally, applying an IoT-WBAN stage in a wellbeing checking framework improves the endeavors of framework usage by sharing data on that specific stage, in this way upgrading the coordination and tasks [7].

2 Cloud-Assisted IoT Application Trends in Health Care

2.1 *Patient One-To-One Care*

As of late, IoT and cloud registering innovation have indicated a vital job in far-off patient checking applications because the associated gadgets let medical care suppliers and doctors notice patients distantly. This pattern prompts fewer admissions to the clinic, more agreeable administrations, and activity cost decrease. The primary component of patient observing is different sorts of sensors and wearable gadgets. They help medical care experts in noticing and analyzing patients' vital organs and

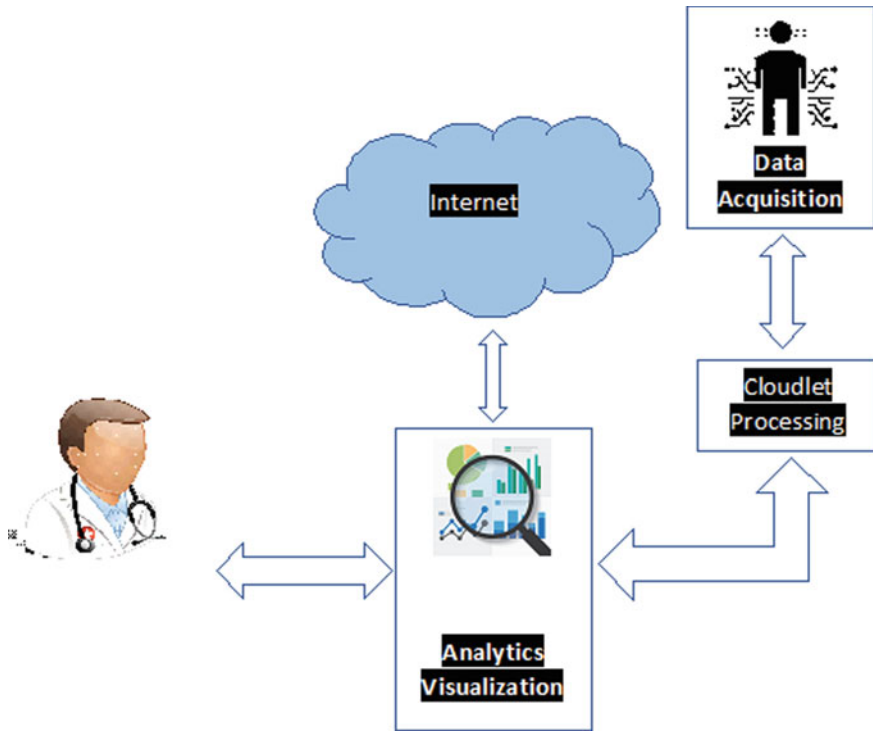


Fig. 1 Framework of cloud-assisted IoT in health care

indications depriving their physical presence. With the help of fitting segments in the patient checking structure, it will end up being an early admonition framework for potential clinical manifestations that could be perilous to the patient whenever left untreated.

2.2 Telemedicine

An expanding prerequisite from another age of the educated populace has pushed for fast reception of telemedicine due to its accommodation, efficiency, and canny highlights. Telemedicine empowers the distant conveyance of medical care administration, so patients can be dealt with distantly utilizing broadcast communications innovation. The discovery in innovation and medical services advancement has significantly improved its convenience and making it an essential piece of far-off patient checking. As of late, on account of the improvement of IoT and cloud processing, telemedicine innovation will see significantly more upgrades that help the correspondence among specialists and patients across existence.

2.3 Drug Supply Chain Management

The expanding utilization of IoT-associated gadgets for correspondence, following, and the executives of the drug production network are improving the current on the way medicine the executives. Progressed medicine of the board carries numerous advantages to customers and the drug business. Another advantage of smearing IoT innovation in the production network is that stock is pulled by request from the power source as opposed to being strapped by the manufacturing plant, which makes the production network more effective. The development of IoT advances, particularly sensors, has supported sensor precision, productivity, and cost-adequacy. Thus, greater climate factors in the production network can be determined precisely at little expense.

3 Cloud-Assisted IoT Technology Trends in Health Care

3.1 Bluetooth

Advancements in wireless associations created conceivable by Bluetooth can naturally log data from clinical hardware like stethoscopes, pacemakers, or glucose screens into PCs or electronic logs, saving the two specialists and patients' significant time and making clinical gear more available. Bluetooth-empowered wellness hardware can make it simpler to follow practice schedules, with pulse screens and GPS trackers that naturally report courses or calories consumed to smartphones or sound system earphones that stream music from a cell phone without wires to impede your exercise [8]. IoT in Bluetooth is being used in various ways, out of which are as shown below.

3.1.1 Bluetooth-Empowered Gadgets

Bluetooth-empowered gadgets trade information, yet also directions. Similarly, as Bluetooth gadgets in a home can kill machines when movement indicators demonstrate that no one is in the room, a Bluetooth-empowered gadget like a glucose screen can send guidelines to change the pace of an IV dribble to more readily suit the patient's insulin levels. Permitting clinical gadgets themselves to make these standard conclusions save clinical staff time, while the gadgets can likewise convey a misery sign to doctors at whatever point they experience a circumstance that needs human clinical judgment.

3.1.2 Patient Screening

Outside clinics, Bluetooth-empowered clinical gadgets screen outpatient progress or track patients needing critical clinical consideration. Bluetooth gadgets, for example, pace producers can send readings to a smartphone for capacity. The patient at that point presents the information to the doctor on the following emergency clinic visit or has the smartphone naturally send the information to the medical clinic over a remote association. In case of a crisis, a Bluetooth-empowered gadget can even provoke a smartphone to settle on a crisis decision—complete with the patient’s area if the smartphone has GPS abilities.

3.2 Light Fidelity (LIFI)

Consistent checking of patient’s ailment in the clinic is either manual or remote constancy (Wi-Fi)-based framework. Wi-Fi-based framework turns out to be delayed in speed because of dramatically expanded adaptability. Another remote correspondence innovation called LIFI was developed by Harald Haas—a German researcher. In this situation, light fidelity (Li-Fi) finds the spots any place Wi-Fi is relevant with extra highlights of high-velocity information organization. Aside from the speed factor, Li-Fi is more reasonable in medical clinic applications for observing the patient’s conditions without recurrence impedance with the human body, improving the patient’s ailments as well as correspondences among the doctors and clinicians. Remote innovation with the Li-Fi framework empowers clinicians to screen patients distantly and give them convenient wellbeing data, updates, and backing [9]. Li-Fi innovation enhances the clinical field to the following level and has plenty of benefits when introduced and utilized advantageously. A portion of the benefits of LIFI incorporates high transfer speed, higher transmission rate, and it can work in regions that are defenseless against radio-recurrence impedance, for example, planes or medical clinics.

3.3 Near-Field Communications (NFC)

Near-field communication (NFC) is a quick, instinctive innovation that allows you to connect safely with your general surroundings with a basic touch. NFC remote closeness innovation is accessible in billions of smartphones, tablets, buyers, and mechanical hardware—with new gadgets showing up practically day by day. NFC is playing a major role in assuming a significant part in medical care by decreasing expenses, expanding productivity, and improving results [10]. Below are some ways of NFC in health care listed as follows:

3.3.1 Instant Apprisers on Patient Care

NFC allows you to follow where individuals are, and who's done what. Clinical staff can know, progressively, where a patient is, the point at which the attendant last visited, or what treatment a specialist just managed. NFC labels and NFC-empowered wristbands can supplant the customary armbands worn by patients and can be refreshed with constant data, for example, when a prescription was last given, or which system should be performed.

3.3.2 Intelligent ID Bracelets

NFC helps in regular circumstances, as well. Individuals with perilous conditions, for example, diabetes, asthma, or hypersensitivities to food or meds, can supplant their metal "Surgeon Alert" armbands with NFC-empowered wristbands that can give more noteworthy detail to specialists on call in a crisis.

3.3.3 Home Monitoring

NFC opens up additional opportunities for home observing since a NFC-empowered wristband can be designed to follow fundamental signs. The patient taps the wristband to a smartphone or tablet, and the clinical information is sent to the specialist's office, where a clinical expert can check it. Individuals who have constant conditions can see a guardian less frequently, and individuals who are recuperating from a sickness or medical procedure can get back sooner.

3.3.4 Safer Medications

At the point when NFC labels are added to a medicine's bundling or marking, tapping the tag with a smartphone or tablet allows you to confirm the prescription's legitimacy, see insights regarding measurements, or read about results and medication collaborations. The tag can likewise give admittance to web joins, to get more data, demand a top off, or contact a clinical expert. These are just some of the ways NFC can enhance health care.

4 Security Challenges in Cloud-Assisted IoT

The physical measures are associated with the sensors in IoT and virtual machines in the cloud. Some of the measures are discussed below.

4.1 Memory Requirements

The memory of IoT gadgets is little, and the vast majority of the gadget's memory is utilized to store an implanted working framework. Thus, the framework that utilizes IoT registering gadgets has restricted memory to perform complex security conventions.

4.2 Speed of Calculation

Almost all IoT registering gadgets have low-power processors; the processor needs to play out various errands including overseeing, detecting, breaking down, saving, and speaking with a restricted force source. Subsequently, power of the processor to do the security method is a difficult issue.

4.3 Power Utilization

Most IoT gadgets have a low battery limit. Thus, there is an instrument that compels them to consequently enter the force saving mode to save power at sensors' inactive time. Along these lines, it is hard for IoT gadgets to perform security conventions constantly.

4.4 Scalability

There is a sharp ascent in the quantity of registering gadgets in the IoT organization. Consequently, it is trying to locate the most appropriate security calculation for the developing number of gadgets in the IoT in the medical care organization.

4.5 Communication Channel

IoT registering gadgets generally took an interest in the organization through numerous remote correspondence conventions. Thus, it is trying to locate a standard security convention that is reasonable for different remote correspondence conventions.

5 Cloud-Assisted IoT Benefits and Impacts

There are plenty of applications utilizing cloud-assisted IoT technologies. This section discusses the three main applications in this area:

5.1 *Smart Environments and Smart City*

The smart city framework is executed by joint effort among government and public and private associations. Albeit, smart urban communities' idea is achieving exposure these days, and there is no single existing city that achieves all necessities for a smart city [11]. A portion of the smart innovations pertinent for smart urban areas incorporates energy, structures, portability, authority and instruction, and medical care. Smart city applications usage uses thoughts from the region of computerized reasoning, implanted figuring, AI, cloud registering, heterogeneous organizations, and biometrics. Likewise, it utilizes various parts including sensors, RFIDs, registering, and organizing objects to amplify the use of assets in various applications. Overseeing different administrations in a smart city requires an unpredictable organizational foundation. Smart city applications are observing and recording the residents' private data consequently, and it is basic to appropriately make sure about this information. Potential dangers to smart urban areas framework are listening in, burglary, forswearing of administration, the disappointment of equipment or programming, producing bugs, lacking testing, and catastrophic events.

5.2 *Telemedicine or E-wellbeing*

Security of wellbeing information is guaranteed by passive consent with individual information insurance laws and guidelines. This will advance the acknowledgment of the e-wellbeing framework by clients with high certainty that their private wellbeing information is ensured and secure [12].

IoT has a high likelihood to progress human wellbeing and security. In wellbeing sciences, advancements have been created lately dependent on their abilities to screen different wellbeing boundaries that can be currently sent by wellbeing gadgets utilizing a passage onto secure cloud base stages where they are put away and broke down. Information accumulated from these gadgets can be put away and examined by the clinical experts causing them to determine and empowering the likelihood to screen the patient from any area and reacting opportune way, in light of the alarm received.

5.3 Smart Transportation System

Smart transportation framework is an application that using a few innovations meaning to improve security, versatility, and capability in open transportation [13].

A portion of the advances significant for the wise vehicle includes:

Traffic control: Smart traffic signals regulate innovation plans to limit the measure of time that vehicles devote standing by and guarantee the flat progression of traffic.

Public transport reconnaissance: As the public travel people create, it ends up being logically basic to utilize a perception structure on the public vehicle to ensure the prosperity and security of public transportation. The public vehicle can remotely be noticed and take an action against any disasters/events.

Parking: A smart stopping controls stopping sensors to give productive use of the parking spots and using time and energy.

6 Conclusion

This paper is helpful for perusers who are keen on learning various parts of IoT and cloud processing in medical care. It bids a total IoT and cloud processing structure for medical services that underpin submissions in using the IoT and cloud registering spine and gives a stage to encourage the transmission of clinical information amid clinical gadgets and far-off workers or cloud registering stages. Medical care associations need creative what's more, savvy techniques to help medical care suppliers discover more beneficial approaches to address developing quantities of patients' information. Electronic Health Records (EHR) combined with cloud registering frameworks bring an answer for the "huge information" challenge where it obliges capacity assets and encourages the way toward sharing patients' information between medical care suppliers. From that point forward, we gather existing innovative work measures in the medical services industry by segments, applications, what's more, end-client, and afterward, critical accomplishments that demonstrate the adequacy of incorporating IoT what's more and cloud registering in medical services were depicted.

References

1. Abidi B, Jilbab A, Haziti ME (2017) Wireless sensor networks in biomedical: wireless body area networks. In: Europe and MENA cooperation advances in information and communication technologies. Springer, Berlin/Heidelberg, Germany; pp 321–329
2. Xu Q, Ren P, Song H, Du Q (2016) Security enhancement for IoT communications exposed to eavesdroppers with uncertain locations. *IEEE Access* 4:2840–2853
3. Scuotto V, Ferraris A, Bresciani S (2016) Internet of things: applications and challenges in smart cities: a case study of IBM smart city projects. *Bus Process Manage J* 22:357–367

4. Stergiou C, Psannis KE, Kim BG, Gupta B (2018) Secure integration of IoT and cloud computing. *Future Gener Comput Syst* 78:964–975
5. Truong HL, Dustdar S (2015) Principles for engineering IoT cloud systems. *IEEE Cloud Comput* 2:68–76
6. Minh DL, Sadeghi-Niaraki A, Huy HD, Min K, Moon H (2018) Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network. *IEEE Access* 6:55392–55404
7. Jiang Y, Cui S, Xia T, Sun T, Tan H, Yu F, Su Y, Wu S, Wang D, Zhu N (2020) Real-time monitoring of heavy metals in healthcare via twistable and washable smartsensors. *Anal Chem* 92(21):14536–14541. <https://doi.org/10.1021/acs.analchem.0c02723>
8. Srimathi B, Ananthkumar T (2020) Li-Fi based automated patient healthcare monitoring system. *Indian J Public Health Res Dev* Feb2020 11(2):393–398 6p
9. Ganiga R, Pai RM, Manohara Pai MM, Sinha RK, Mowla S (2020) Integrating NFC and IoT to provide healthcare services in cloud-based EHR system. In: Sengodan T, Murugappan M, Misra S (eds) *Advances in electrical and computer technologies*. Lecture Notes in Electrical Engineering, vol 672. Springer, Singapore. https://doi.org/10.1007/978-981-15-5558-9_33
10. Sengupta S, Bhunia SS (2020) Secure data management in cloudlet assisted IoT enabled e-health framework in smart City. *IEEE Sens J* 20(16):9581–9588. <https://doi.org/10.1109/JSEN.2020.2988723>
11. Birje MN, Hanji SS (2020) Internet of things based distributed healthcare systems: a review. *J Data Inf Manage* 2:149–165. <https://doi.org/10.1007/s42488-020-00027-x>
12. Hussain S, Mahmud U, Yang S (2020) Care-talk: An IoT-enabled cloud-assisted smart fleet maintenance system. *IEEE Internet Things J*. <https://doi.org/10.1109/JIOT.2020.2986342>

Energy-Efficient Dynamic Source Routing in Wireless Sensor Networks



Dileep Reddy Bolla, P. Ramesh Naidu, Jijesh J J, Vinay T.R,
Satya Srikanth Palle, and Keshavamurthy

1 Introduction

Nowadays, immense advancements are occurring in the fields of wireless sensor network (WSN). WSN is a combination of numerous sensor hubs which communicate with each other using the wireless channel. Communication has turned out to be critical for exchanging data between nodes from one point to another at the conditions of high energy consumption. The challenging energy conservation model of WSN leads to a critical impact on various domains like health, manufacturing, and safety [1, 2].

The fundamental issues of the WSN are the high power consumption by the nodes, end-to-end delay, and throughput. Every node in WSN acts itself as a router to transmit the data from one end to another [3], so every node needs to be active for potential communication, due to this scenario huge power consumption at every node level became an obvious factor, this eventually degrades the performance of the sensor networks concerning its network lifetime, as routing is the key essence of wireless networks functioning mainly with the role of network hubs in which every system hub acts as a router with continuous energy usage, several routing algorithms manage

D. R. Bolla (✉) · P. R. Naidu · V. T.R

Department of CSE, Nitte Meenakshi Institute of Technology, Yelahanka, Bangalore, India
e-mail: dileep.bolla@gmail.com

P. R. Naidu

e-mail: ramesh.naidu@nmit.ac.in

V. T.R

e-mail: vinay.tr@nmit.ac.in

J. J J

Department of ECE, Sri Venkateshwara College of Engineering, Bangalore, India

S. S. Palle · Keshavamurthy

Department of Electronics and Communications, Atria Institute of Technology, Bangalore, India

the routing process much effectively till now ensuring a potential communication, but these algorithms exhibit narrow focus concerning energy dealing. The design of an optimal routing protocol that mitigates high energy consumption among nodes always suits the current scenario for effective network establishment [3, 4].

The routing protocols have been classified as hierarchical and flat routing protocols, in which flat is classified as proactive and reactive. Keeping in mind choosing an apt routing protocol is the fundamental concept in this manuscript; the design aspects among various reactive routing protocols such as AODV, DSR and proactive routing protocol such as DSDV have been analyzed and implemented through performance comparison ultimately to come out with an optimum routing scenario for energy management between nodes; here, the prime focus is on sensor nodes for effective power management through the modification of existing DSR routing protocol further to enhance the network lifetime [5–7]. So this chapter introduces the modification of the existing DSR routing algorithm to mitigate high energy consumption among sensor nodes through residual energy measuring mechanism, apparently, the modified protocol intends to route the data between the nodes that are with high residual energy levels and gives up the least energy node path, ultimately for power saving among those to improve the network lifetime. The manuscript involves in dealing with the prior related implementation in Chapter 2 ascertaining the possible aspects in dealing the framed objectives, routing protocol classification and residual energy measuring mechanism dealt in Chapter 3, performance aspects considering various parameters discussed in Chapter 4 followed by results and its discussion in Chapter 5 concluding the complete scenario at the end.

2 Related Work

The WSNs are mainly classified based on the proactive or reactive protocols, the classification of routing protocols is proactive energy routing protocol and the table-driven routing protocols, and in this, each node here makes an attempt in order to maintain consistency and updates the information to the neighboring nodes in the network [1–3].

In this research work, we have made the necessary changes in the network, and thus, the routing protocol forwards the route, and it can use the distance vector algorithms or either link-state algorithms which contain the next-hop address and in turn the destination address [4]. Further, the routing protocols can be categorized based on the reactive energy-aware routing which is of an on-demand routing and this establishes the route through the route discovery and the routes are been maintained. From source node the routing is been established based on the route discovery process, and thus, it forwards the route requests once the route requests are established the route establishes the connection to the destination nodes. Further, this can be communicated by sending the route-reply back to the source node through the neighboring nodes or the intermediate nodes, and the route gets updated [8, 9].

The routes established need to be maintained properly by using the process of internal data using the cache of the route until the destination becomes inactive or the route gets dissolved [10, 11]. Then, the route needs to be reestablished based on the requirements, whereas in the table-driven routing protocols the route maintenance needs to be done for every node. In this aspect, the DSR, dynamic source routing, and AODV, ad-hoc on-demand distance vector are considered for the performance evaluation along with the proposed protocol [12–14].

The complete wireless sensor networks are understood basically by deploying the N no of sensor nodes in the WSN scattered system on a large scale. The routing in the wireless sensor networks can be categorized as distributed or centralized and further this is been also classified as proactive, reactive or hybrid routing protocols.

3 Routing Protocol Classification

In a routing protocol, the information from the surroundings is collected and recorded by the sensor nodes and sent to different stations and end-users. A large portion of the past work on routing in wireless networks involved tracking and keeping up the correct path to the destination.

During the transfer of data, the nodes select the optimal path. The distributed algorithm executed by the sensor node will produce a typical routing table in order to minimize resource utilization. The selection of an optimal routing algorithm is the obvious factor for the sensors in order to perform well; both the reactive and proactive routing protocols serve this requirement for effective communication.

3.1 Proactive Routing Protocols

In proactive or table-driven routing protocols, the mobile node routing information is broadcasted to the system to track the destination path. The routing information regarding the number of nodes and hops in the surrounding is maintained in the routing table of each node. All the nodes endeavor to keep the updated routing data in the entire hub of the system even though the network topology changes. Hence in this type of routing protocol, the route is previously known so the information can be sent immediately. The optimized link-state routing (OLSR) and destination-sequenced distance vector (DSDV) are the popular proactive routing protocols. The disadvantages are the individual quantity of data for maintenance and slow response on failures along with restructuring.

3.2 Reactive Routing Protocols

In the case of on-demand routing to reactive routing, the maintenance made is nominal in routing protocols based on the movement of the system hubs. The discovery of the routes is performed only when a sourcing hub wishes them and in turn, sets up a connection to send and receive the packets. In these protocols, route discovery and route maintenance are the key fundamental factors/strategies. Based on the requests received from the routes, the data is sent from the source to the nearest hubs and after that the requesters are forwarded to the nodes adjacent to them and this process is known as the route discovery process.

The target node receives the route request a route-reply packet is reversed through the neighbors to the source hub. In the process of route maintenance, the route is established but if suddenly route failure is faced then another route is established. As time passes, each node tries to learn the routing paths. Dynamic source routing (DSR) and ad-hoc on-demand distance vector (AODV) are well-known reactive routing protocols. The disadvantages are extreme flooding leads to clogging of network and to find route high latency time is required.

3.3 Destination Sequenced Distance Vector (DSDV)

The hybrid routing is a combination of both reactive and proactive behaviors having the potential to propose higher scalability. Routing is primarily fixed with some table-driven prospecting routes and the demand is served from the nodes which are additionally activated through on-demand driven flooding. Zone routing protocol (ZRP) is one of the hybrid routing protocols. The disadvantages are both amount of additional hubs activated and the response to traffic demand depend on traffic gradient volume.

3.4 Ad-Hoc On-Demand Distance Vector (AODV)

This proactive routing protocol was developed based on the Bellman–Ford algorithm for wireless networks [2] where each hub maintains a table to find the shortest path from hop to every other hop.

Each node frequently updates destination path information because of the random topology. The neighbor hub exchanges the routing table data, and every hub updates the new routing data. The data is cached if it cannot discover its destination.

At that point, information packets are permitted to receive till the capture report appears from the destination. In this routing protocol, a most extreme size of buffering is accessible in memory to gather those information packets until the routing data is not received. Low latency and loop-free paths are some of the advantages of this

protocol. The disadvantages are multipath routing is not supported; bandwidth is occupied by unused paths and consumes more bandwidth because of overhead due to larger network.

3.5 Dynamic Source Routing (DSR)

The AODV is a routing protocol for wireless networks developed by Perkins, E. Belding-Royer and S. Das which is used in ZigBee. This routing protocol has the capability of both unicast and multicast routing. The path to the destination is found in AODV when the hub requests to send the data; then, routes are maintained from the source as long as required. Whenever a request is received from the AODV router for message transfer, the presence of the route in the routing table will be checked. The routing table includes target address, address of next node, target sequence number and node count. In AODV, once the route exists, message is sent by the router to the subsequent node. Or else information is sent in a queue and to find a route the route request (RREQ) is initiated. In the AODV protocol for route discovery route request (RREQ) and route-reply (RREP) messages are used in the network. For any route repairs, it will broadcast route error (RERR) and HELLO messages. If RERR is received by the source node, then the route discovery process starts. The routing overhead in AODV is potentially less.

3.6 Dynamic Source Routing (DSR)

DSR protocol is introduced by David B. et al. in 1994; this protocol is designed thinking of multi-hop communication in wireless networks. As these networks majorly work with ad-hoc networks, so they do not require any networks which are existing, and in this aspect, they work based on a self-configuring mechanism in discovering the route. In this process, we have two phases, firstly the route discovery and secondly the route maintenance. initially after establishing the route; the routes are maintained by updating the information. However, this protocol does not support the multicasting and the packet headers.

3.7 Energy-Efficient Dynamic Source Routing (EEDSR)

The energy-efficiency DSR is been proposed mainly for the MANET application to reduce the energy consumption by using the energy efficiency route metrics and the routes are been established based on the minimum hop count metrics, and further, the routes are been established based on the minimum energy routes such the network lifetime seems to be improved in the proposed work algorithm or methodology used.

The EEDSR protocol mainly focuses on minimizing the end-to-end transmission energies for the transfer of packets and is also based on the established path that takes minimum energy, based on the weighted combination of both. And, the added feature here is based on the approach for energy efficiency is known as minimum energy transmission power routing (METPR); using this novelty, we can save energy compared to that of the existing DSR protocol. This can be estimated using the total energies of the energy consumption of each hop communication.

So in this scenario, we have varied the packet size and analyzed the performance of the network as in Algorithm-1. It is been observed that the transmission energy differs from the shortest hop routing if the nodes can adjust the transmission power levels. In this scenario, multiple short hops are comparatively advantageous.

Algorithm 1: Proposed EEDSR Protocol

- 1: Initial network with N nodes
 - 2: Select a node S as source node
 - 3: Select the neighboring nodes from N nodes
 - 4: Select the node D as a destination node
 - 5: Establish the delay parameter in the channel
 - 6: Establish the packet type with the parameters like energy path and hop count
 - 7: Calculate the energy consumption of the available routes
 - 8: Initiate the route request and receive the route reply
 - 9: initial hop count = 0 and energy = 0.00 mW
-

In the routing process, we have a sender node and a destination node; when a sender node wants to communicate to a destination node, based on its route cache, it decides which route to be selected, and further best route out of the existing route will be chosen if there is no route that exists; then, the route discovery process may be initiated to establish a route. Further, if any route is available in the existing routing table, it will establish the route; else, the route has to be discovered. The process of route discovery can be formed based on broadcasting the route request packet over the network.

4 System Model and Performance Metrics

The amount of energy level in a node is represented by the energy model. At the starting, the node has an initial energy value, and when the packet is transmitted and received, it has transmitted and receives power level. The help of a power proficient routing matrix in the routing table reduces the conception of power in WSN as seen in most of the routing protocols used for energy efficiency [3–5]. Thus, power efficiency can be introduced in the routing protocol to transfer the data packet.

4.1 The Energy Model

The energy is dependent upon the power consumption in the active, sleep and idle states

$$EA = P_{active} \times T_{active} \quad (2)$$

$$ES = P_{sleep} \times T_{sleep} \quad (3)$$

$$EI = P_{idle} \times T_{idle} \quad (4)$$

$$E = EA + ES + EI \quad (5)$$

$$E = P_{active} \times T_{active} + P_{sleep} \times T_{sleep} + P_{idle} \times T_{idle} \quad (6)$$

where P represents the power consumption and P_{active} , P_{sleep} and P_{idle} represent route exists the active, sleep, idle power consumptions and where the T_{active} , T_{sleep} , T_{idle} represents the time spent by the transceiver in the corresponding states. This parameter may vary based on the number of bits to be transmitted and the bandwidth of the channel.

4.2 Throughput

It is also referred to as the packet delivery ratio. It is the ratio of the total number of packets received at the destination to the total number of packets sent from the source node as in Eq. (7).

$$\text{Throughput} = \frac{\text{TotalpacketsReceived} * 8}{(\text{LastPacketReceived} - \text{FirstPacketReceived})} \quad (7)$$

4.3 To-End Delay (Average)

The average time taken by the packets to reach the destination, wherein which all the delays like queuing delay, interference delay, route discovery delay, etc., is also referred to as path optimality or E2ED as shown in Eq. (8).

$$E2ED = \frac{\text{sum of all the delays of each CBR packet received}}{\text{Number of CBR packets received}} \quad (8)$$

4.4 Average Jitter

It is defined as the variability over time of the packet latency across the network. If any network has a constant latency, it means it is having no jitter. If a network with N packets where $i = 1-N$, and if N is greater than 2, then the jitter can be expressed as in Equation (9).

$$J = \text{delay}(i + 1) - \text{delay}(i) \quad (9)$$

Average jitter can be calculated as in Eq. (10).

$$J_{\text{avg}} = \frac{J(1) + J(2) + J(3) + \dots + J(N - 1)}{N - 1} \quad (10)$$

4.5 Energy Consumption

The energy can be consumed based on the mode of operation of the corresponding node. It is further observed that when an idle node and sleep node also consume the power the simulation may go worse in this scenario. It is been noted that the idle mode and sleep modes need to be taken care of in the simulation environment.

5 Simulation and Results

The simulations carried out in this protocol were in network simulator-2. The simulations carried out were compared with the existing protocols and found that the proposed algorithm outperforms with better efficiency. The simulation parameters as shown in Table 1.

The simulation parameters have been analyzed for different speeds of the nodes like 10 m/sec, 15 m/sec and 20 m/sec for the throughput and the end-to-end delay and the jitter parameters; it is observed that the proposed EEDR protocol outperforms the existing state of are routing protocols like AODV, ZRP and DSR protocols, and the simulation results obtained for the parameters proposed in Table 1 are discussed in Figs. 1, 2, 3, 4, 5, 6, 7, 8 and 9.

6 Conclusion

The research work carried out discussed various protocols and the comparative analysis has been presented in the manuscript, the throughput and end-to-end delay; jitter

Table 1 Parameters of simulation

S. No.	Criterion	Value
i	Channel type	Wireless channel
ii	Number of nodes	20 node
iii	Topology size	600 × 600
iv	Packet size	512 bytes
v	Traffic type	Constant bit rate (CBR)
vi	Antenna model	Omni antenna
vii	Mobility model	Random mobility model
viii	Parameters	Temperature, BP, HR
ix	Routing protocol	DSDV, AODV, DSR
x	DSDV, AODV, DSR	MAC
xi	Simulation Tool	NS-2.35

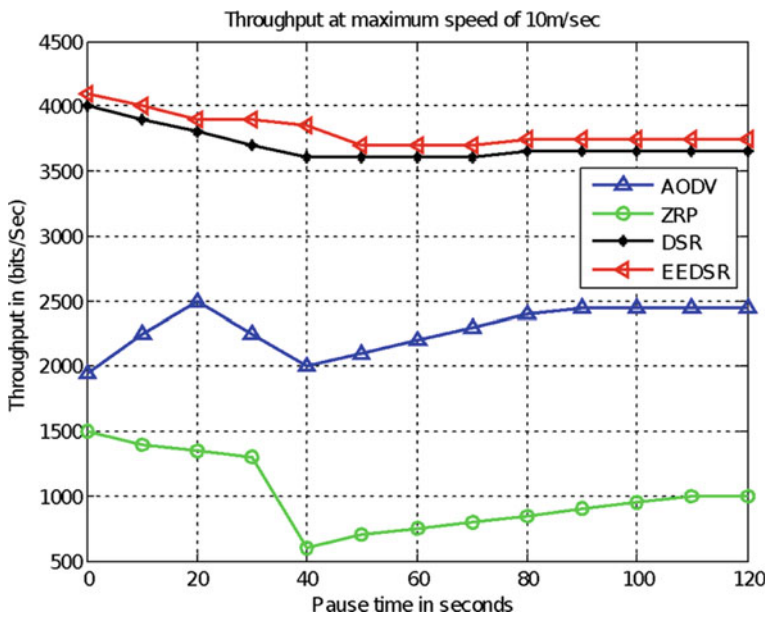


Fig. 1 Throughput for a maximum speed of 10 m/sec

was analyzed in this article for a maximum speed of 10m/sec, 15m/sec, 20m/sec. Proposed EEDR protocol outperforms the existing state of are routing protocols like AODV, ZRP and DSR protocols with respect to the communication and computation of the devices end to end. Finally, the analysis was performed comparatively. As we are aware, energy efficiency and quality of service are vital in the frequencies in low-radio. The capacity of the ad-hoc networks in wireless networks is exhibited. Finally, we calculated and evaluated the routing protocols with various speeds and node

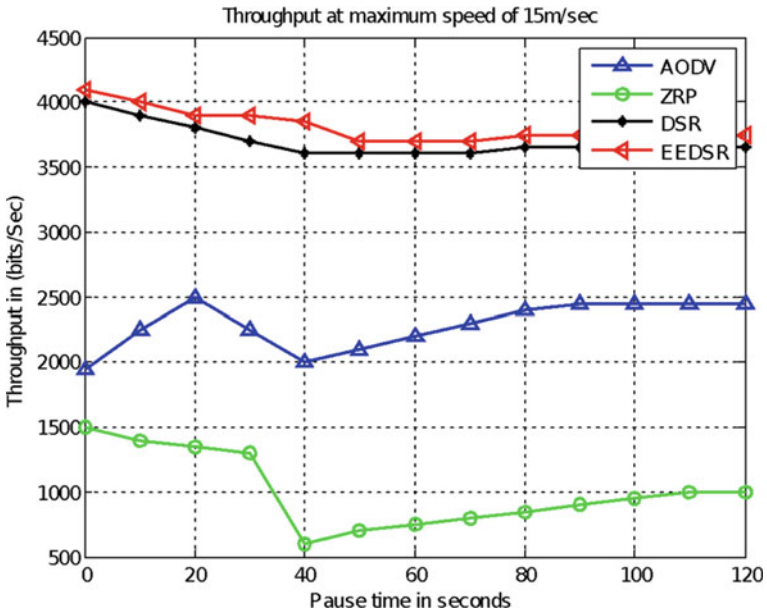


Fig. 2 Throughput for a maximum speed of 15 m/sec

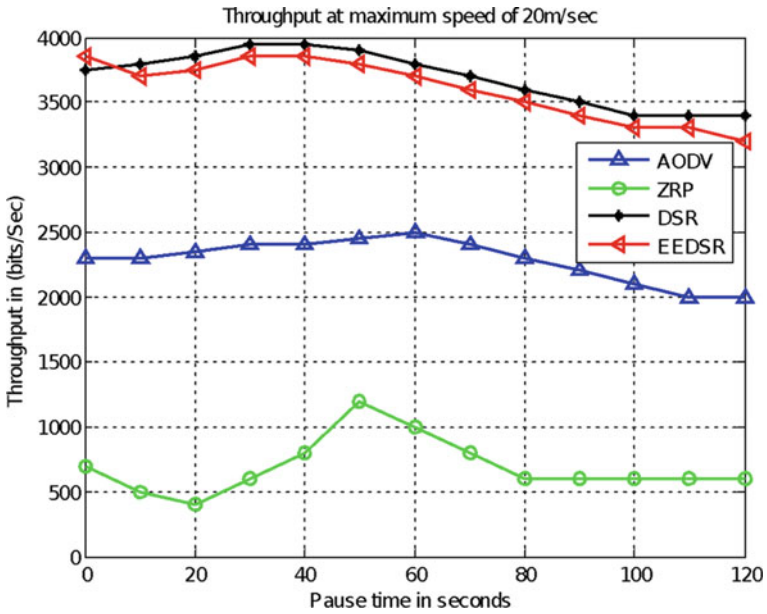


Fig. 3 Throughput for a maximum Speed of 20 m/sec

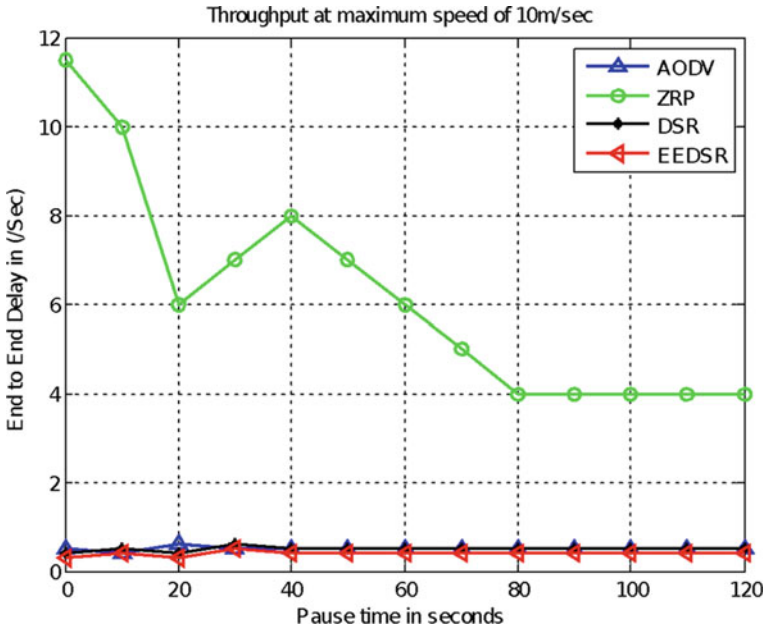


Fig. 4 End-to-end delay for nodes with maximum speed of 10 m/sec

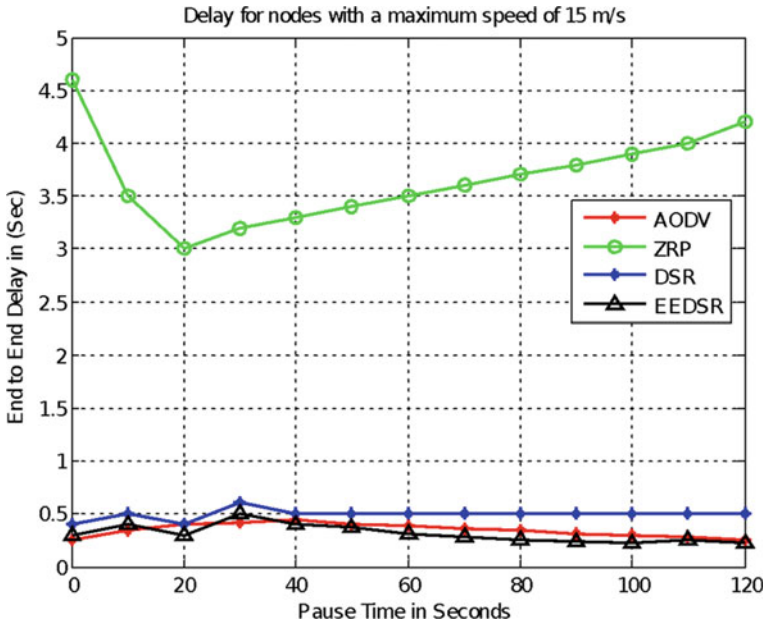


Fig. 5 End-to-end delay for nodes with maximum speed of 15 m/sec

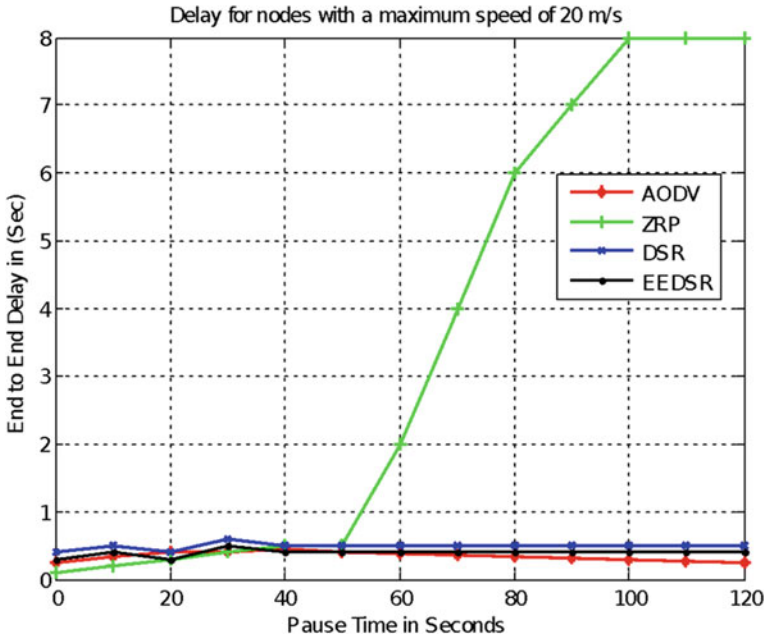


Fig. 6 End-to-end delay for nodes with maximum speed of 20 m/sec

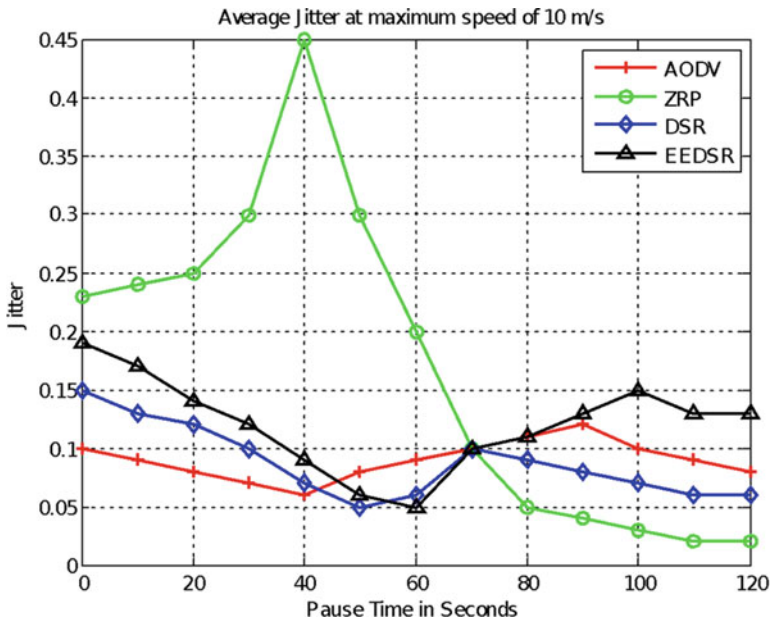


Fig. 7 Average jitter for nodes with maximum speed of 10 m/sec

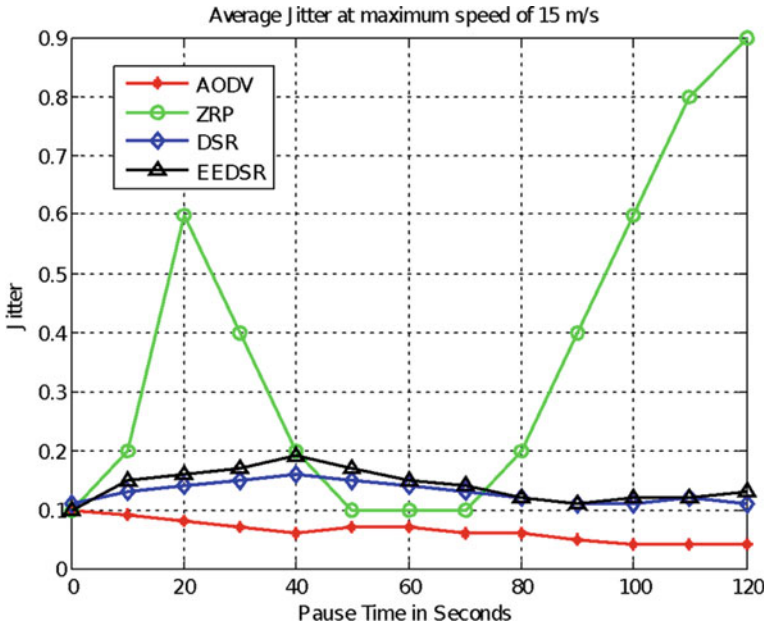


Fig. 8 Average jitter for nodes with maximum speed of 15 m/sec

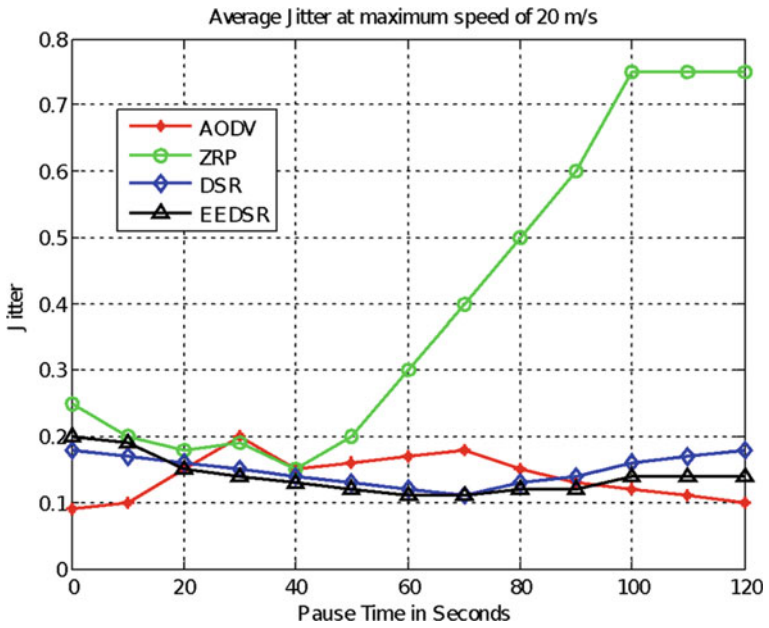


Fig. 9 Average jitter for nodes with maximum speed of 20 m/sec

densities. The proposed model is analyzed, and comparative analysis in a simulation environment is presented to achieve the Quality of Service and energy efficiency with throughput, end-to-end delay and jitter calculations. The proposed EEDSR protocol performance is analyzed in a simulator environment with a demonstration of the simulation environment in network simulator 2 environments.




References

1. Ali A et al (2020) Adaptive bitrate video transmission over cognitive radio networks using cross layer routing approach. *IEEE Trans Cognitive Commun Netw* 6(3):935–945. <https://doi.org/10.1109/TCCN.2020.2990673>
2. Ramly M, Abdullah NF, Nordin R (2021) Cross-layer design and performance analysis for ultra-reliable factory of the future based on 5G mobile networks. *IEEE Access* 9:68161–68175. <https://doi.org/10.1109/ACCESS.2021.3078165>
3. Salameh HAB, Bani Irshaid M, Al Ajlouni M, Aloqaily M (2021) Energy-efficient cross-layer spectrum sharing in CR green IoT networks. *IEEE Trans Green Commun Netw* 5(3):1091–1100. <https://doi.org/10.1109/TGCN.2021.3076695>
4. Guo H Wu R, Qi B, Liu Z (2021) Lifespan-balance-based energy-efficient routing for rechargeable wireless sensor networks. *IEEE Sens J* 21(24):28131–28142. <https://doi.org/10.1109/JSEN.2021.3124922>
5. K. Sangaiah et al. (2021) Energy-aware geographic routing for real-time workforce monitoring in industrial informatics. *IEEE Internet Things J* 8(12):9753–9762. <https://doi.org/10.1109/JIOT.2021.3056419>
6. Xu H, Huang L, Qiao C, Zhang Y, Sun Q (2012) Bandwidth-power aware cooperative multipath routing for wireless multimedia sensor networks. *IEEE Trans Wireless Commun* 11(4):1532–1543. <https://doi.org/10.1109/TWC.2012.020812.111265>
7. Khan A et al (2021) EH-IRSP: energy harvesting based intelligent relay selection protocol. *IEEE Access* 9:64189–64199. <https://doi.org/10.1109/ACCESS.2020.3044700>
8. Bolla DR, Shivashankar (2017) An efficient protocol for reducing channel interference and access delay in CRNs. In: 2017 2nd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT), pp 2247–2251. <https://doi.org/10.1109/RTEICT.2017.8257000>
9. Shankar S, Jijesh J, Bolla DR, Penna M, Sruthi PV, Gowthami A (2020) Early detection of flood monitoring and alerting system to save human lives. In: 2020 International conference on recent trends on electronics, information, communication & technology (RTEICT), pp 353–357. <https://doi.org/10.1109/RTEICT49044.2020.9315556>
10. Bolla DR, Jijesh JJ, Palle SS, Penna M, Keshavamurthy, Shivashankar (2020) An IoT based smart E-fuel stations using ESP-32. In: 2020 International conference on recent trends on electronics, information, communication & technology (RTEICT), pp 333–336. <https://doi.org/10.1109/RTEICT49044.2020.9315676>
11. Jambli MN, Shuhaimi WBWM, Lenando H, Abdullah J, Suhaili SM (2014) Performance evaluation of AODV in MASNETs: study on different simulators. *IEEE international symposium on robotics and manufacturing automation (ROMA)* 2014:131–135. <https://doi.org/10.1109/ROMA.2014.7295875>
12. Wazid M, Das AK, Kumar N, Alazab M (2021) Designing authenticated key management scheme in 6G-enabled network in a box deployed for industrial applications. *IEEE Trans Ind Inf* 17(10):7174–7184. <https://doi.org/10.1109/TII.2020.3020303>
13. Zhao J, Wang Y, Lu H, Li Z, Ma X (2021) Interference-based QoS and capacity analysis of VANETs for safety applications. *IEEE Trans Veh Technol* 70(3):2448–2464. <https://doi.org/10.1109/TVT.2021.3059740>

14. Jabbari A, Mohasefi JB (2022) A secure and LoRaWAN compatible user authentication protocol for critical applications in the IoT environment. *IEEE Trans Ind Inf* 18(1):56–65. <https://doi.org/10.1109/TII.2021.3075440>

Impact on Squeeze Film Lubrication on Long Cylinder and Infinite Plane Surface Subject to Magnetohydrodynamics and Couple Stress Lubrication



C. K. Sreekala , B. N. Hanumagowda , R. Padmavathi, J. Santhosh Kumar, and B. V. Dhananjayamurthy 

1 Introduction

Thrust bearings are a particular type of rotary bearings that permanently rotate between the parts and are intended to carry an axial load. They are commonly used in automotive (like in modern cars which uses helical gears) marine and aerospace applications. To improve the life and use of such equipment, the ideal performance of these moving parts is of greatest importance. The knowledge that the application of magnetic and electric fields enhances the load supporting capacity of the liquid metal bearings resulted in the development of magnetohydrodynamics lubrication. Liquid metals have highly conducting properties and have become an area of interest recently. Many magnetohydrodynamic lubrication problems have been analysed for a few years [1–3].

Earlier study of Newtonian fluid do not consider the size of fluid particles and is not an acceptable engineering approach for the analysis of fluids with microstructure additives. Hence, on the basis of proposed microcontinuum theories [4, 5] and Stoke's [6] microcontinuum theory, couple stress fluid model has been used by several authors to study hydrodynamic lubrication [7–11].

The combined effect of couple stresses and magnetohydrodynamics has been analysed by many authors. [12–14]. They all found that combined effect MHD and couple stress provides a promising increase in the bearing characteristics.

C. K. Sreekala (✉) · R. Padmavathi · B. V. Dhananjayamurthy
Department of Mathematics, Nitte Meenakshi Institute of Technology, Yelahanka, Karnataka, India
e-mail: sreekalarajeesh@gmail.com

B. N. Hanumagowda · J. Santhosh Kumar
Department of Mathematics, School of Applied Sciences, REVA University, Bengaluru, Karnataka, India

“In the application of squeezing film bearings, lubricated joints and injection moulding systems the squeeze flow between a cylinder and a plane is also important. The couple stress effects of squeeze film characteristics between a cylinder and a plane surface have been studied by Jaw Ren Lin. et.al,” [15]. A further study is motivated by this way as we have no idea how the collective effect of transverse magnetic fluid and couple stress fluid affects the cylinder plane system.

2 Mathematical Formulation

Figure 1 represents the geometry of a long cylinder of radius R advancing towards an infinite plane surface with a velocity $-V$.

“The basic governing equation like continuity and momentum equations based on the usual thin film assumptions is as follows.

$$\mu \frac{\partial^2 u}{\partial z^2} - \eta \frac{\partial^4 u}{\partial z^4} - \sigma B_0^2 u = \frac{\partial p}{\partial x} \tag{1}$$

$$\frac{\partial p}{\partial z} = 0 \tag{2}$$

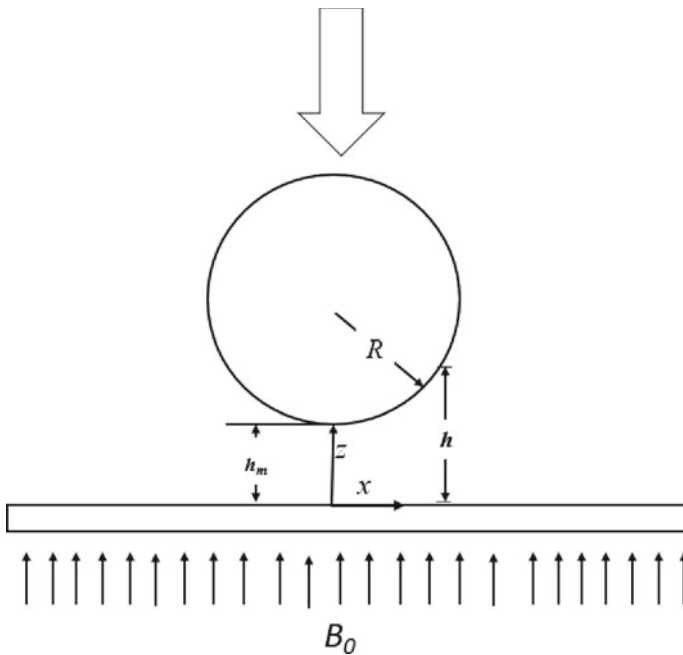


Fig. 1 Squeeze film geometry between a cylinder and a plane surface

$$\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0 \tag{3}$$

The boundary conditions are

(i) At the topmost surface, $z = h$

$$u = 0, \quad w = -V, \quad \frac{\partial^2 u}{\partial z^2} = 0 \tag{4}$$

(ii) At the bottom surface, $z = 0$

$$u = 0, \quad w = 0, \quad \frac{\partial^2 u}{\partial z^2} = 0 \tag{5}$$

Solving momentum expression (1) subject to the conditions (4) and (5) we get

$$u = \frac{1}{\sigma B_0^2} \frac{\partial p}{\partial x} \left\{ \left(\frac{A^2}{(A^2 - B^2)} \frac{\text{Cosh} \frac{B}{2l}(2z - h)}{\text{Cosh} \frac{Bh}{2l}} - \frac{B^2}{(A^2 - B^2)} \frac{\text{Cosh} \frac{A}{2l}(2z - h)}{\text{Cosh} \frac{Ah}{2l}} \right) - 1 \right\} \tag{6}$$

where [12]

$$A = \left\{ \frac{1 + (1 - 4\sigma B_0^2 l^2 / \mu)^{1/2}}{2} \right\}^{1/2}, \quad B = \left\{ \frac{1 - (1 - 4\sigma B_0^2 l^2 / \mu)^{1/2}}{2} \right\}^{1/2}$$

In the film region, $x \ll R$ and hence, film thickness is approximated to $h = h_m + \frac{x^2}{2R}$

Plugging expression (6) in to the continuity expression (2) and substituting the Eqs. (4) and (5), we derive the modified Reynolds equations in the form

$$\frac{\partial}{\partial x} \left\{ \frac{12h_{m0}^2}{M_0^2} \frac{\partial p}{\partial x} \left\{ \left(\frac{2lA^2}{B(A^2 - B^2)} \tan h \frac{Bh}{2l} - \frac{2lB^2}{A(A^2 - B^2)} \tan h \frac{Ah}{2l} \right) - h \right\} \right\} = 12V\mu \tag{7}$$

Introducing dimensions less parameters

$$x^* = \frac{x}{R}, \quad P^* = \frac{Ph_{m0}^2}{\mu RV}, \quad l^* = \frac{l}{h_{m0}}, \quad h_m^* = \frac{h_m}{h_{m0}}, \quad \beta = \frac{h_{m0}}{R}, \quad h^* = \frac{h}{h_{m0}} \tag{8}$$

The Reynolds expression takes the form

$$\frac{\partial}{\partial x^*} \left\{ g(h^*, l^*, M_0) \frac{\partial P^*}{\partial x^*} \right\} = \frac{12}{\beta} \tag{9}$$

where

$$g(h^*, l^*, M_0) = \frac{12l^*}{M_0^2} \left\{ \left(\frac{2A^{*2}}{B^*(A^{*2} - B^{*2})} \tan h \frac{B^*h^*}{2l^*} - \frac{2B^{*2}}{A^*(A^{*2} - B^{*2})} \tan h \frac{A^*h^*}{2l^*} \right) - \frac{h^*}{l^*} \right\} \tag{10}$$

The boundary condition for fluid pressure is $p^* = 0$ when $x^* = \pm 1$,

$$\text{And } \frac{dp^*}{dx^*} = 0 \text{ when } x^* = 0 \tag{11}$$

Integrating expression (9) and applying the boundary condition (11), we get the non-dimensional pressure

$$p^* = - \int_{x^*}^1 \frac{12x^*}{\beta g(h^*, l^*, M_0)} dx^* \tag{12}$$

The dimensionless load is

$$W^* = - \int_{x^*=-1}^{x^*=1} \int_{x^*}^1 \frac{12x^*}{\beta g(h^*, l^*, M_0)} dx^* dx^* \tag{13}$$

The non-dimensional time of approach is

$$T^* = - \int_{h_m^*}^1 \left\{ \int_{x^*=-1}^{x^*=1} \int_{x^*}^1 \frac{12x^*}{\beta g(h^*, l^*, M_0)} dx^* dx^* \right\} dh_m^* \tag{14}$$

3 Result and Discussion

3.1 Dimensionless Film Pressure (p^*)

The paper describes characteristics of the bearings due to squeeze film lubrication and by the collective impact of magnetohydrodynamics and a couple stresses on a cylinder plane system. The magnetohydrodynamics is described by Hartmann number M_0 , and couple stress parameter l^* characterizes couple stresses.

Figure 2 represents variation of p^* against x^* with $\beta = 0.04$, $l^* = 0.2$, $h_m^* = 0.8$. The dotted line characterizes the non-magnetic case ($M_0 = 0$), whereas the solid lines denote the magnetic case $M_0(2 \sim 6)$. It is seen in the graph that with the

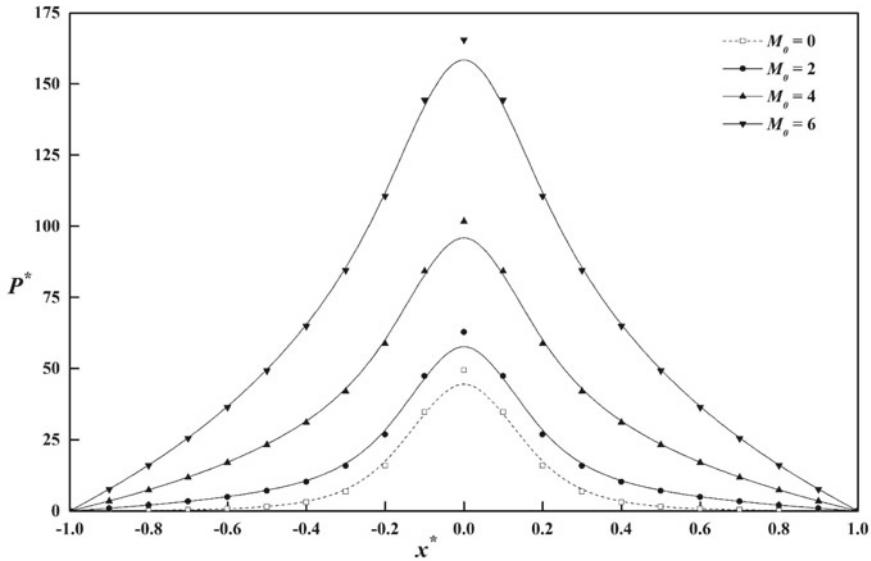


Fig. 2 Variation of non-dimensional pressure P^* with x^* for different values of M_0 with $l^* = 0.2, \beta = 0.04, h_m^* = 0.8$

increment in Hartmann number M_0 , the fluid film pressure enhances significantly. Figure 3 is plotted by varying the couple stress parameter l^* through (0 ~ 0.6) by keeping $M_0 = 2, \beta = 0.04, h_m^* = 0.8$ and is viewed clearly that film pressure grows significantly with the incrementing values of l^* . Compared with the Newtonian lubricant, couple stress fluids will provide more fluid film pressure. In Figure:4, the variation of fluid pressure is plotted by varying the minimum film thickness and fixing $l^* = 0.2, \beta = 0.04, M_0 = 2$. A decline in film pressure is observed for an increasing film thickness.

3.2 Non-Dimensional Load (W^*)

In Figure: 5, W^* (load) is plotted against h_m^* (min film thickness) for several values of Hartmann number M_0 by fixing $l^* = 0.2, \beta = 0.04, h_m^* = 0.8$. $M_0 = 0$ represents the non-magnetic case, whereas $M_0(2 \sim 6)$ represents the magnetic case. It is viewed that as the value M_0 increases, W^* also increases. Figure 6 illustrates the variation in W^* , for increasing values of the l^* . It is visible that as l^* increases W^* also increases.

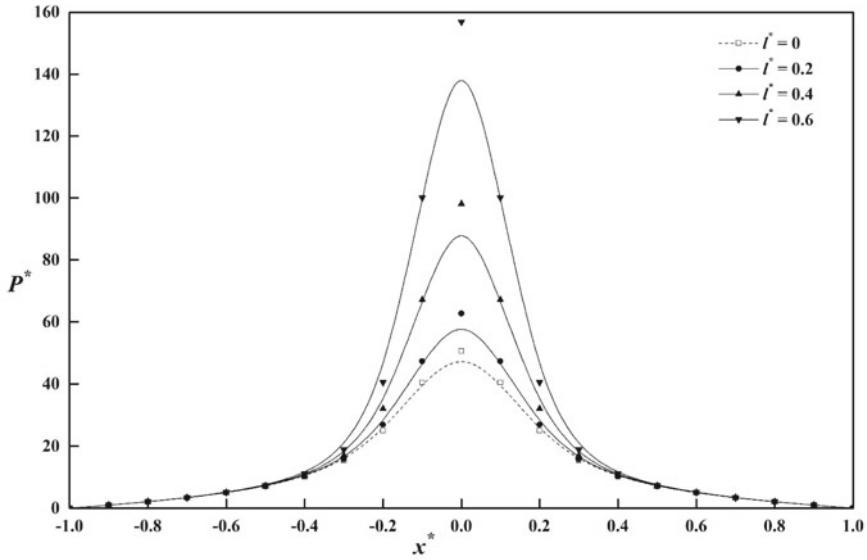


Fig. 3 Variation of non-dimensional pressure P^* with X^* for different values of l^* with $M_0 = 0.2$, $\beta = 0.04$, $h_m^* = 0.8$

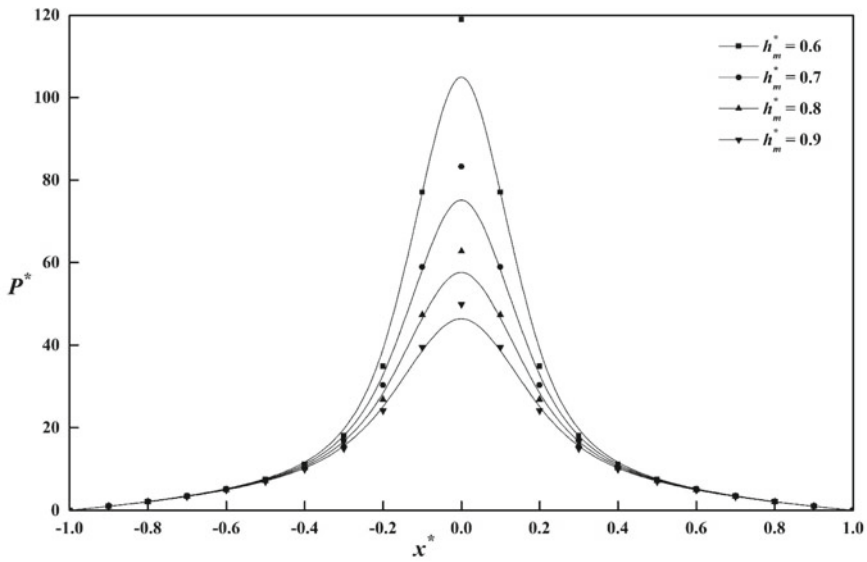


Fig. 4 Variation of non-dimensional pressure P^* with x^* for different values of h_m^* with $l^* = 0.2$, $\beta = 0.04$, $M_0 = 0.8$

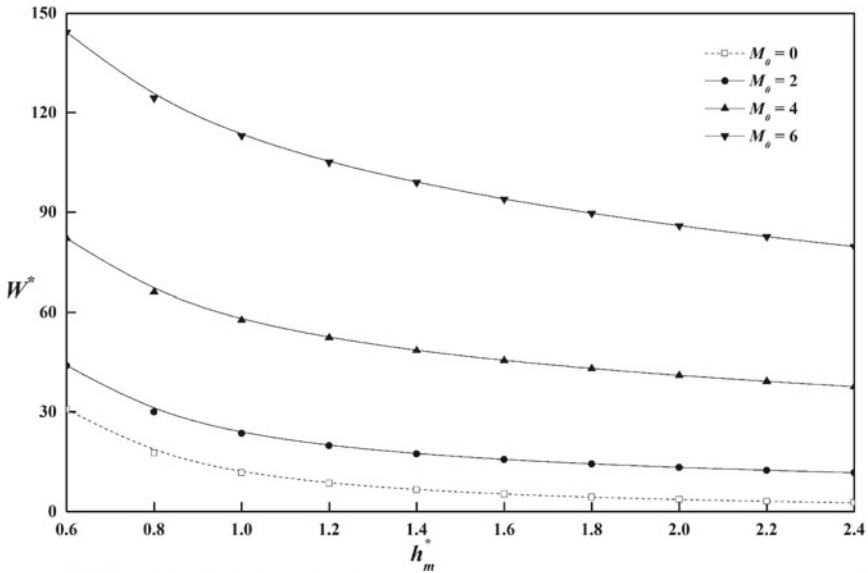


Fig. 5 Variation of non-dimensional load carrying capacity W^* with h_m^* for different values of M_0 with $l^* = 0.2$, $\beta = 0.04$

3.3 Squeezing Time

Figures 7 and 8 represent dimensionless min. film thickness h_m^* as a function of dimensionless squeezing time T^* . Figure 7 illustrates that the squeezing time is extended for increasing values of the magnetic field M_0 . Also in Fig. 8, an extended squeezing time is visible for increasing values of l^* .

4 Conclusions

It is acknowledged in this paper that

- The characteristics of bearing like: film pressure, squeezing time, and load supporting capacity enhance significantly in the magnetic case than the non-magnetic case.
- Comparing with the classical case, the characteristics of bearing improve significantly by greasing with couple stress fluid.

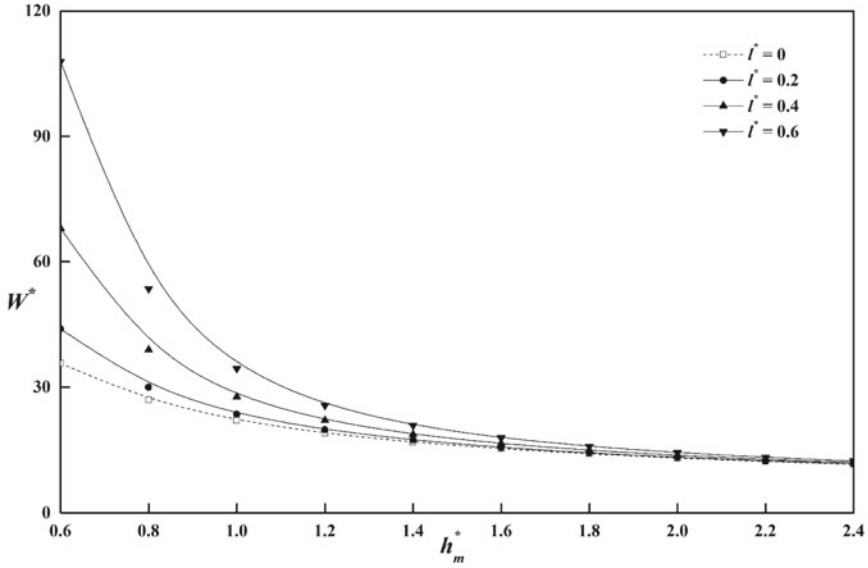


Fig. 6 Variation of non-dimensional load carrying capacity W^* with h_m^* for different values of l^* with $M_0 = 0.2$, $\beta = 0.04$

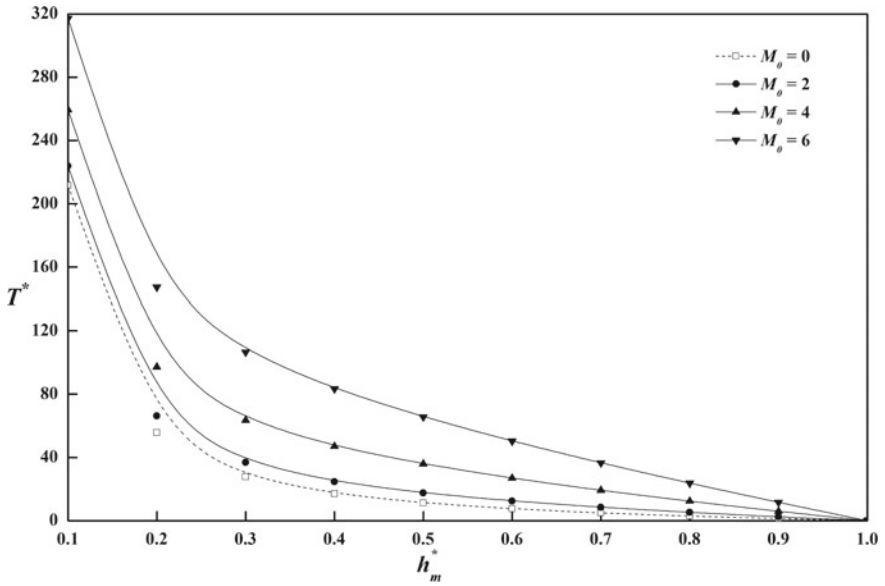


Fig. 7 Variation of non-dimensional load carrying capacity T with h_m^* for different values of M_0 with $l^* = 0.2$, $\beta = 0.04$

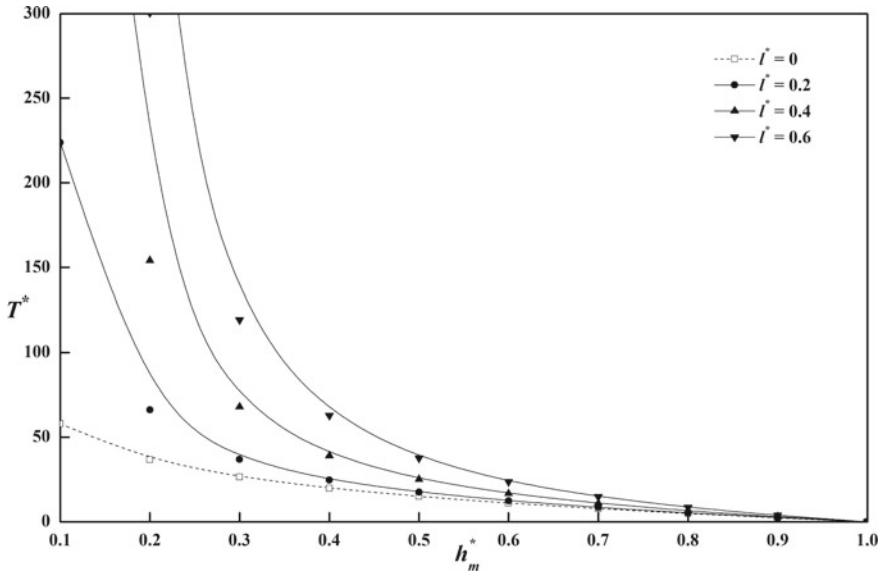


Fig. 8 Variation of non-dimensional load carrying capacity T^* with h_m^* for different values of l^* with $M_0 = 0.2$, $\beta = 0.04$

References

- Hughes WF, Elco RA (1962) MHD lubrication flow between parallel rotating disks. *J Fluid Mech* 13:21–32, View at: [Google Scholar](#)
- Kuzma DC, Maki ER, Donnelly RJ (1964) The MHD squeeze film. *J Fluid Mech* 19:395–400, View at: [Google Scholar](#)
- Hamza EA (1988) Magneto hydrodynamic squeeze film. *J Tribol* 110(2):375–377
- Ariman T, Turk MA, Sylvester ND (1973) Microcontinuum fluid mechanics—a review. *Int J Eng Sci* 11(8):905–930, View at: [Google Scholar](#)
- Ariman T, Turk MA, Sylvester ND (1974) Applications of microcontinuum fluid mechanics. *Int J Eng Sci* 12(4):273–293, View at: [Google Scholar](#)
- Stokes VK (1966) Couple stresses in fluids. *Phys Fluids* 9(9):1709–1715
- Lin JR (2000) Squeeze film characteristics between a sphere and a flat plate: couple stress fluid model. *Comput Struct* 75:73–80
- Bujurke NM, Jayaraman G (1982) The influence of couplestresses in squeeze films. *Int J Mech Sci* 24:369–376
- Ramanaiha G (1979) Squeeze films between finite plates lubricated by fluids with couplestress. *Wear* 54:315–320
- Sinha P, Singh C (1981) Couplestresses in the lubrication of rolling contact bearings considering cavitation. *Wear* 67:85–91
- Naduvanamani NB, Hiremath PS, Gurubasavaraj G (2005) Effect of surface roughness on the couple-stress squeeze film between a sphere and a flat plate. *Tribology Int* 3
- Naduvanamani NB, Fathima ST, Hanumagauda BN (2011) Magneto-hydrodynamic couple stress squeeze film lubrication of circular stepped plates. *Proc Mech Eng Part I J Eng Tribology* 225:1–9
- Naduvanamani NB, Rajashekar M (2011) MHD Couplestress squeeze-film characteristics between a sphere and a plane surface tribology—materials. *Surf Interfaces* 5:94–99

14. Naganagowda HB (2016) Effect of magnetohydrodynamics and couple stress on steady and dynamic characteristics of plane slider bearing. *Tribology Online* 11(1):40–49
15. Lin J-R, Liao W-H, Hung C-R (2004) The effects of couple stresses in the squeeze film characteristics between a cylinder and a plane surface. *J Mar Sci Technol* 12(2):119–123

Collapse Detection Using Fusion of Sensor



Sushmita A. Pattar, A. C. Ramachandra , N. Rajesh, and C. R. Prashanth

1 Introduction

This document constitutes Springer's guidelines for the preparation of proceedings papers. These may be stand-alone proceedings or part of a series. Here is a list of some of our main proceedings series:

A fall is defined as an unexpected and accidental change in the body position from higher to lower. Fall detection systems have been presented [1] to reduce more health problems. Such unexpected falls are dangerous to the people who are elder and ill, and it may sometime lead to death in the aged people or ill people or both. Automatic fall detection can decrease the risk by detecting falls of the human being. There are lots of research done in 3 different types of fall detection approaches, viz., vision-based, ambient-based and wearable-based approaches. Vision-based approach depends on the detection of posture, inactivity and so on. An ambient-based approach deals with audio, video and vibration information. Both these approaches are highly accurate but are expensive. Wearable approach generally depends on the accelerometer which is low cost and occupies less space compared to vision and ambient device and to use both low cost and less space with high accuracy. We have proposed a new method by fusion with the heartbeat sensor with an accelerometer.

The wearable detection system uses a single sensor which has low cost and less space, and it will give less accuracy and gives the fake alarm, by fusing two or more sensors compared to the vision and ambient based detection approach. To achieve

S. A. Pattar
L and T Technology Services, Bangalore, India

A. C. Ramachandra (✉) · N. Rajesh
Nitte Meenakshi Institute of Technology, Bangalore 560064, India
e-mail: ramachandra.ac@nmit.ac.in

C. R. Prashanth
Dr Ambedkar Institute of Technology, Bangalore 560056, India

a higher level of accuracy, we [2, 3] have selected the heart rate sensor by using a multi-dimensional fusion of physiological and kinematic parameters. Even the heart rate sensor has helped in the form of cost and space comparing other physiological sensors which are used in hospitals.

The source of the data is classified in to 3 items: fall, non-fall and nearly fall with the different variation.

2 Block Diagram

From Fig. 1. block diagram, the dataset is taken from the GitHub website, and both heartrate and accelerometer data are taken separately. From the heart rate sensor and accelerometer, data are combined, creating datasheet which is input to train dataset. These train data are raw; there is no proper alliance and method and has some missing [4, 5] value, zero values and other. In order to get proper output, it should pre-processes the data, and it is done in the pre-processing input dataset block. After this, it should select the better algorithm which fits the model in terms of high accuracy and efficiency and low mean square error. In the machine learning, there are many different algorithms, and select the algorithm which will fit the model properly with minimum error, high efficiency and high accuracy and train the model. From the below block are the heart rate sensor and accelerometer data from the real word and should be given to the train model. In the block of train model, the data are taken from the test data from the pre-processing block and real-time block and find out the person is fall or non-fall.

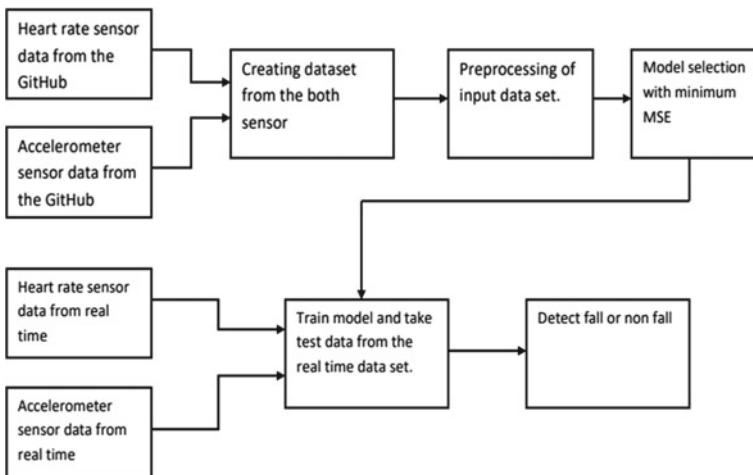


Fig. 1 Block diagram of the model

Fig. 2 Flowchart of the model

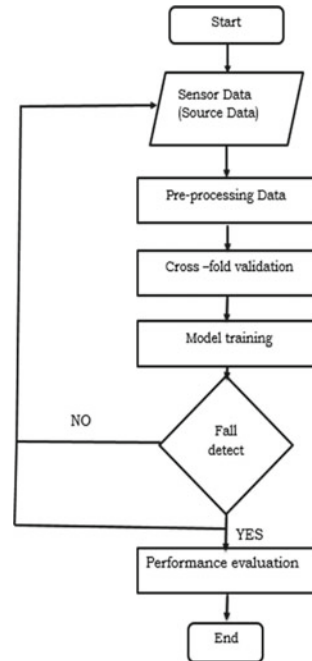


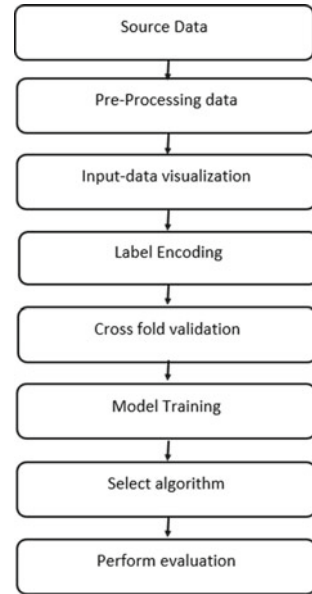
Figure 2 tells about the basic idea of working of the model (fall detector). From the first block start, the input data are taken from the source data which is from the website called GitHub. These data are raw data and pre-process the data by adding the missing value, removing the null value in the row, etc. After the pre-processing of the input data, there is overfitting in the given input from train dataset to test dataset; in order to remove these overfittings, we should do k -cross-fold validation. Find the best algorithm for the model in order to minimize the error and increase efficacy. Then use if condition; if the condition is true, then it is fall, or if it is false, then it is non-fall, wherever the output is again sent to the dataset for not to repeat next iteration. By detecting the fall and non-fall, calculate the performance evaluation by calculating the sensitivity, specificity, accuracy, recall, precision and $F1$ -score and end of the algorithm (Fig. 3).

3 Proposed Model

3.1 Source Data

The fall data contain different activities of 21 people of different gender and age. Dataset was collected from GitHub. The participants are both men and women. It is

Fig. 3 Implementation flow for fall detection



collected from an accelerometer, heart rate sensor and gyroscope sensor and divided into 3 types (6=fall, 9 non-falls and 4=near to fall). The dataset has 12 parameters that are collected from the participant w , x , y and z which are quaternions of the gyroscope. A_x , A_y and A_z are the axis of the accelerometer signal (g). Droll, ditch and dyawn are the angular velocities of the gyroscope and heart PPG sensor (Fig. 4).

3.2 Pre-processing Data

The pre-processing of the input data is used to check null values and categorical attributes, if there are any null values and different categorical values present in the given input data. For the categorical attributes, create a list for a categorical column [6] with the empty list with the help of for loop and divide object data types from the given data. Some of the categorical columns which are not necessary for the model just remove from the dataset by using drop instruction if any value is present in the given input. To find the null value in a particular column with the help of Boolean value and to fill the more accurate value with the help the pivot table (Figs. 5 a and b).

sl.no	w	x	y	z	ax	ay	az	droll	ditch	dyan	heart	time	Unnamed: 13	Unnamed: 14	Unnamed: 15	Unnamed: 16	Unnamed: 17	action	
0	11	-0.8331	0.4897	0.1680	-0.1942	0.018	0.012	0.002	-0.4	0.1	0.8	1023.0	2018.0	5	11.0	13.0	17	2.847	2
1	12	-0.8331	0.4897	0.1679	-0.1943	0.016	0.013	0.002	-0.4	0.2	0.9	1023.0	2018.0	5	11.0	13.0	17	2.885	2
2	13	-0.8332	0.4895	0.1681	-0.1943	0.015	0.012	0.008	-0.5	0.2	1.1	1023.0	2018.0	5	11.0	13.0	17	2.905	2
3	14	-0.8332	0.4895	0.1680	-0.1944	0.017	0.013	0.007	-0.6	0.1	0.9	1023.0	2018.0	5	11.0	13.0	17	2.909	2
4	15	-0.8332	0.4895	0.1680	-0.1944	0.015	0.014	0.003	-0.9	0.0	1.0	0.0	2018.0	5	11.0	13.0	17	2.925	2

```
train.tail()
```

sl.no	w	x	y	z	ax	ay	az	droll	ditch	dyan	heart	time	Unnamed: 13	Unnamed: 14	Unnamed: 15	Unnamed: 16	Unnamed: 17	action	
17305	117308	-0.7138	0.4122	0.4545	-0.3373	0.314	0.030	0.111	-49.3	-159.8	89.4	0.0	2018.0	5	11.0	13.0	15	18.847	
17306	117307	-0.7280	0.4197	0.4430	-0.3112	0.317	-0.084	0.106	-38.3	-159.3	87.2	0.0	2018.0	5	11.0	13.0	15	18.888	
17307	117308	-0.7424	0.4252	0.4312	-0.2882	0.335	0.023	0.068	-26.0	-159.8	89.0	0.0	2018.0	5	11.0	13.0	15	18.907	
17308	117309	-0.7561	0.4301	0.4181	-0.2612	0.366	0.047	-0.014	-31.6	-159.2	82.8	0.0	2018.0	5	11.0	13.0	15	18.911	
17309	117310	-0.7678	0.4365	0.4051	-0.2360	0.372	-0.046	-0.131	-46.2	-153.7	75.7	0.0	2018.0	5	11.0	13.0	15	18.926	

Fig. 4 Input data details

3.3 Input Data Visualization

Input data visualization is done with the package called Seaborn. It is a Python data visualization library built on top of matplotlib and provides [7] a high-level interface for informative statistical graphics. Figure 6a shows that particular column is taken as input. Bar grasp shows particular number of value and used preprocessing input, and curve graphs represent normalization of particular column value. In Fig. 6b, countplot graph of the action is taken as separate attribute where x axis represents different type of action present in dataset and y axis represents number of actions present in the input dataset.

3.4 Label Encoding

Label encoding is assigned a different value of object dataset for better performance. For the label encoding, import model from sklearn import and the column action is converted into numerical value based on the specific action. Without label ending, prediction may effect to get a better solution for the given model to convert to categorical attribute (action) to label encoder for model improvement, and model can process as numerical value.

Fig. 5 **a** Different type of datatype present in the dataset, **b** number of null value in the given dataset

```

a <class 'pandas.core.frame.DataFrame'>
RangeIndex: 21210 entries, 0 to 21209
Data columns (total 19 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   sl.no       21210 non-null   object
1   w           21210 non-null   float64
2   x           21210 non-null   float64
3   y           21210 non-null   float64
4   z           21210 non-null   float64
5   ax          21210 non-null   float64
6   ay          21210 non-null   float64
7   az          21210 non-null   float64
8   droll       21210 non-null   float64
9   ditch       21210 non-null   float64
10  dyan        21210 non-null   float64
11  heart       21210 non-null   float64
12  time        21210 non-null   float64
13  Unnamed: 13 21210 non-null   object
14  Unnamed: 14 21210 non-null   float64
15  Unnamed: 15 21210 non-null   float64
16  Unnamed: 16 21210 non-null   object
17  Unnamed: 17 21210 non-null   float64
18  action      17310 non-null   float64
dtypes: float64(16), object(3)
memory usage: 3.1+ MB

b In [18]: df.isnull().sum()

Out[18]: sl.no      0
         w          0
         x          0
         y          0
         z          0
         ax         0
         ay         0
         az         0
         droll      0
         ditch      0
         dyan       0
         heart      0
         time       0
         Unnamed: 13 0
         Unnamed: 14 0
         Unnamed: 15 0
         Unnamed: 16 0
         Unnamed: 17 0
         action     3900
         dtype: int64
    
```

3.5 Correlation Matrix

Correlation matrix will measure the degree of relatedness of variables. For a pair of data, correlation analysis can lead to a numerical value that represents the degree of relatedness for the pair of variables. The correlation coefficient is denoted as r . as given in figure from blue to orange color which shows the relativeness of the variable. If r value is near to 1, variables are directly responsible; if r value is near to -1 , variables are inversely proportional; and if r is 0, there is no relation between variables (Fig. 7).

Fig. 6 **a** Data value and normalization of value of graph, **b** different types of action graph

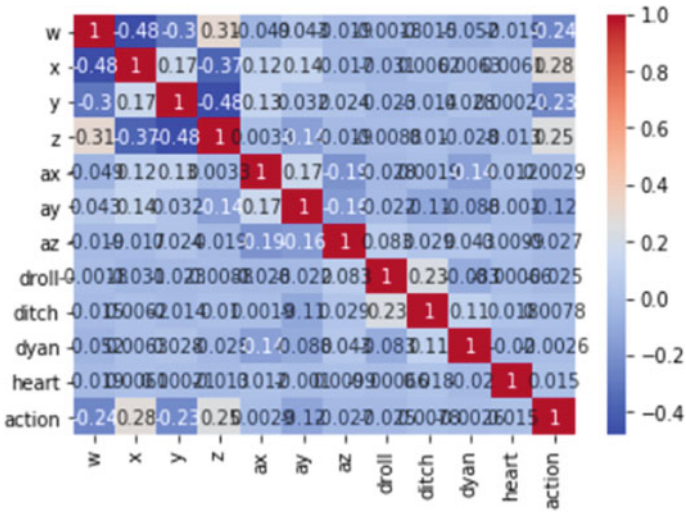
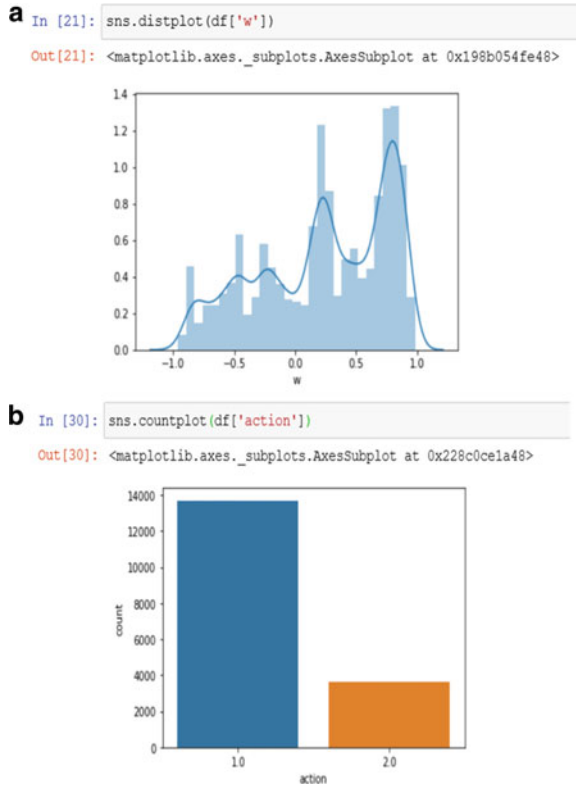


Fig. 7 Correlation matrix

3.6 Cross-Fold Validation

Cross-validation is used to remove overfitting of the given data. The input dataset splits the train and test data for cross-validation. In case given train data which does not aware of pattern of test data, in real word, the accuracy will get reduced. To get more accuracy, we are using k fold cross-validation. For example, take 21210 samples in 10 folds where $K=10$. For 10 iterations, take 1/10 of the data as test dataset. Remaining data are taken as train dataset. Second iteration takes second set of 1/10 data as test data and reaming train data, and repeat these processes up to $k(10)$ iteration. Take average of the score of 10 iteration score.

3.7 Model Training

In machine learning, there are different types of algorithms for different models. Every model has its own advantages and disadvantages. To select the algorithm which suits the best for the model, we have to calculate mean square error and [8–11] cross-validation for every algorithm. The value of mean square error is less means that the proposed model is near the fitting line, and cross-validation of the model is used to remove overfitting of the data.

From the above Fig. 8, the graph from the linear regression model is shown. Its statistical model attempts to show the relation between input variable (x) and output variable (y) with the linear equation. It is taken from sklearn linear model. Mean square error of linear regression is 0.1132.

Ridge regression is based on the simple principle of linear regression and also comes under the category of regularization ($L2$ regression). It is used when several attributes are more to visualization. Figure 9 is the graph of ridge regression where it is taken from the sklearn linear model. Mean square value of the ridge regressor algorithm is 0.12911.

Least absolute shrinkage and selection operator (Lasso) regression. Regularization parameter is multiplied by summation of absolute value. Figure 10 is the graph of Lasso regression where it is taken from the sklearn linear model. Mean square value of the Lasso regressor algorithm is 0.1532.

The decision tree is a graphical representation of all the possible solutions to a decision based on the certain condition. It starts with the root and then branches off to a number of decisions and the condition of tree. It will begin with adding a root node for the tree, all the nodes receive the list of rows input, and root will receive the entire training set, then each node asks for the true and false question about one other feature, and in response to the question, it will split to partition the dataset into two different subsets; these subsets become input to the child node. Figure 11 is the graph of decision tree regression where it is taken from the sklearn linear model. Mean square value of the lasso regressor algorithm is 0.0.

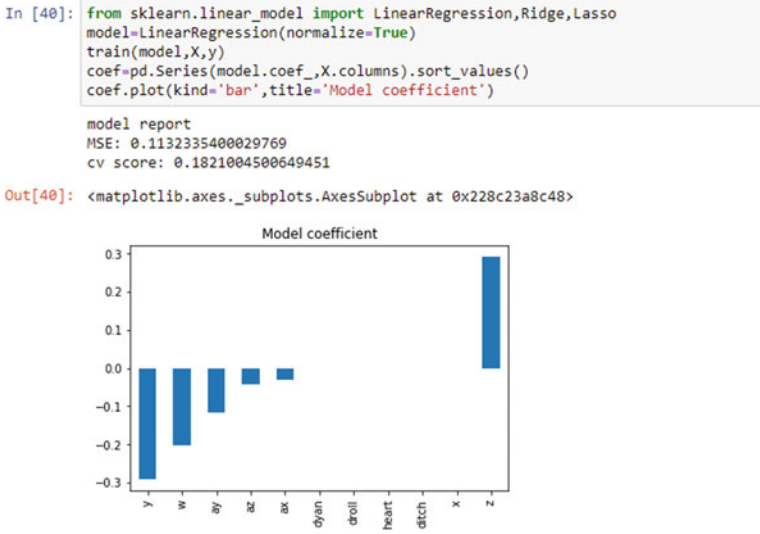


Fig. 8 Graph of linear regression model

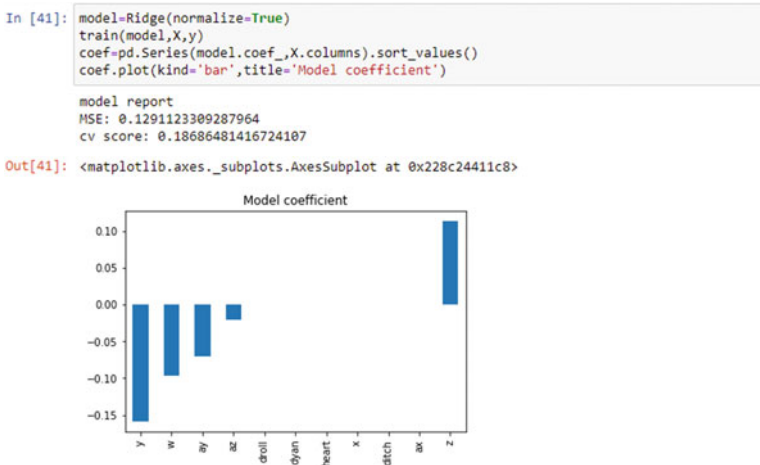


Fig. 9 Graph of ridge regression model

Random forests are made out of decision trees. The disadvantage of the decision tree does not perform well on the real dataset; it tends to overfit, meaning it will perform well in the training dataset but not on the test dataset. The decision tree has high variance and low bias (datasets that are not used in training data). To overcome this problem, using decision trees in a different form is called random forest. Random forest uses the ensemble learning method in which the prediction is based on the combined results of various individual models. It will take the entire dataset and

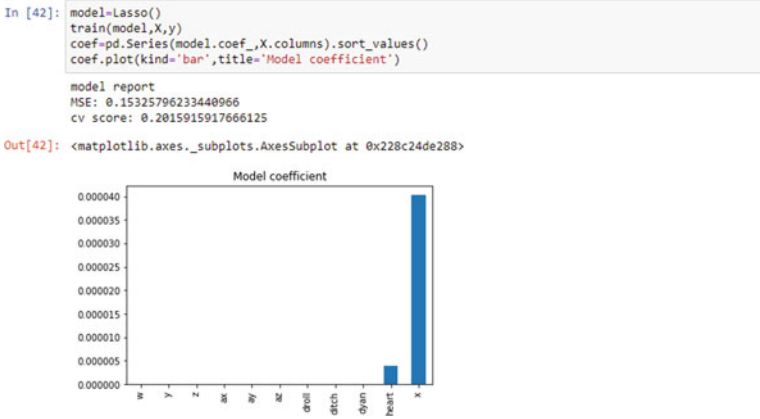


Fig. 10 Graph of Lasso regression model

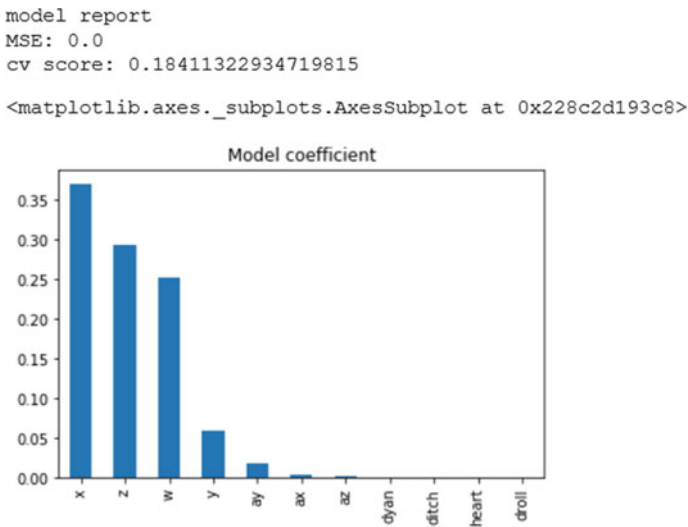


Fig. 11 Graph of decision tree regression model

crest subset of the data. The size of the data remains the same, and all the subsets are equal to a number of rows means taking a subset on the random base with replacement. Figure 12 is the graph of random forest regression where it is taken from the sklearn linear model. Mean square value of the lasso regressor algorithm is 0.00013 (Table 1).

From the above table, there are 5 different algorithms and mean square error value of algorithm. Linear regression is the simplest algorithm, it works on basis of linear equation, its MSE is more, and accuracy of the model becomes low. Ridge algorithm is used for multidirectional data for model selection, its MSE is more compared

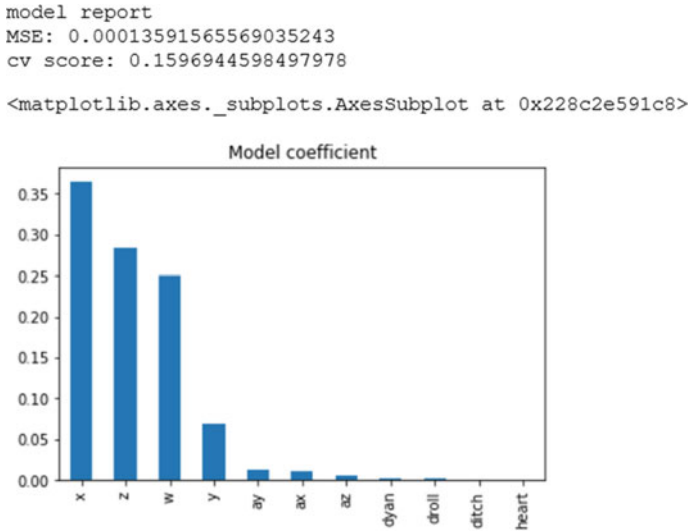


Fig. 12 Graph of random forest regression model

Table 1 Different algorithm, MSE, and CV score

Algorithm	MSE	CV score
Linear regression	0.1132	0.1821
Ridge	0.1291	0.1868
Lasso	0.1532	0.2015
Decision tree regressor	0.0	0.1841
Random forest regressor	0.0001359	0.1596

to linear regression, and accuracy becomes less with time management. Lasso is another algorithm which is used to embed system shrinkage method that performs both variable selection and regularization at the same time, and it has more MSE and high accuracy. Decision tree regression has MSE very low compared to other algorithms and has very high accuracy. Random forest regressor has small MSE, but random forests are made out of decision trees. The disadvantage is the decision tree does not perform well on the real dataset; it tends to overfit, meaning it will perform well in the training dataset but not on the test dataset. The decision tree has high variance and low bias. To overcome this problem, using decision trees in a different form is called random forest.

Table 2 Parameter used in performance evaluation and value

Parameter	Value
True positive	3320
True negative	128
False positive	14
False negative	0
Sensitivity	1.0
Specificity	0.901408
Accuracy	0.995956
Recall	1.0
Precision	0.99580
F1-score	0.997696

3.8 Evolution

The performance evaluation is calculated by the different parameters, after performing the confusion matrix for the given training dataset. The rows correspond to the prediction, and columns correspond to actual values, which contains true positive (t_p), true negative (t_n), false negative (f_n), false positive (f_p) (Table 2).

- Sensitivity tells what percentage of participants who have fallen were correctly identified.

$$\text{Sensitivity} = \frac{\text{true negative}}{\text{true positive} + \text{false negative}}$$

- Specificity tells that what percentage of patients without fall well correctly identified.

$$\text{Specificity} = \frac{\text{true negative}}{\text{true negative} + \text{false positive}}$$

- Accuracy is calculated as the total number of correct prediction (true positive and true negative) divided by total number of dataset.

$$\text{Accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{true positive} + \text{true negative} + \text{false positive} + \text{false negative}}$$

- Recall is calculated as the percentage of actual negative results out of all predicted negative values from the model.

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

- Precision is calculated as the percentage of actual positive results out of all predicted positive values from the model.

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

- *F1*-score is calculated as a balance between precision and recall and takes into account both of these values. And the model is accurately predicting both.

$$f1 - \text{score} = \frac{2 * \text{recall} * \text{precision}}{\text{recall} + \text{precision}}$$

4 Result

The proposed model is used to predict the fall and non-fall which is helpful for people who are sudden fall without any notice. Prediction is done in real time and already built-in data.

From Fig. 13 above, the prediction of the value is taken as already built-in test dataset, and the prediction value is taken as the separate data file to compare actual test data and predicted value data.

From Fig. 14, the prediction value is detected from real-time test data. To create real-time test data, data values are manually entered, and it is predicted action.

```

Out[45]:


|       | w      | x    | y       | z       | ax     | ay    | az    | droll | ditch | dyan  | heart |
|-------|--------|------|---------|---------|--------|-------|-------|-------|-------|-------|-------|
| 17310 | 0.7522 | 3526 | -0.5188 | -0.2861 | -0.246 | 0.245 | 0.429 | 59.3  | 25.3  | -57.1 | 0.0   |
| 17311 | 0.7562 | 3538 | -0.5225 | -0.2704 | -0.327 | 0.347 | 0.445 | 72.6  | 28.6  | -65.6 | 0.0   |
| 17312 | 0.7602 | 3547 | -0.5261 | -0.2532 | -0.399 | 0.352 | 0.482 | 68.4  | 29.0  | -73.2 | 0.0   |
| 17313 | 0.7639 | 3548 | -0.5287 | -0.2362 | -0.405 | 0.387 | 0.516 | 52.9  | 25.9  | -75.5 | 0.0   |
| 17314 | 0.7671 | 3539 | -0.5301 | -0.2206 | -0.405 | 0.314 | 0.479 | 48.6  | 22.5  | -73.6 | 0.0   |



In [46]: from sklearn.ensemble import RandomForestRegressor
model=RandomForestRegressor()
model.fit(X,y)

Out[46]: RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
max_depth=None, max_features='auto', max_leaf_nodes=None,
max_samples=None, min_impurity_decrease=0.0,
min_impurity_split=None, min_samples_leaf=1,
min_samples_split=2, min_weight_fraction_leaf=0.0,
n_estimators=100, n_jobs=None, oob_score=False,
random_state=None, verbose=0, warm_start=False)

In [47]: pred=model.predict(x_test)
pred

Out[47]: array([1. , 1. , 1. , ..., 1.98, 1.98, 1.98])

```

Fig. 13 Result from built-in dataset

```

In [51]: x_test.iloc[0]
Out[51]: w      0.7522
         x    3526.0000
         y     -0.5188
         z     -0.2861
         ax    -0.2460
         ay     0.2450
         az     0.4200
         droll  59.3000
         ditch  25.3000
         dyan  -57.1000
         heart   0.0000
         Name: 17310, dtype: float64

In [52]: model.predict([x_test.iloc[0]])
Out[52]: array([1.])

In [53]: model.predict([[-0.8332, 7954, 0.1681, -0.1943, 0.015, 0.012, 0.008, -0.5, 0.2, 1.1, 1023.0]])
Out[53]: array([2.])

```

Fig. 14 Result from real-time dataset

5 Conclusion

We have discussed the fall detection with the fusion of heart rate sensor and accelerometer which act as both accelerometer and gyroscope with the help of machine learning algorithm. Then, machine learning algorithm will fit the model according to the dataset and model. Our results are compared with the existing method from the one of the literature survey, i.e. cluster based on fall detection which detects the fall only using the unsupervised cluster algorithm. The accuracy and efficiency are less. In order to increase the accuracy, sensitivity, and efficiency, we used same dataset but with the help of different machine learning algorithms. There are two algorithms which give less mean square error and high efficiency; they are decision tree regressor and random forest regressor. The decision tree regressor has very least MSE, but it is less accurate for real-time data. In order to avoid this disadvantage, we have used random forest regressor, and it gave less MSE and high accuracy and $F1$ -score.

References

1. Khojasteh SB, Villar JR, Chira C, González VM, de la Cal E (2018) Improving fall detection using an on-wrist wearable accelerometer. Sensors (Basel, Switzerland)
2. Lee M-S, Lim J-G, Park K-R, Kwon D-S (2017) Unsupervised clustering for abnormality detection based on the tri-axial accelerometer. ICCAS-SICE, 2017
3. Lee J-S, Tseng H-H (2019) Development of an enhanced threshold-based fall detection system using smartphones with built-in accelerometers. IEEE Sens J
4. Bourke AK, Van de Ven PW, Chaya AE, O'Laughlin GM, Nelson J (2018) Testing of a long-term fall detection system incorporated into a custom vest for the elderly. Eng Med Biol Soc
5. Youngkong P, Panpanyatep W (2021) A novel double pressure sensors-based monitoring and alarming system for fall detection. In: Second international symposium on instrumentation, control, artificial intelligence, and robotics, 2021 .
6. Chen Y, Du R, Luo K, Xiao Y (2021) Fall detection system based on real-time pose estimation and SVM. In: IEEE 2nd international conference on big data, artificial intelligence and internet of things engineering, 2021

7. Martínez-Villaseñor L, Ponce H, Brieva J, Moya-Albor E, NúñezMartínez J, Peñafort-Asturiano C (2019) Up-fall detection dataset: a multimodal approach. *Sensors*
8. Gupta A, Srivastava R, Gupta H, Kumar B (2020) IoT based fall detection monitoring and alarm system for elderly. In: IEEE 7th Uttar Pradesh section international conference on electrical, electronics and computer engineering, 2020
9. Aziz O, Musngi M, Park EJ, Mori G, Robinovitch SN (2017) A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials. *Med Bio Eng Comput*
10. Zhao S, Li W, Cao J (2018) A user-adaptive algorithm for activity recognition based on k-means clustering, local outlier factor, and multivariate gaussian distribution
11. Fakhruddin SS, Gharghan SK (2019) An autonomous wireless health monitoring system based on heartbeat and accelerometer sensors. *J Sens Actuator*

Secured Storage of Information Using Audio Steganography



M. R. Sowmya, K. N. Shreenath, Saritha Shetty, Savitha Shetty, Salman Wajid, and Yashas Kantharaj

1 Introduction

Electronic correspondence is the soul of numerous associations. A significant part of the data conveyed consistently should be kept confidential. Data, for example, financial reports, worker information, and clinical records should be conveyed in a manner that guarantees confidentiality and trustworthiness.

The present enormous interest in Web applications expects information to be sent in a protected way. Information transmission in open correspondence framework isn't secure as a result of capture attempts and ill-advised control by snoop. Along these lines, the appealing answer for this issue is steganography, which is the workmanship and study of composing covered-up messages so that nobody, aside from the sender

M. R. Sowmya (✉) · S. Wajid · Y. Kantharaj
Department of Computer Science and Engineering, NMIT, Yelahanka, Bangalore, India
e-mail: sowmya.mr@nmit.ac.in

S. Wajid
e-mail: 1nt17cs162.salman@nmit.ac.in

Y. Kantharaj
e-mail: 1nt17cs212.yashas@nmit.ac.in

K. N. Shreenath
Department of Computer Science and Engineering, Siddaganga Institute of Technology,
Tumakuru, India
e-mail: shreenathk_n@sit.ac.in

S. Shetty
Department of MCA, NMAM Institute of Technology, Karkala, India
e-mail: shettysaritha1@nitte.edu.in

S. Shetty
Department of CSE, NMAM Institute of Technology, Karkala, India
e-mail: shettysavi1@nitte.edu.in

and mean beneficiary, associate the presence with the message, a type of safety through indefinite quality. Sound steganography is an idea of hiding the information by concealing it into another medium, for example, sound record. In this paper, we principally talk about various kinds of sound steganography strategies, benefits, and hindrances.

Sound steganography deals with a strategy to cover a strange message in a sound record. Moreover, audio steganography can be used for secret watermarking or covering ownership or copyright information in the sound that can be affirmed later to legitimize ownership rights.

Audio steganography encrypts the very presence of a message so that if successful it generally attracts no suspicion at all. Using audio steganography, information can be encrypted in audio files of mp3 or wav format and thus stored or transmitted without grabbing the attention of any third party who is looking for any sensitive data. Thus, ensuring data privacy by taking advantage of loopholes in the human sensory system.

The main objective of this paper is to utilize the technique of audio steganography to embed any kind of information like a txt, pdf, jpeg, or png file into a cover audio file which results in stego-object of mp3/wav format, depending on the type of information embedded into it. The user is given a key to extract the information from the stego-object.

2 Literature Survey

Hmood [3] say since safe information transfers are growing day by day, steganography has become very relevant, and new techniques have been used. Steganography is a technique that the information needed is hidden in all other information so that the second information does not alter substantially and continues to be the same as the initial information. This thesis is proposing a new approach to concealing encrypted smartphone image in an audio file.

Jian et al. [6] say the application has been successfully developed and reached its projected objectives, this application that allows user to embed and extract their message in wav audio, which will offer confidentiality in the communication. However, one constraint for this application is that the applying supports document to be embedded in wav audio. Therefore, additional file formats like video and documents may be added to the applying as an activity medium. This mechanism can be further worked upon by detailed research on file formats.

Chandrakar et al. [7] say the application of the steganographic rule has been designed accentuation sweetening in capability and security of message transmission. This user-friendly application maintains the three major aspects of the user's message privacy upon covertly hiding the message. Messages are randomly mixed before hiding which adds another advantage in terms of security. Mistreatment of the least significant bits [LSBs] algorithm adds to the advantage of operating with any audio

file format. Audio quality is kept intact without any deterioration even in cases where two or more audio files are used, to reduce the suspicion of any secret communication.

Timothy et al. [8] say there is a variety of proven ways for applying steganography to cover info at intervals of audio data. During this analysis work, an associate in nursing audio steganography system for MP3 and MP4 that uses mistreatment separate circular function rework (DCT) and unfold spectrum techniques were developed. It completely was shown through implementation and subjective experimentation that the developed audio steganography system supports MP3 and MP4 digital audio format. The system developed has the flexibility to enter a secret message of a size that is up to 500 kb; however, the system has the power to infix a text size of 250 kb concerning the digital audio length or size with no distortion and can retain a similar size once embedding text into it. The work has been able to develop a sturdy hand system that will help secure and share a great amount of sensitive information or info without arousing suspicion. This method is so suggested for security agencies and different organizations that think about data security as being of uttermost priority. This method could help send covert battlefield information via an innocuous cover audio signal.

Indrayani et al. [9] say steganography on the WAV format audio, using multiple levels of LSB manipulation ranging from (LSB + 1 to LSB + 6), has a successful result. The higher the LSB manipulation, the larger the secret data can be embedded, but the stego-object resulted will have high noise. PSNR and spectrograms are used to compare various sizes of audio files to measure the noise level.

3 System Requirements Specifications

3.1 Functional Requirements

User(s) Registration

The user has to register if not registered already

Input: User name, distinctive Id, face data

Output: Successful registration message is displayed.

System Login

Using credentials login to system.

Wrong use of credentials is handled by error handling.

Input: User credentials.

Output: Successful message for valid credentials.

Using Audio file to embed Secret file.

The user has to select the file of any format which has to be hidden.

A smaller size cover audio file is chosen by the user.

A relatively larger size larger audio file is chosen by the user again.

The secret file is embedded with the smaller audio file first and again embedded in the larger (second) audio file.

A key for recovery and a stego-object with an embedded file in it is returned to the user.

Input: Secret file, smaller size cover audio file, relatively larger size larger audio file.

Output: Key for recovery, secret file in a stego-object

From the audio file extracting the secret file

Input: Stego-object, key for recovery

Output: Secret file

3.2 *Quality of Service Requirements*

Security: Data used are sensitive; it should be secured.

- **Maintainability:** The system must be maintained as technology progresses. This is possible because the code is generic and contains comments.
- **Portability:** The application is platform independent.
- **Robust:** The system must be able to withstand failures.
- **Modularity:** As the system is built in a modular fashion, any new procedure can be added.
- **Cost-efficient:** The system must be cost-effective.

3.3 *User Requirements*

Interactive system: User-friendly system.

4 Design

Two audio files are used to embed any kind of secret file. Any type of secret file can be hidden in the cover audio file. Once the secret file is embedded, the recovery key and stego-object which has the secret file is returned to the user. This stego-object can be saved in a hard disk or over the cloud. This can also be used in communication between two individuals. In order to extract the secret file, recovery key and stego-object are required. Once the file is extracted, it can be returned with no damage to the file (Fig. 1).

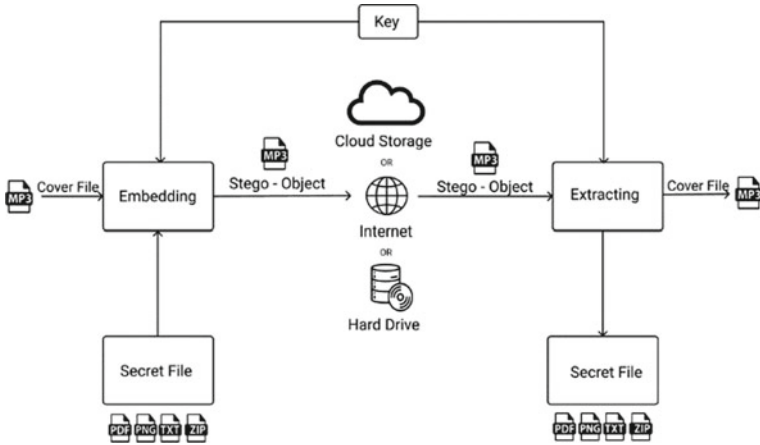


Fig. 1 Architecture diagram

5 Implementation

5.1 Registration

User needs to register an account that is stored into the local database using SQLite3, and only, a user can use the program.

5.2 User Input

A secret file of any format be it documents (pdf, txt, docx), images (jpg, jpeg, png) has to be chosen by the user which has to be hidden. As cover files, two audio files of wav format have to be chosen. To ensure that the subsequent stego-object is noise-free, the first audio should be smaller, and the second audio should be larger than the first.

5.3 Embedding the Secret File into the Audio Files

Prepare: The smaller and larger files are prepared for the secret file’s embedding in this section. To compute the number of LSB required to correctly embed the secret file into the audio files, values such as channel number, wave width, and wave frames are obtained and processed. Each time an audio file is embedded, the prepare function is invoked.

5.4 *Hide Data*

The secret file, which is now turned into a binary array, is then embedded with the readied audio file using the LSB approach after the prepare procedure is completed. Secret files' binary values are modified to match the values of audio files, allowing them to be successfully hidden.

5.5 *Recover Key*

After a successful embedding of the secret file, the user is given a base 64 recovery key that is applied to retrieve the secret file resulting stego-object.

5.6 *Extracting*

Once the user wants to extract the secret data, he has to login to the software, select the stego-object and recovery key, and using the same processing and LSB method used for embedding, the secret file is retrieved (Fig. 2).

5.7 *Description of Process*

Registration:

The program asks the user for credentials like username, password, and email id and stores them into a local DB called data. db using SQLite3. The function reg() does the abovementioned process (Fig. 3).

Secret file to binary conversion:

Since we are using LSB encoding to hide the data, the entire secret file is converted into a binary array. This is done using the f.open built-in function (Fig. 4).

Embedding:

Since the audio file contains a waveform, there is a lot of processing involved. In the processing phase, the width of the wave, frame, and required LSBs are calculated and repeated twice for both audio files. Once the processing is complete, the binary array from the secret file is embedded into the LSB of the waveform, and using the base64 function, a recovery key is generated (Fig. 5).

Extraction:

Once the user wants to extract the secret data, he has to login to the software, select the stego-object and recovery key, and using the same processing and LSB method used for embedding, the secret file is retrieved (Fig. 6).

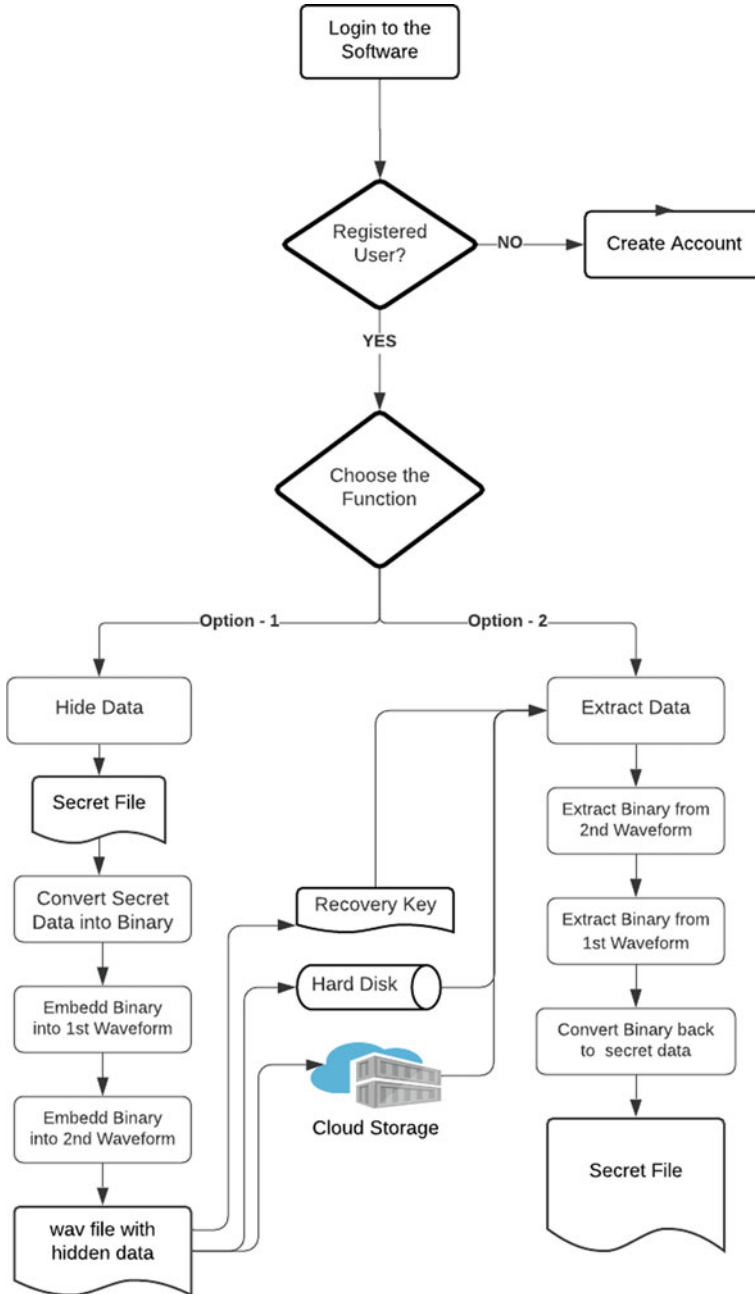


Fig. 2 Methodology diagram



Fig. 3 Registration process

Fig. 4 File conversion process

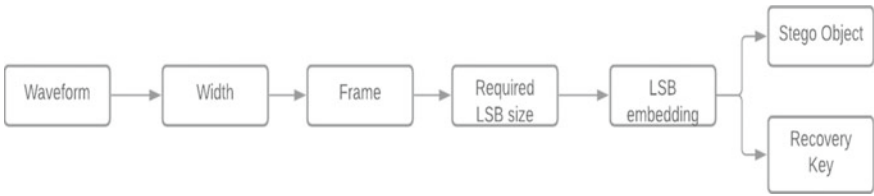


Fig. 5 Embedding process

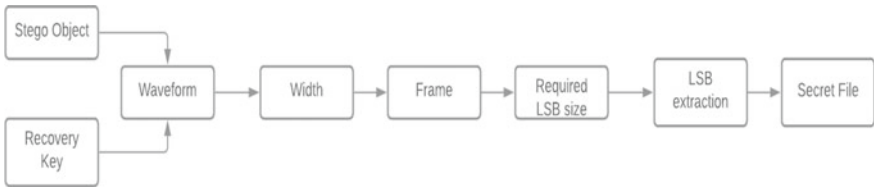


Fig. 6 Extraction process

6 Experiments and Results

The system uses two audio files to ensure that the resulting stego-object has minimum noise. To check if we have achieved our objective, the resulting stego-object is verified over two metrics. The two metrics are PNSR and spectrogram. In the PNSR, minimum PNSR value a stego audio file must have to be called robust is 30 dbs. The application was 90% successful in obtaining the 30 dB value. When employing steganography, a spectrogram is utilized for comparing the frequencies of two audio samples within a file. The system shows small difference in the resulting graphs for about 95% of the tests.

6.1 Peak Signal-To-Noise Ratio (PSNR)

We use PSNR as a metric to check the noise level contrasting the original audio to the stego-object. The PSNR esteem was acquired from the examination of sign strength with the steganographic procedure. A high PSNR value shows great sound quality. On the other hand, a low PSNR value deteriorated sound quality with a lot of noise. A sound quality rating of a minimum of 30 dB is considered best [13]. The PSNR formula is written in Eq. 1.

$$PSNR = 10 \times 10 \log \left(\frac{\sum_{i=1}^m x_1^2}{\sum_{i=1}^m (x_1 - x_0)^2} \right) \tag{1}$$

x^0 = peak signal WAV audio cover before steganography.

x^1 = peak signal WAV audio cover after steganography.

Tables 1 and 2 are the tests we have carried out against various file formats and sizes of the secret file, smaller audio file, and larger audio file, to find the PSNR. As the results are evident, the system returns PSNR values of 30 dB or greater for every combination we tried out. We can also concur that a larger audio file is more robust simply because there are a lot of bits to be worked with.

Table 1 PSNR test case for various file formats

File format	File size (KB)	Audio file size (KB)		PSNR (dB)
		Smaller audio	Larger audio	
.txt	500	784	1619	31.078
.pdf	459	784	1619	31.079
.jpeg/.jpg	502	784	1619	30.999
.png	492	784	1619	30.998

Table 2 PSNR test case for text format

File format	File size (KB)	Audio file size (KB)		PSNR (dB)
		Smaller audio	Larger audio	
.txt	100	143	250	30.999
	100	261	1050	34.297
	500	784	1619	31.078
	500	1050	5105	35.168

6.2 Signal Spectrum

Signal spectrum is an approach to see the distinction in the after-effects of the execution of the different steganography methods utilized. There is no huge contrast between the LSB technique and the changed LSB technique, so the natural eye won't discover the contrast. However, there is a distinction in the spectrogram between the first sound information and the second information. This occurs because the information changes given are no longer after the sound information data are put away in the header (Figs. 7 and 8).

The differences between the stego-object in Fig. 3 which has the secret file embedded with less noise, and contrast it with stego-object in Fig. 4 which has a secret file embedded with two cover audio files.

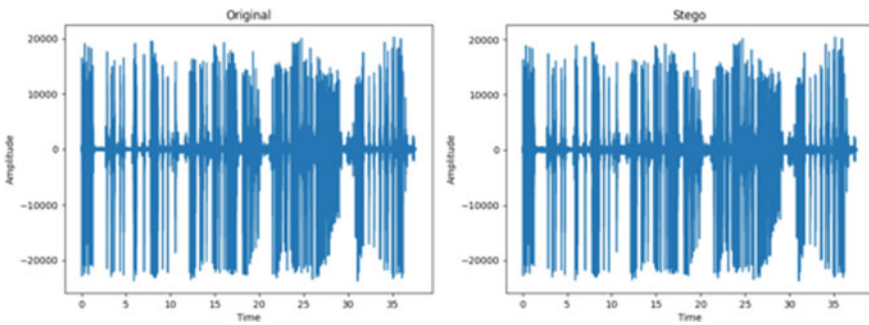


Fig. 7 Spectrogram for single audio cover

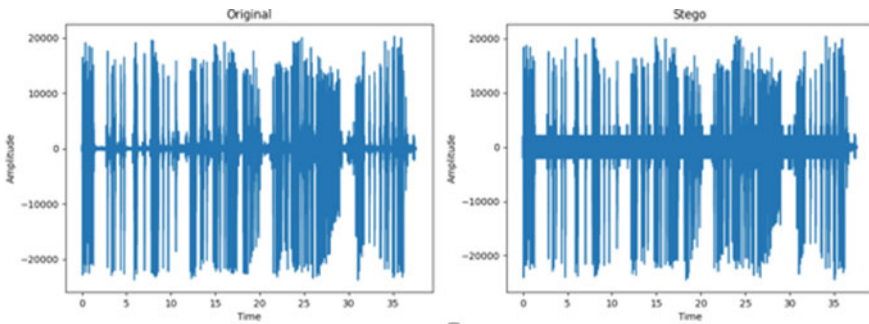


Fig. 8 Spectrogram for double audio cover

7 Conclusion

The system is successful in achieving its objectives to hide multiple file formats into an audio file. It is robust and secure in the sense; the extracted secret file is intact without any corruption, and both audio files have minimum to no noise, while they are embedded with secret files. While we can successfully have embedded the files into wav format, we have a few issues with mp3 file embedding; the mp3 file is a heavily compressed audio format which brings the issues like size imbalance. These issues can be addressed in the future.

References

1. Artz D (2001) Digital steganography: hiding data within data: *Internet Computing*. IEEE 5(3):75–80. <https://doi.org/10.1109/4236.935180>
2. Amin MM, Salleh M, Ibrahim S, Katmin MR, Shamsuddin MZI (2003) Information hiding using steganography. In: 4th National conference of telecommunication technology. <https://doi.org/10.1109/NCTT.2003.1188294>
3. Hmood DN, Khudhiar KA, Altaei MS (2012) A new steganographic method for embedded image in audio file. *Int J Comput Sci Secur (IJCSS)* 6(2):135–141
4. Morkel T, Eloff JH, Olivier MS (2005) An overview of image steganography. *Proceedings of the ISSA*
5. Chung YY, Xu FF, Choy F (2006) Development of video watermarking for MPEG2 video. In: *TENCON 2006—2006 IEEE region 10 conference*. <https://doi.org/10.1109/TENCON.2006.343843>
6. Jian CT, Wen CC, Rahman NH, Hamid IR (2017) Audio steganography with embedded text. In: *IOP conference series: materials science and engineering*. Vol 226. <https://doi.org/10.1088/1757-899X/226/1/012084>
7. Chandrakar P, Choudhary M, Badgaiyan C (2013) Enhancement in security of lsb-based audio steganography using multiple files. *Int J Comput Appl* 73(7) (0975-8887). <https://doi.org/10.5120/12754-9705>
8. Timothy AO, Adebayo A, Junior GA (2020) Embedding text in audio steganography system using advanced encryption standard, text compression, and spread spectrum techniques in Mp3 and Mp4 file formats. *Int J ComputAppl* 177:46–51. <https://doi.org/10.5120/ijca2020919914>
9. Indrayani R (2020) Modified LSB on audio steganography using WAV format. In: *3rd International conference on information and communications technology*, pp 466–470. <https://doi.org/10.1109/ICOIACT50329.2020.9332132>
10. Python wave library: <https://docs.python.org/3/library/wave.html>
11. Python base64 library:<https://docs.python.org/3/library/base64.html>

Run-time Control Flow Model Extraction of Java Applications



Gokul Saravanan, Goutham Subramani, P. N. S. S. Akshay, Nithesh Kanigolla, and K. P. Jevitha

1 Introduction

In today's world, software applications are being increasingly used everywhere. Software applications provide additional features to embedded devices, smartphones, etc., in addition to typical software such as spreadsheets and presentation slides. Particularly in embedded application domains, it is critical to ensure that the deployed software is right, stable and reliable. The working of software has drastically changed and involves a lot of complex in-built and third-party libraries. Run-time control flow model extraction for these softwares depends extensively on these libraries, and it is important for an extraction tool to recognize these important modules.

Run-time control flow model extraction is a way for extracting data from a running system, and it is a method focused on computational system execution and analysis. The run-time models are useful to understand the working of large complex applications and also to aid verification of vital properties of an application, to ensure that the application is working as intended based on the properties. Run-time control flow model extraction tools for Java applications have been typically a few, and the

G. Saravanan · G. Subramani · P. N. S. S. Akshay · N. Kanigolla · K. P. Jevitha (✉)
Department of Computer Science and Engineering, Amrita School of Engineering,
AmritaVishwaVidyapeetham, Coimbatore, India
e-mail: kp.jevitha@cb.amrita.edu

G. Saravanan
e-mail: cb.en.u4cse17116@cb.students.amrita.edu

G. Subramani
e-mail: cb.en.u4cse17117@cb.students.amrita.edu

P. N. S. S. Akshay
e-mail: cb.en.u4cse17138@cb.students.amrita.edu

N. Kanigolla
e-mail: cb.en.u4cse17124@cb.students.amrita.edu

amount of code coverage in these tools is low. A lot of code in the application part is left undetected. We have created a bytecode instrumentation tool based on the DiSL framework in order to extract the run-time execution trace of the applications. The application code along with its libraries can be instrumented, and their execution trace can be collected using the tool. This trace can be used to construct the run-time model of the application, on which the properties of interest can be verified.

2 Literature Review

We provide a brief literature review in this section.

In [1], the author presents domain-specific language for instrumentation (DiSL), which is a new framework created specifically for the study of dynamic programs. Bytecode instrumentation is being relied on by the majority of analysis applications used for Java. Dynamic analysis can be time consuming as it works on low-level code libraries. Despite the fact that AOP provides a higher amount of abstractions, AOP languages like AspectJ are not preferable for analysis, because the weaving of code used by these languages does not look for the overhead costs. Unlike other languages, DiSL provides an open joint model which enhances the code coverage.

In [2], the author explains the use of DiSL and provides an explanation through examples over its features such as data parsing, static and dynamic context and loop guard and explains the steps taken to make DiSL function efficiently in Java.

In [3], the author explains different dynamic analysis tools, their uses and the model and language these tools adopt and then explains about DiSL and how it compares with other dynamic analysis tools and evaluates the comparison.

In [4], the author describes the JIVE tool. It represents a unique take on the time constraints and analysis of Java programs. It explores understanding the code and debugging, with different custom views of the structure of the class objects.

In [5], the author introduces integrated components to extract finite-state models from the source code of the program called Bandera. The paper explains the components of Bandera, namely Slicer, Abstraction Engine, Back End and UI. It further demonstrates Bandera by applying it on a threaded pipeline-based program in Java.

In [6], the author provides a technique to extract the CFG of a program from an incomplete bytecode program. Incomplete bytecode consists generally of a few components for which there is no availability of these components. They also used the ConFlex tool to extract the control flow graph from the program.

In [7], the author introduces reflexion models for the extraction of models from state machines. They further explain the components and the scope of these reflexion models. Finally, they explore the potential uses of the model for various case studies.

In [9], the author explains the use of finite-state models for discrete systems. It also presents algorithms for model extraction and also for model abstraction in order to reduce the size of extracted models for longer executions.

In [10], the author describes JIVE and its two forms of run-time visualizations of Java programs—object diagrams and sequence diagrams. It also proposes a labelling

scheme based upon regular expressions to compactly represent long sequences and an algorithm for computing these labels in the representations.

In [11], the author describes how JIVE helps in consistency of the Java run-time behaviour with design-time specifications point of view.

In [12], the author proposes a technique that provides a clear and concise picture of the history of program execution with respect to entities of interest to a programmer. It presents the technique along with experimental results from summarizing several different program executions in order to illustrate the benefit of our approach.

In [13], the author describes Adrenalin RV, a run-time verification tool for android. Adrenalin RV overcomes the limited bytecode coverage issue found in many run-time verification tools. Adrenalin RV is based on DiSL which is a dynamic program analysis framework. Adrenalin RV uses load-time weaving to intercept every class that is loaded by the VM during the load time, which removes the problem of static weaving which only intercepts the APK classes. This allows Adrenalin RV to monitor all classes (including zygote).

In [14], it builds a new run-time verification framework to perform RV on multiple processes on the android platform. This new framework is built upon Adrenalin RV which uses DiSL, a dynamic program analysis. In this framework, the event order is recorded by extending libraries such as Binder. This framework also has extended the regular expressions RE which enables RE that supports multiple processes.

In [15], the author presents an updated framework for run-time verification non-single android processes. This framework extends android's standard in-built IPC which is overridden in the Binder library, and for interaction between two processes, it deploys a shared-memory service. It is built on top of the multi-process framework and Adrenalin RV by the same author.

3 Architecture

The architecture is depicted in Fig. 1. The application can be used for run-time control flow model extraction using the source code or compiling it into a jar. By modifying the properties of DiSL, it can run corresponding to the method used (source code or JAR file).

A custom DiSL code for the application is stored along with the application in the file system.

The code along with application or source code is executed in the JVM environment, and run-time trace extraction takes place. The trace contains the order of execution of methods during the run time. The method trace obtained is then used in an analysis model which will produce the state diagrams of the method executions.

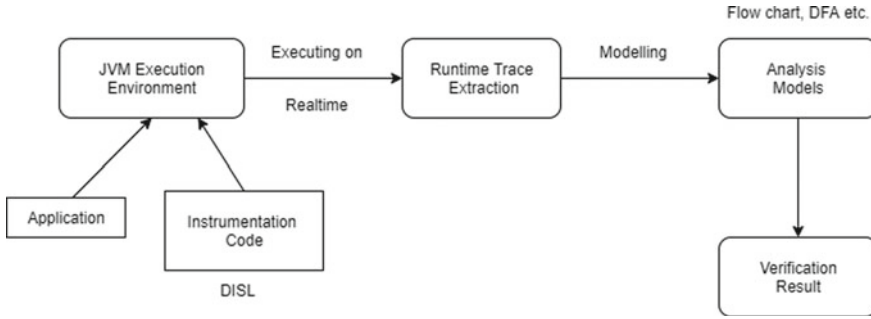


Fig. 1 Architecture

4 Implementation

DiSL provides various functionalities to instrument a class. DiSL has a number of libraries to make instrumentation to cover a wide amount of code possible. By doing so, we explored and learned more about different functionalities such as marker class, annotations and different contexts. One of the most important parts of the framework is marker class, there were several default marker classes available to use, and it was also possible to create custom markers to suit the needs of the research.

Marker classes point out the shadow event points in which the code is to be implemented. In other words, marker class specifies the location in which the code has to be instrumented in the target class.

We have used `BodyClassMarker` and `MethodInvocationMarker`. `BodyClassMarker` refers to the body of the methods and is one of the default classes, whereas `MethodInvocationMarker` is a custom marker that refers to the method call of the functions.

The next part of DiSL is the annotations allowed such as `@After`, `@Before`, and `@AfterReturning`. The annotations mention whether the instrumentation code is to be inlined before or after the aforementioned marker. Then we have to mention the methods for which we want to instrument the code in a particular format. (e.g. `Main.Run()`, `Fork.*`).

Once we were finished with the markers and annotations, we started to work with the contexts. The contexts are the data carriers, and there are many types of them such as method static context which contains all the necessary information about the method including the class it belongs to, the file in which the method code exists and the thread in which it is being executed.

Next, we used dynamic context which will give us the instance of the object itself which we can use to manipulate the data. There are also many other contexts such as argument context which we can use to get the data of the arguments of a method.

We predominantly used method static context and dynamic context to extract the desired data. Once we had the data, we started working on exporting the data into the

desired format for the project, i.e. CSV. We used `FileWriter` class to write the traced data into the output file.

Once the coding part is done, we started to instrument the target files using the DiSL framework in JVM. The input of the DiSL should be in the format of two JAR files: one file containing the target classes and the other file containing instrumented DiSL classes. During execution, whenever the target classes are called by the JVM for execution, they will be instrumented using the instrumentation class. The instrumentation process will be done by the DiSL framework. While the target application is being executed, the trace data will be written to the CSV file.

Once the CSV has been derived, we used the JIVE tool in order to build the state diagram. JIVE is a plug-in extension of Eclipse IDE which allows us to build the state diagram with the CSV in the required format. So we modify the format of CSV in order to match the requirements for the JIVE tool. Then we used the trace file to build our state diagram using the JIVE tool.

4.1 DiSL

For instrumentation of Java applications, DiSL has been chosen due to the wide coverage of the application DiSL provides for bytecode instrumentation.

Many dynamic analysis tools for programs written in managed languages such as Java rely on bytecode instrumentation. Bytecode instrumentation covers more areas than traditional instrumentation techniques. As shown in Fig. 2, the instrumentation covers the scope of the class mentioned at the 'Before' tag. Tool development is often tedious because of the use of low-level bytecode manipulation libraries. While aspect-oriented programming (AOP) offers high-level abstractions to concisely express certain dynamic analyses, the join point model of mainstream AOP languages such as AspectJ is not well suited for many analysis tasks, and the code generated by weavers in support of certain language features incurs high overhead.

DiSL contains the following features which are being implemented: instrumentations, snippets, markers, control of snippet order, synthetic local variables, thread-local variables, static context information and dynamic context information. Based on these features, custom DiSL code is created which can retrieve the trace file.

The instrumentation file will try to access the information related to the method which is being accessed and store the relative information in CSV format. The trace would consist of all the methods invoked during the program.

The trace file which produces information similar to Fig. 3 would consist of the current thread in which the method was invoked, serial no., the corresponding class's source file, the status of the method (entry, exit or called) and context and target method. By obtaining the trace file, we can generate state diagrams.

```

@Before(marker = BodyMarker.class, scope = "Main.main")
public static void beforeMethod(DynamicContext di, MethodStaticContext m) {
    String obj="";
    if(di.getThis()==null)obj="Main";
    else obj=di.getThis().toString();
    try {
        FileWriter writer = new FileWriter("trace.csv");
        writer.write("\n"+Thread.currentThread().getName()+
            "\n","\n1","\n"+m.thisClassSourceFile()+":0","\n"+
            "Method Entered","\n"+Context= "+
            Thread.currentThread().getName()+",Target="+
            m.thisMethodFullName()+"\n");
        writer.write("\n\n");
        writer.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
    
```

Fig. 2 Sample instrumentation code using DiSL

main	1	Main.java:0	Method Entered	Context= main,Target=Main.main
main	2	Main.java:0	Method Call	Context= main,Target=Main.main
main	3	Main.java:0	Method Entered	Context= main,Target=Main.methodOne
main	4	Main.java:0	Method Exit	Context= main,Target=Main.methodOne
main	5	Main.java:0	Method Call	Context= main,Target=Main.main
main	6	Main.java:0	Method Entered	Context= main,Target=Main.methodTwo
main	7	Main.java:0	Method Exit	Context= main,Target=Main.methodTwo
main	8	Main.java:0	Method Call	Context= main,Target=Main.main
main	9	Main.java:0	Method Entered	Context= main,Target=Main.methodThree
main	10	Main.java:0	Method Exit	Context= main,Target=Main.methodThree
main	11	Main.java:0	Method Call	Context= main,Target=Main.main

Fig. 3 Sample trace

4.2 hasNext Property

‘hasNext’ is a method of Iterator class in Java. It is used to iterate a list of objects. This method has to be called before the next method. If the ‘next’ method is called directly, it will lead to an error as the operation is not safe.

So we have used DiSL to extract the control flow model of the program implementing the hasNext property as shown in Fig. 4.

4.3 Dining Philosophers Problem

The dining philosophers problem is a common example in the design of algorithms to demonstrate synchronization problems and solutions. It has been one of the classic

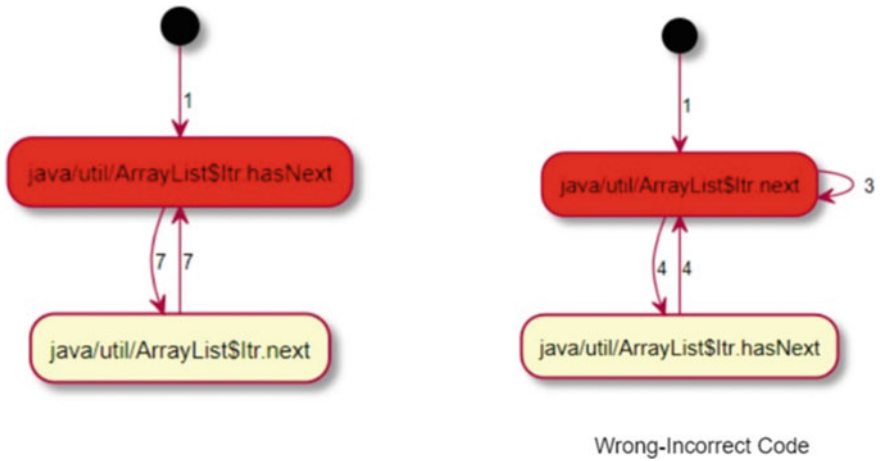


Fig. 4 State diagram generated for hasNext example

problems used to explain synchronization issues in multiple threaded problems and explore different ways to attempt the problem, which is one of the key reasons for choosing it. The issue was created to demonstrate the difficulties of preventing deadlock, a situation where multiple threads wait for each other indefinitely.

Working on an example program like this might work properly, but to check whether the trace and state diagrams can be generated with a more sophisticated program that uses a multi-threading concept is necessary to understand the scope of the work. For implementing dining philosophers, we have studied the packages provided by the DiSL framework and wrote instrumentation codes to extract method calls and trace.

We have tested the instrumentation code written, a part of the code shown in Fig. 5, for the dining philosopher as an example problem which is a multi-threaded program and have obtained the trace files. We have successfully modelled the obtained trace files into state diagrams.

With the example proving positive results, proceeding with a bigger Java application proved to be the important part of the research.

4.4 Single-Layer Perceptron Application

A single-layer perceptron is a learning algorithm for binary classifiers. Binary classifiers determine if an input, which is typically interpreted as a series of vectors, belongs to one of the several classes. It is a single-layer neural network in simple terms. It is used in supervised learning as a rationale for choosing it. Also, the single-layer perceptron application is Java executable, it is an open-source project available on GitHub [8], and it works on the Ubuntu platform.

```

@After(marker = BodyMarker.class, scope = "Fork.p*")
public static void afterMethod1(DynamicContext di, MethodStaticContext m) {
    String obj="";
    if(di.getThis()==null)obj="Main";
    else obj=di.getThis().toString();
    try {
        FileWriter writer = new FileWriter("trace.csv",true);
        writer.write("\\"+Thread.currentThread().getName()+"\\",\\"1\\",\\""+m.thisClassSourceFile()+":0\\",\\""+Method Exit\\",\\"Context= "+Thread.currentThread().getName()+",Target="+m.thisMethodFullName()+"\\"");
        writer.write("\r\n");
        writer.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

Fig. 5 Partial instrumentation code for dining philosophers problem

It aids in the classification of the input data. In addition, it receives several inputs and may return output when it reaches a certain point.

This can then be used to determine the target of a sample in the sense of supervised learning and classification.

Single-layer perceptron Java application [8] uses the single-layer perceptron neural network for binary classification of data. This application is implemented using Java Swing. The DiSL instrumentation implemented for this Java application required a higher amount of instrumentation methods to bring maximum code coverage. A portion of the instrumentation code is shown in Fig. 6.

After implementing, we have successfully managed to get the trace of a Java executable program. We have also obtained the state diagram for the resulting trace. Then we filtered the trace to obtain a higher-level state diagram.

We have selected perceptron, a Java application that produces a single-layer perceptron model of any data set given, as the testing application for instrumentation. We have attached the trace files, state diagrams, statistics of the trace file and other required screenshots.

```

@Before(marker = BodyMarker.class, scope = "Perceptron.l*(..)",order=1)
public static void before2(DynamicContext di, MethodStaticContext m) {
    System.out.println(m.thisMethodFullName());
    try {
        FileWriter writer = new FileWriter("trace.csv",true);
        writer.write("\\"+Thread.currentThread().getName()+"\\",\\"1\\",\\""+m.thisClassSourceFile()+":0\\",\\""+Field Write\\",\\"Context= "+Thread.currentThread().getName()+",Target="+m.thisMethodFullName()+"\\"");
        writer.write("\r\n");
        writer.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

Fig. 6 Partial instrumentation code for single-layer perceptron application

We have trained the single-layer perceptron using the training data set we have created. We have trained the data set based on the parameters like learning rate, threshold, maximum convergence and initial weights range. Based on the test data we have chosen, we have obtained final threshold, synaptic weights, training recognition rates and testing recognition rates as the final results.

5 Results and Discussion

5.1 Trace Files and Models Obtained of the Dining Philosophers Problem

Figure 7 shows the trace generated for the dining philosophers problem execution, and each row represents a different nature of the program. The trace files contain information such as thread name, index, file name and action performed at that particular stage with method name included. Based on the trace, there are three different states: thinking, hungry and eating, and we can determine from those trace whether the philosopher was thinking or hungry or eating.

We have also obtained the models like field state diagram and method call state diagrams shown in Figs. 8 and 9 using the trace files we have generated. In the field state diagram, the terms T, H and E stand for thinking, hungry and eating. The diagram represents the situation of each philosopher in different stages of time. Field state diagram mentions the value of the state whether the philosophers are thinking, hungry or eating. If you consider the first state, there are 5 Ts which represent that they are five values which are threads and all the five are thinking. The other states also similarly represent the different stages of the process.

Thread-1	69	Main.java:0	Method Entered	Context= Thread-1,Target=Philo.Hungry
Thread-1	70	Main.java:0	Method Exit	Context= Thread-1,Target=Philo.Hungry
Thread-1	71	Main.java:0	Method Call	Context= Thread-1,Target=Philo.Hungry
Thread-1	72	Main.java:0	Method Call	Context= Thread-1,Target=Philo.run
Thread-1	73	Main.java:0	Method Entered	Context= Thread-1,Target=Fork.pickup
Thread-1	74	Main.java:0	Method Call	Context= Thread-1,Target=Fork.pickup
Thread-1	75	Main.java:0	Method Exit	Context= Thread-1,Target=Fork.pickup
Thread-1	76	Main.java:0	Method Call	Context= Thread-1,Target=Philo.run
Thread-1	77	Main.java:0	Method Entered	Context= Thread-1,Target=Philo.Eating
Thread-5	78	Main.java:0	Method Exit	Context= Thread-5,Target=Philo.Thinking
Thread-5	79	Main.java:0	Method Call	Context= Thread-5,Target=Philo.Thinking
Thread-5	80	Main.java:0	Method Call	Context= Thread-5,Target=Philo.run

Fig. 7 Trace file for dining philosopher

AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.lambdaLoadFile\$6
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.startTrain
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.trainPerceptron
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.getRandomNumber
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.getRandomNumber
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.getRandomNumber
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.testPerceptron
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.testPerceptron
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.trainPerceptron
AWT-EventQueue-1	1	Perceptron.java:0	Field Write	Context= AWT-EventQueue-1,Target=Perceptron.trainPerceptron

Fig. 10 Trace file for single-layer perceptron application

5.2 Trace File and Model We Have Obtained for Single-Layer Perceptron Application

Image shown in Fig. 10 is the method trace that was obtained by executing the Java application single-layer perceptron application.

Each row represents a different stage of the program. The trace files contain information such as thread name, index of the state, file name and action performed at that particular stage with method name included. This trace file is helpful in obtaining a method state diagram of the application.

Using the trace file generated, we have obtained method call state diagram, shown in Fig. 11, for the perceptron. The obtained state diagram contained methods related to the UI application. So to obtain a state diagram for only the core aspects of the application, the trace file obtained is modified in the instrumentation code to ignore these UI methods and correspondingly a new state diagram is obtained as shown in Fig. 12. In the method call state diagram, it can be noted that the method invocation order can be obtained and used for verification purposes. The single-layer perceptron application consists of methods such as loadFile, setFrame, resetData, startTrain, trainPerceptron, RandomNumber and TestPerceptron.

6 Conclusion and Future Work

In this article, we used a different technique for run-time control flow model extraction of Java applications with the help of the DiSL framework. We have successfully generated the state diagrams and method call diagrams from the method trace obtained. We have extracted the trace for the selected Java applications, and we have created models for that trace extracted correspondingly. Furthermore, instrumentation for a classical Java problem has been carried out with positive results. Finally, we provided a case study, where the technique was implemented and the result was obtained and analysed.

This research is done with Java applications. Currently, it can be used on Java applications to extract the required control flow models. There are real-world applications that use Java applications in their embedded systems. These applications can

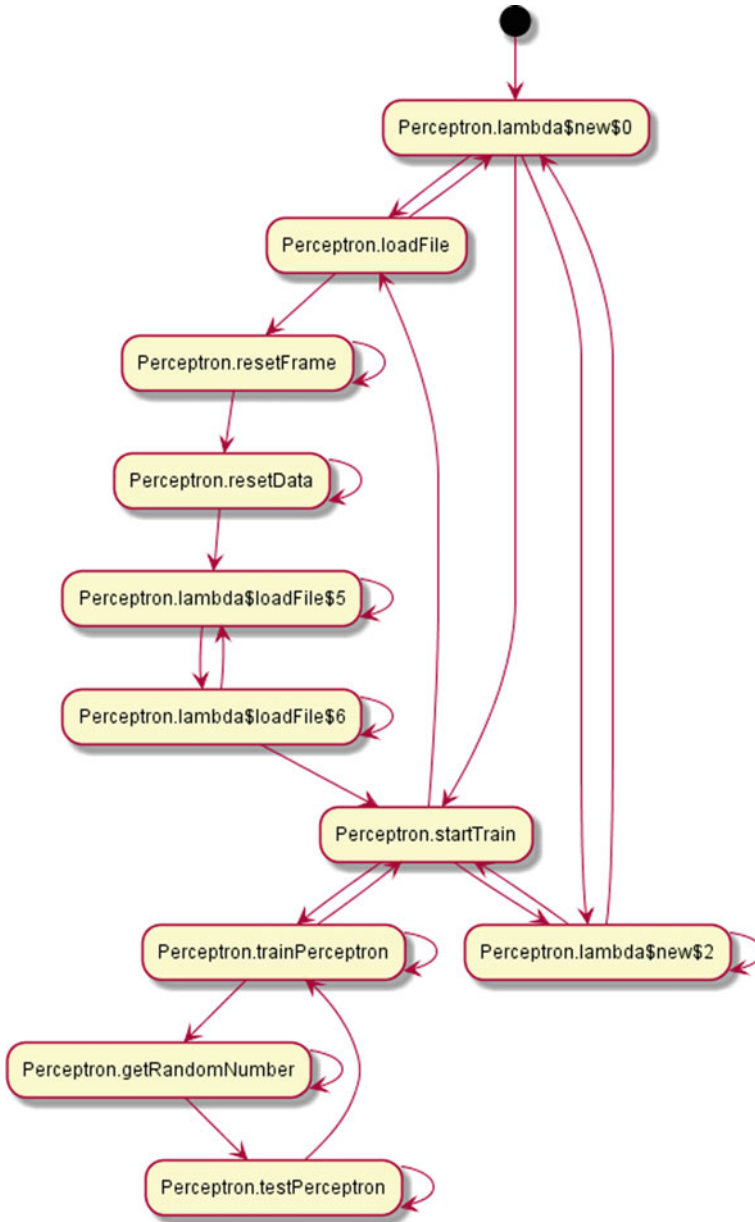


Fig. 11 Modified method state diagram for single-layer perceptron application

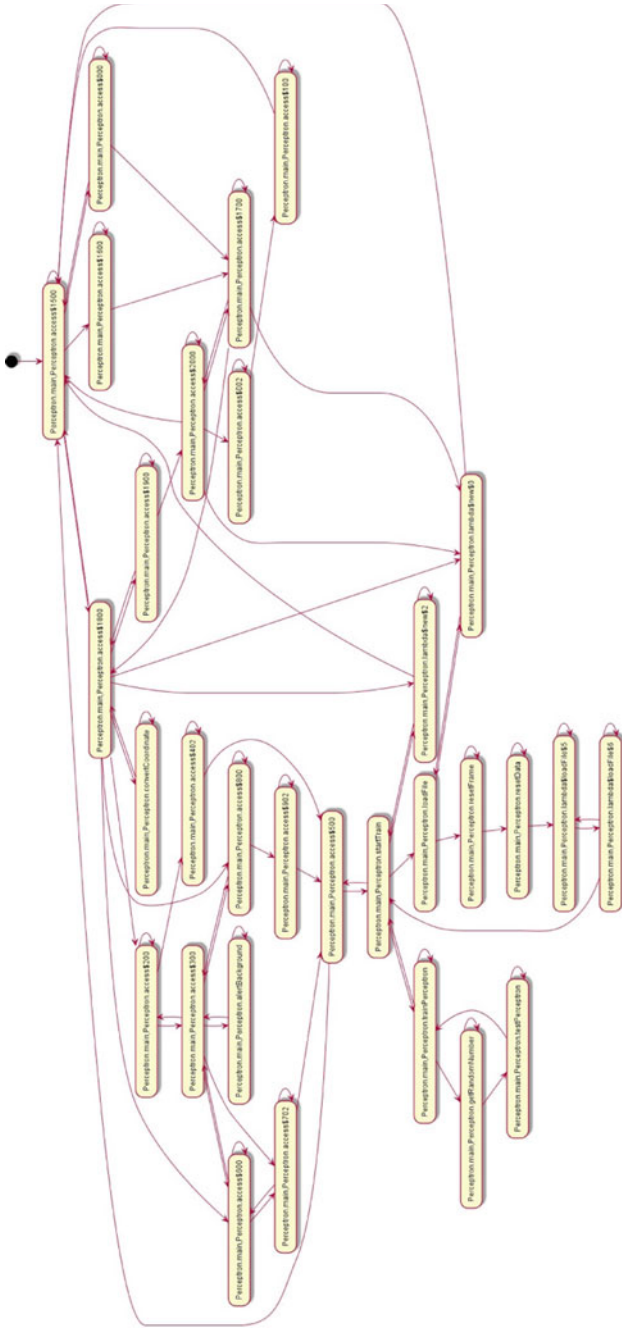


Fig. 12 State diagram obtained for perceptron application

also be used for the same purposes, as these applications are uncommon and not popular at present but have a big role in future. It can be used to extract models for many real-world Java applications containing various methods and classes. The bytecode instrumentation can be improved in future to support events like field values.

References

1. Marek L, Villaz A, Zheng Y, Ansaloni D, Binder W, Qi Z (2012) DiSL: a domain-specific language for byte code instrumentation. In: AOSD '12: proceedings of the 11th annual international conference on aspect-oriented software development
2. Marek L, Zheng Y, Ansaloni D, Sarimbekov A, Binder W, Tuma P, Qi Z (2012) Java bytecode instrumentation made easy: the DiSL framework for dynamic program analysis. In: Asian symposium on programming languages and systems (APLAS)
3. Marek L, Zheng Y, Ansaloni D, Bulej L, Sarimbekov A, Binder W, Tuma P (2015) Introduction to dynamic program analysis with DiSL. *Sci Comput Program* 100–115
4. Gestwicki PV, Jayaraman B (2004) JIVE: Java interactive visualization environment. In: OOPSLA '04: companion to the 19th annual ACM SIGPLAN conference on object-oriented programming systems, languages, and applications
5. Wu J (2016) Single-layer perceptron [source code]. <https://github.com/Jasonnor/Perceptron>
6. Corbett JC, Dwyer MB, Hatcliff J, Laubach S, Pasareanu CS, Robby, Zheng H (2000) Bandera: extracting finite-state models from Java source code. In: ICSE '00: proceedings of the 22nd international conference on Software engineering
7. de Carvalho Gomes P, Picoco A, Gurov D (2014) Sound control flow graph extraction from incomplete Java bytecode programs. In: Conference: fundamental approaches to software engineering
8. Said W, Quante J, Koschke R (2018) Reflexion models for state machine extraction and verification. In: 2018 IEEE international conference on software maintenance and evolution (ICSME)
9. Jevitha KP, Jayaraman S, Jayaraman B, Sethumadhavan M (2021) Finite-state model extraction and visualization from Java program execution. In: *Software: practice and experience*
10. Jayaraman S, Jayaraman B, Lessa D (2017) Compact visualization of Java program execution. In: *Software: practice and experience*
11. Jayaraman S, Hari D, Jayaraman B (2015) Consistency of Java run-time behaviour with design-time specifications. In: Proceedings of eighth international conference on contemporary computing (IC3), Noida
12. Jayaraman S, Diwakar KK, Jayaraman B (2014) Towards program execution summarization: deriving state diagrams from sequence diagrams. In: Seventh international conference on contemporary computing (IC3)
13. Weaving HS, Rosa A, Javed O, Binder W (2017) ADRENALIN-RV: Android runtime verification using load-time weaving. In: IEEE international conference on software testing, verification and validation (ICST)
14. Sun H, North A, Binder W (2017) Multi-process runtime verification for android. In: 24th Asia-Pacific software engineering conference (APSEC)
15. Villaz A, Sun H, Binder W (2018) Capturing inter-process communication for runtime verification on Android. In: International symposium on leveraging applications of formal methods

Accelerating Real-Time Face Detection Using Cascade Classifier on Hybrid [CPU-GPU] HPC Infrastructure



B. N. Chandrashekhar and H. A. Sanjay

1 Introduction

In the recent era, for security purposes, authentication is mandatory in all domains, e.g., robotics, surveillance, law enforcement, interactive game applications. Authentication is the process of recognizing a person's identity, this can be done using a biometric system. It uses human features like fingers structures, faces facts, etc. By using these features, a biometric system compares with existing data then identifies a person. There are plenty of biometric systems like iris recognition, fingerprint recognition, face detection, etc., are available in the market. These biometric systems are used to identify criminals easily in any sectors.

The face is a complex multiple-layer construction and needs great registering procedures for acknowledgment. The face is our essential and first focal point of consideration in public activity assuming a significant part in the character of a person. We can perceive the number of countenances learned for the duration of our lifetime and personality that face at glance even later years [1]. There might be varieties in faces because of maturing and interruption. To identify the given person, we need something which should be more reliable for verifications or identifications. To suit this requirement, there is a device that uses finger impressions and voice of the persons that is called the biometric system that uses automatic methods to identify the persons accurately. The qualities are quantifiable and interesting. This is an average

B. N. Chandrashekhar (✉)

Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore 560064, India
e-mail: chandrashekar.bn@nmit.ac.in

H. A. Sanjay

Department of Information Science and Engineering, M.S. Ramaiah Institute of Technology, Bangalore 560054, India

circumstance where the degree of safety given is given as the measure of cash the fraud needs to acquire unapproved access [2].

There are a few strategies in distinguishing faces. Highlight-based methodologies utilize nearby facial elements, nose, mouth, eyes, and the underlying connection among them. These strategies are thought of as strong against brightening changes, impediments, and perspectives. Nonetheless, great superiority pictures are essential, and the computational techniques are costly. Another methodology is appearance-based techniques where the position of the face is viewed as a two-class design acknowledgment issue [3]. The arrangement depends on highlights determined from pixel esteems in the hunt window. A few element types are being used, for example, a Haar-like classifier is made by utilizing factual learning over a huge arrangement of tests [4].

Face identification is costly. It is a challenging task for hybrid [CPU-GPU] foundations [5]. The graphics processing units (GPUs) have enormous equal figuring assets, just as superior execution with drifting point activities and high memory data transfer capacity in hybrid [CPU-GPU] foundations. With the appearance of compute unified device architecture (CUDA) [6] and open detail for computation vision (OpenCV), these assets have opened up to conventional figuring [7]. Not like equally distributing [8] on CPUs-based HPC infrastructures[clusters] [9], light computation in hybrid [CPU-GPU] foundations a GPU has parallelism an enormous number of lightweight bit occasions (additionally called strings, work-things, or microkernels) are lined and dispatched. Regularly, every one of these will deal with calculations for one result component, implying that one handling cycle is acknowledged with up to a huge number of microkernels. Notwithstanding, as GPUs are exceptional processors, they must be used in speeding up calculations that can be fitted to GPU design speed increase.

The rest of the paper is structured as follows. Section 2 explains the related work of face detection. Section 3 hybrid [CPU-GPU] infrastructures used for real-time face detection. Section 4 faces detection strategy Sect. 5 describes the experimental results. We briefly conclude in Sect. 6.

2 Related Work

Real-time face detection with hybrid infrastructure inspires the attention of researchers in recent years. Several types of research have been shown overall performance acceleration of real-time face detection in different domains.

Daschoudhary et al. [4], authors considered face recognition and the head posture's position, SimpleCV and OpenCV libraries are utilized. The test result registered by utilizing SimpleCV and OpenCV on computer vision system authors observed 30 frames per seconds under 1080p with advanced precision and rapidity for face location and head present position happened by using libraries and the previously mentioned equipment.

Makela et al. [2], the authors present one space of interest is computationally substantial PC vision calculations, like face identification and acknowledgment. This

work accelerated the face detection algorithm by using accelerators like GPU with OpenCL. The main idea to accelerate was to improve the performance by preserving the functionality equally. Overall, from this implementation, we can see that GPU performs faster execution.

Sharma et al. [10], author proposed surveyed two calculations for recognizing individuals in night vision recordings. The proposed problem area calculation utilizes the dark body radiation hypothesis and the foundation deduction calculation utilizes the distinction picture got from the info picture and a created foundation picture. The outcome examination is done of the investigations performed on these methodologies.

Viola et al. [11] depicts how to handle the images very fast by using the artificial intelligence approach for visual article discovery which is equipped for handling pictures. In this study, author considered three key obligations. The first is the vital picture which authorizes the features utilized by the indicator to be processed quickly. The second is an AdaBoost-based classifier for learning calculation, which selects a few features from a larger set of data very proficiently. The third obligation is a course joining classifier, which gradually locates the picture to proximately inclined.

Lescano et al. [12], authors utilized the GPUs for the training phase to reduce the amount of time needed to process training data, and even authors made performance comparisons with the other works. Outstanding results were obtained by using the CUDA framework to reach adequate times for the training phase.

Mutneja et al. [13] explored three principle obligations in this work. Primarily, by using Haar features dependent on the Viola–Jones structure, the authors accomplished a noteworthy acceleration by parallelizing the location on GPU. Also, through AdaBoost investigation growing more inventive and productive methods for choosing classifiers for the errand of face recognition, which can additionally be summed up for an article location. Thirdly, execution of parallelization methods of an altered adaptation of Viola–Jones faces identification calculation in the mix with skin tone separating to diminish the hunt space has been finished. We have had the option to accomplish an impressive decrease in the pursuit existence cost by utilizing skin tone separating related to the Viola–Jones calculation. For 54.31%, at the picture goal of $640 * 480$, time cost decrease on GPU time versus CPU time has been accomplished by the proposed parallelized calculation.

3 Hybrid [CPU-GPU] Infrastructures Used for Real-Time Face Detection

To make effective resources utilization of multi-core CPUs and multi-core GPUs in heterogeneous computing platforms and to improve the productivity as well as the performance of the compute-intensive real-time face detection applications.

Figure 1 shows the proposed hybrid [CPU + GPU] architecture, where every node comprises CPU and GPUs having different computing capability graphics cards plugged on the slot. The GPU chips can be used together to make complete utilization

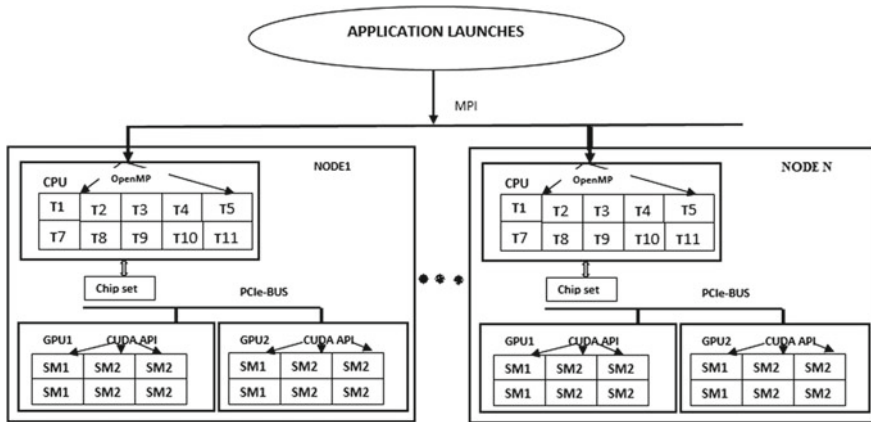


Fig. 1 Framework of hybrid [CPU + GPU] cluster environment with each node having two different computing capabilities GPU

of the CPUs and GPUs computation capacity. The GPUs performance changes with different workloads, while the performance of the CPU is moderately stable for the load running on it. In hybrid [CPU + GPU] architecture, PCIe-bus is used to interconnect CPUs and GPUs. Numerous cores with respective caches are available for every CPU. GPU consists of several streaming multiprocessors (SMs). In this architecture, CPUs are responsible for allocating tasks, initiating computation and controlling the GPUs, and finally, fetching the computed results from the GPUs [14].

Once compute-intensive real-time face detection application launches to the hybrid [CPU-GPU] infrastructures, we have considered pinned memory technique with a single MPI process. Single MPI process per node which improves the internodes communication overhead and memory bandwidth. MPI process initially transfers the computed workload to respective nodes in a heterogeneous cluster and then on each respective node for dynamic distribution of workload on the CPU and GPUs based upon their computing capabilities, speed, and hardware specification.

The MPI cycle produces numerous OpenMP threads as the number of cores in the CPU inside each node increases. Just the main thread helps out GPU and the others perform important math activities equally. On GPU, kernels will be launched parallelly, then by using grid dimensions, several blocks, and several threads, it processes the kernels very rapidly. Lastly, the results of the computed kernel will be transferred back to the CPU memory using the MPI process which is shown in Fig. 2.

4 Face Detection Strategy

Only CUDA-based GPU face detection has various issues such as:

- 1. Data communication overhead between CPU-GPU and vice versa
- 2. Only supports NVIDIA's GPUs

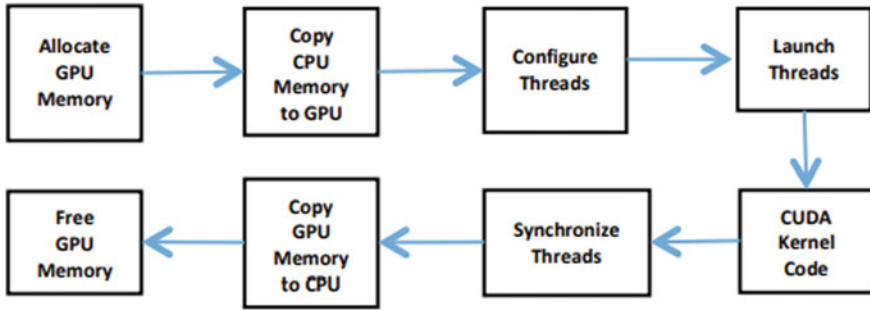


Fig. 2 Data transfer cycle in hybrid [CPU + GPU] cluster environment

3. Application dependency on the previous stage due to this efficiency will not be noticed on only CUDA-based GPU
4. Due to performance overhead, CUDA does not support exception handling but CUDA managed it by using thousands of threads.

To address these issues in this research work, we have considered hybrid [CPU-GPU] infrastructure to implement Viola–Jones face detector on CPU with OpenCV and GPU with CUDA improved the performance. This performance was cutting-edge in different ways to accelerate its speed.

4.1 Viola and Jones Algorithm

In the recent era, still in so many domains to detect the faces used Viola–Jones algorithm-based real-time face detection system. A face recognition algorithm should have dual important highlights, exactness, and speed. Three fixings are working in the show to empower a quick and exact detection. For efficient computational resource allotment, the basic picture includes integral image feature computation, AdaBoost feature selection, and loading of cascade classifier for efficient computational resource allocation.

1. Integral Image feature Computation

The basic picture is characterized as the summation of the pixel benefits of the first picture. The worth at any area (x, y) of the basic picture is the number of the picture’s pixels above and to one side of the area (x, y) . The figure beneath represents the vital picture age. The vital picture utilizes basic rectangular to determine a halfway portrayal of a picture. The cluster of pixels is known as an image. (x, y) is the pixel at the area complete? So, if the basic image is $A(x, y)$ and the vital picture is $AI[x, y]$, then the basic image is registered as displayed outlined in Fig. 3 (Figs. 4 and 5).

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

$$S(x, y) = s(x, y - 1) + (x, y)$$

$$ii(x, y) = ii(X - 1, y) + S(x, y)$$

Perhaps the 2D wave calculation is the most effective way to decrease the spatial information reliance made by basic pictures is as shown in Fig. 6. We have identified some of the issues concerned with this algorithm when deployed to hybrid [CPU-GPU] infrastructure as it includes scanty non-straight gets to of memory. In this way, a productive strategy is to carry out the necessary picture algorithm is by thinking about the stuff to perform rows and columns calculation.

Henceforth, the means of calculation are as per the following:

1. Evaluate the information from the rows in the picture
2. Rendering the computed results
3. Evaluate the rows in the results rendered
4. Result creating in the wake of rendering the last result.

Fig. 3 Summed area of integral image

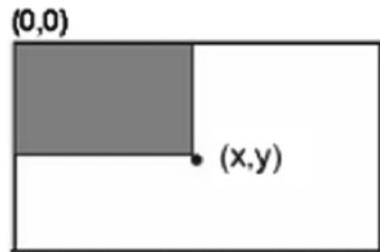


Fig. 4 Summed area of rotated integral image

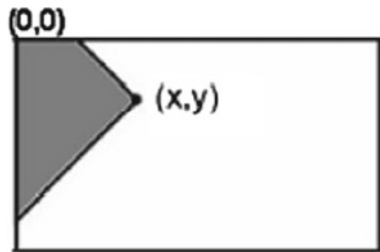


Fig. 5 a Edge features, b line features, c center surround features

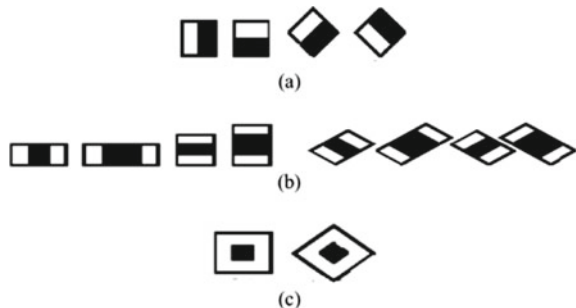
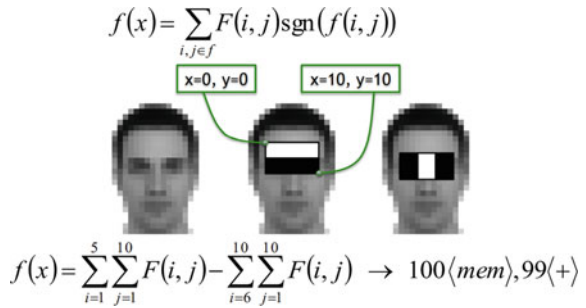


Fig. 6 Integral image computation using 2D wave approach



Filtering of all rows or columns is some way or another an alternate process contrasted with examining just a solitary vector. Since the information in a single row is lacking to give sufficient responsibility to the GPU, it is wasteful to filter the picture columns ‘n’ times and causes extra time overhead. Henceforth, this overhead can be tackled by using parallel programming paradigm CUDA square to handle the picture stripe separately square containing many threads on each row as displayed in Fig. 7.

$$I(x, y) = \sum_{(i,j)=(0,0)}^{(x,y)} F(i, j)$$



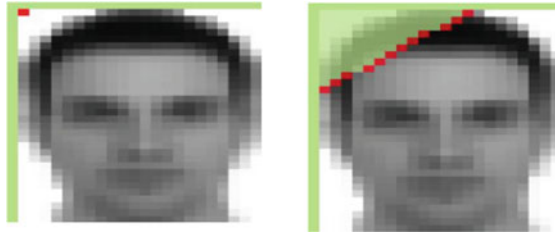
$$f(x) = (I(10, 5) + I(0, 0) - I(10, 0) - I(0, 5)) - (I(10, 10) + I(0, 5) - I(10, 5) - I(0, 10)) \rightarrow 6\langle mem \rangle, 5\langle + \rangle, 2\langle << \rangle$$

On CPU with OpenCV:

$$I(x, y) = F(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1) \quad xy > 0$$

Fig. 7 Integral Image generation of scan algorithm with 4 thread C'UDA blocks [15]

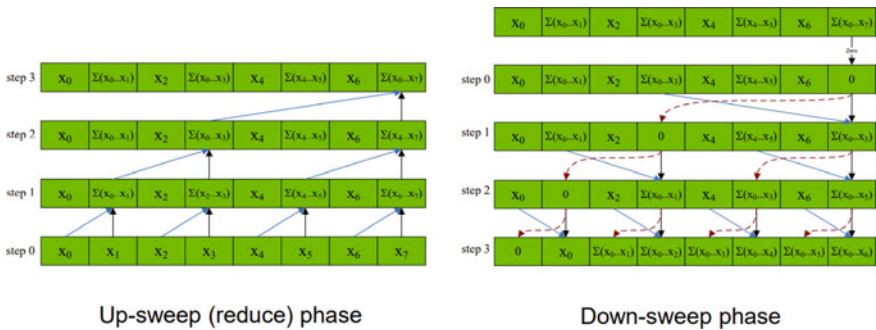
		Input								
		1	1	1	1	1	1	1	1	
Carrier	Iter #	Output								
0	0	0	1	2	3					
4	1	0	1	2	3	4	5	6	7	
8	2	0	1	2	3	4	5	6	7	8



On GPU way with CUDA:

$$I(x, y) = \sum_{(i,j)=(0,0)}^{(x,y)} F(i, j) = \sum_{i=0}^x \sum_{j=0}^y F(i, j)$$

Threads will heap many threads for every repetition from the row into memory over elite sweep and expansion of transporter. For the most portion of the transporter worth will be introduced to zero toward the start, however, it will be refreshed with the amount of all components being looked over to the second later the principal cycle. At last, the threads will move the outcome into the global memory. The scan is performed in 2 phases:



2. AdaBoost feature selection

AdaBoost (Adaptive Boosting) capacities arrange numerous highlights by just choosing specific elements. Supporting happens in emphasis, gradually adding the powerless student into one solid student. One weak learner gains from preparing information every emphasis. Then, at that point, the feeble beginner is added to the solid beginner. Later, the feeble beginner is supplementary, the information is then, at that point, altered for each mass. The information that experienced the classification blunder will encounter mass improvement and effectively ordered information will be exposed to mass decrease. Therefore, the feeble beginner in the following emphasis will be more focused on information that has been landing mile confidential by the past feeble beginner [16]. Adaptive boosting for highlight choice and an intentional course for effective calculation asset allocation [17].

3. Loading of Cascade Classifier for efficient computational resource allocation

On CPU, using OpenCV-based cascade classifier pictorial structure is saved without any effort for studying. On GPU, whole cascade classifier data is loaded and analyzed, collected pictorial information is saved in the DRAM of the GPU with the restricted surface location.

The Viola–Jones algorithm feature is a staggered arrangement [15, 18]. The arrangement of this Viola–Jones comprises 3 stages where each stage staples a portion of the image may be acknowledged to check whether in the image area face is present or not. Even this process is repeated to check whether a portion of the image is present or not. The slog process of different stages characterization is displayed in Fig. 8.

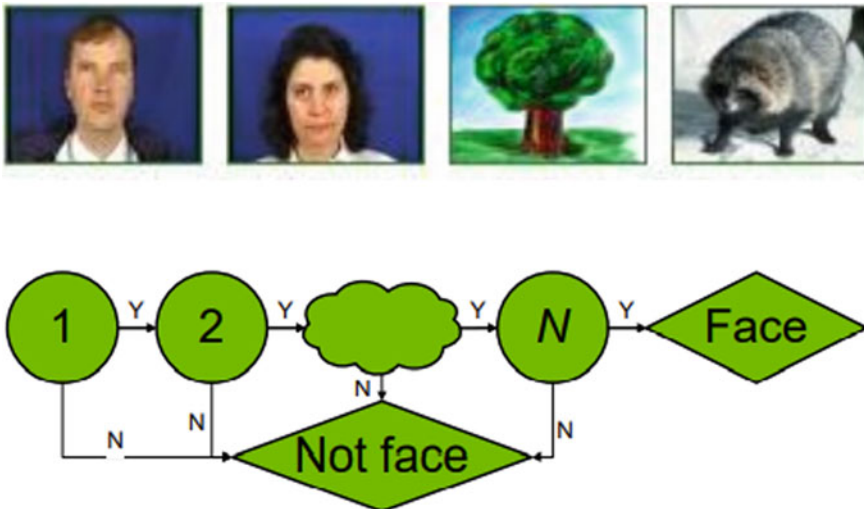


Fig. 8 Multi-stage cascade classifier

4. *Viola–Jones algorithm on CPU-based HPC infrastructure*

Step 1. The identification interaction starts with extricating the essential worth in the picture.

Step 2. The essential worth of the picture will be utilized in the feature such as eye, nose, mouth ear, and chin recognition method utilizing the classifier document.

Step 3. The algorithm gains facial features for the identification procedure and computes the recognition time.

Step 4. Algorithms start finding all faces by marking squares around the face and detection time.

5. *Viola–Jones algorithm on the Hybrid [CPU-GPU] HPC infrastructure*

Step 1. Face recognition starts with extracting the necessary worth from the picture.

Step 2. Then, at that point, the necessary worth of the picture will be utilized in the component recognition process utilizing the classifier document.

Stage 3. The recognition procedure is completed on the CUDA-based GPU.

Stage 4. The recognition procedure happens until the algorithm obtains facial features and computes the recognition time.

Stage 5. The algorithm considers faces presently the image. Presently, time is required to recognize faces.

The real-time face recognition procedure is carried out by utilizing the hybrid [CPU-GPU] HPC infrastructure by using the CUDA library. The hybrid HPC infrastructure the real-time face recognition procedure will take a few phases for face recognition, between others:

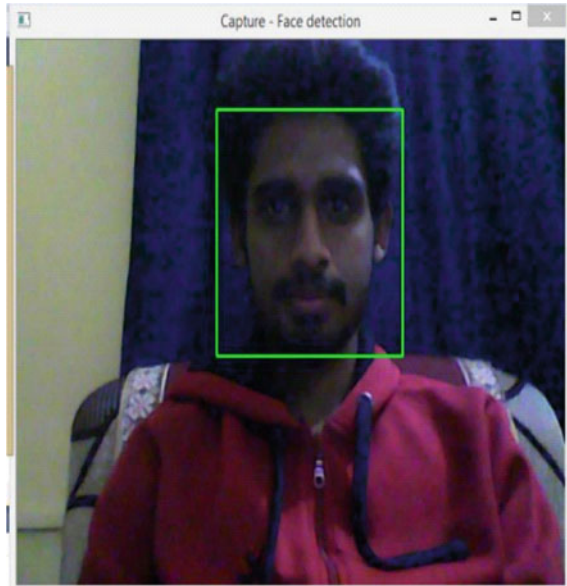
- (a) Freight the image having faces,
- (b) Assign memory on CPU,
- (c) Assign memory on GPU utilizing CUDA Malloc library work,
- (d) Recovering information from the input picture on the host,
- (e) Copying information from host memory into device memory,
- (f) Handling the device memory in the piece is to track equally processing as per the frameworks, squares, and strings required for equal cycles,
- (g) Save the result information in host memory,
- (h) Freeing device memory for different tasks

5 Experimental Setup and Performance Analysis

Experiments were conducted for validating the accelerated real-time face detection using cascade classifier on hybrid [CPU-GPU] HPC infrastructure, and the experiments were conducted on 2 different clusters:

Nodes	Nodes names	No. of cores/socket	Processors	Clock speed (GHz)	RAM (GB)	PCIe	Operating system	Ethernet (Mbps)	MPI library	Compiler
2	Dell precision T3610	8	Intel Xeon E5-1600 CPU	3.70	128	3.0x	Fedora 24	100	MPIC H2-1.2	GCC version 4.4.7
		227	Two-NVIDIA Quadro K2000-GPUs	2000	2	3.0x				nvcc version 5.0
3	Dell precision R5500	4	Intel Xeon 5600 CPU	2.40	192	3.0x	Fedora 24	100	MPIC H2-1.2	GCC version 4.4.7
		192	Two-NVIDIA Quadro 2000	1250	1	3.0x				nvcc version 5.0
1	Power edge R720	24	Intel Xeon E5-2620 CPU [server]	2	768	3.0x	Fedora 24	100	MPIC H2-1.2	GCC version 4.4.7
		448	NVIDIA TESLA M 2075	1.15	6	3.0x				nvcc version 5.0

Fig. 9 Real-time single face detecting



5.1 Experimental Results

In the experiment conducted, we have provided two options mainly the training phase and the testing phase. In the training phase, we fed the computer with several faces of human and non-human faces. In the testing phase, we executed our code on CPU-based HPC infrastructure and hybrid [CPU-GPU]-based infrastructure formulated our results. We also tried testing our code of face detection on animal faces and computed results; all the snapshots are listed in Figs. 9 and 10.

The execution on hybrid [CPU-GPU]-based infrastructure utilizes the OpenCV + CUDA-based cascade classifiers with roughly 2430 Haar-like classifiers for front-facing faces. Tests dependent live video take care of 24 fps from the camera were led, with a portion of the recognition outcomes displayed in Figs. 9 and 10.

Proposed real-time face detection was tested on a hybrid [CPU-GPU] HPC infrastructure for their accelerated performance (Figs. 11, 12 and 13).

A comparative study between the CPU-based HPC infrastructure and the proposed hybrid [CPU-GPU] HPC infrastructure acceleration in fps is listed in Table 1. The greatest fps execution of our projected execution is 37.9871 frames per second in Hybrid [OpenCV + CUDA] HPC infrastructure, in the examination of the CPU-based HPC infrastructure execution of 2.71 frames per seconds.

Figure 14 shows the FPS performance comparison of the proposed hybrid [CPU-GPU] infrastructure against CPU-based HPC infrastructure for varying resolutions, where X-axis shows the different resolutions and Y-axis shows the frames per second. During the experiments on hybrid [CPU-GPU] infrastructure for lower resolutions

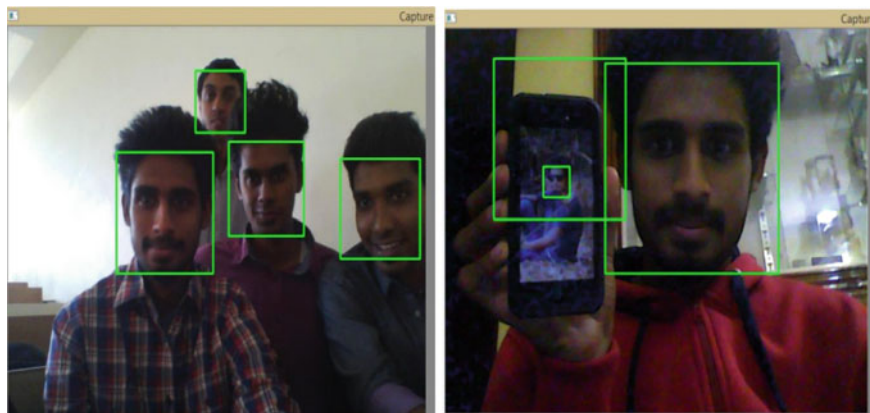


Fig. 10 Real-time multi-face detecting

Fig. 11 Not detecting a face in real time

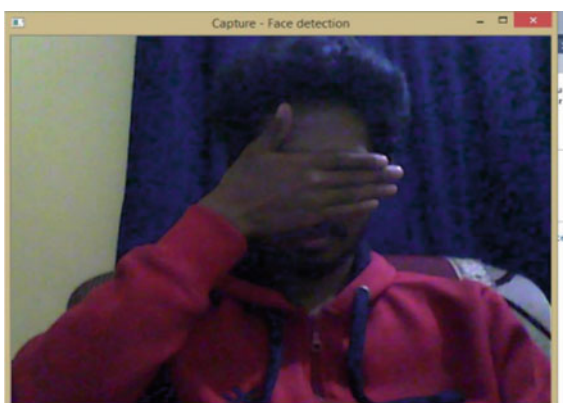


Fig. 12 Animal face in real time



Fig. 13 Testing animal face and human face together in real time

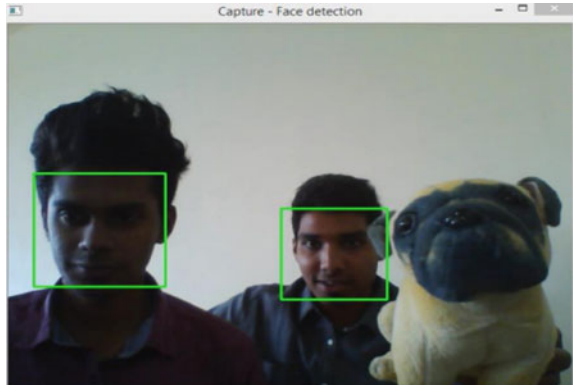


Table 1 Performance comparison between CPU-HPC infrastructure and hybrid HPC infrastructure

Resolution	Frames per seconds (FPS)			Average detection time (μ s)		
	CPU-based HPC infrastructure	Hybrid [CPU-GPU] HPC infrastructure	Speedup	CPU-based HPC infrastructure	Hybrid [CPU-GPU] HPC infrastructure	Speedup
640 \times 480	1.70	43.78	25.75294118	0.956236	0.024378	0.025494
720 \times 480	1.73	42.67	24.66473988	0.995612	0.027878	0.028001
800 \times 600	1.75	40.32	23.04	1.061782	0.787878	0.742034
1024 \times 768	1.77	39.00	22.03389831	1.218787	0.987999	0.810641
1152 \times 864	1.97	38.98	19.78680203	1.549898	1.029922	0.66451
1280 \times 768	2.13	38.75	18.19248826	1.679899	1.217668	0.724846
1366 \times 768	2.36	38.71	16.40254237	1.787897	1.565466	0.875591
1440 \times 900	2.54	38.54	15.17322835	1.876755	1.564543	0.833643
1680 \times 1050	2.59	38.21	14.75289575	1.929870	1.678799	0.869903
1920 \times 1200	2.67	38.01	14.23595506	1.967576	1.769789	0.899477
2048 \times 1536	2.71	37.98	14.01476015	1.887672	1.178923	0.624538

640 \times 480, we have observed a speedup of 25.75 frames per second against CPU-based HPC infrastructure. Similarly, for higher resolutions 2048 \times 1536 experiments, we have noticed a speedup of 14.014 frames per second against CPU-based HPC infrastructure.

Figure 15 comparison of real-time face detection time on the proposed hybrid [CPU-GPU] infrastructure against CPU-based HPC infrastructure for varying resolutions, where X-axis shows the different resolutions and Y-axis shows the average detection time. During the experiments on hybrid [CPU-GPU] infrastructure for lower resolutions 640 \times 480, we have observed that detection time speed up of 0.025494 computation time when compared against CPU-based HPC infrastructure. Similarly, for higher resolutions 2048 \times 1536 experiments, we have noticed

Fig. 14 Performance comparison between CPU-HPC and hybrid HPC in real time

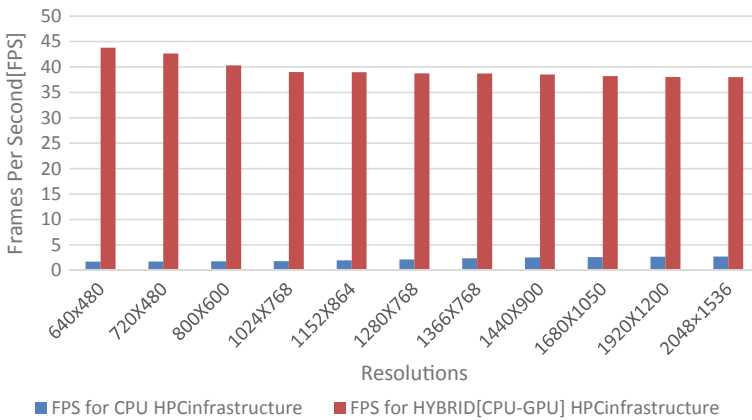
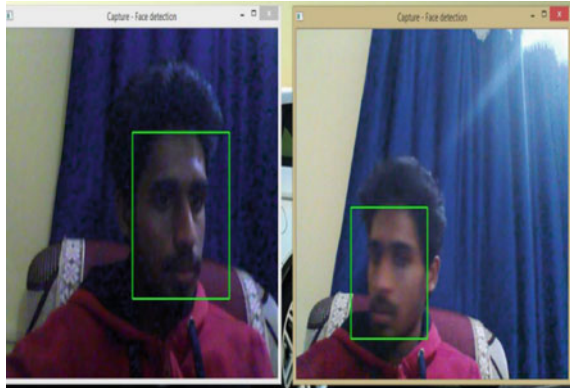


Fig. 15 FPS performance comparison of the proposed hybrid [CPU-GPU] infrastructure against CPU-based HPC infrastructure

a speedup of 0.624538 frames per second against CPU-based HPC infrastructure (Fig. 16).

6 Conclusion

The performance of real-time face detection is a challenging issue in various domains. To report this challenge, in this paper, we have used hybrid [CPU-GPU] infrastructure with Viola and Jones algorithm with cascade classifier. And from the results, it was proven that its average detection time and frames per second of real-time face detection is better than the only CPU-based HPC infrastructure. Exploratory results showed that the proposed hybrid infrastructure achieved an average of 18.91 frames

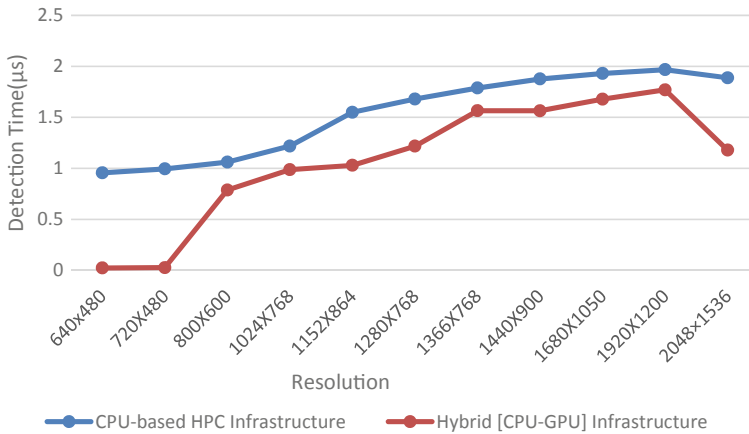


Fig. 16 Comparison of real-time face detection time on the proposed hybrid [CPU-GPU] infrastructure against CPU-based HPC infrastructure

per second and average detection time up to 0.64 times computational speed up based on varying resolutions from 640×480 to 2048×1536 when compared against CPU-based HPC infrastructure.

References

- Jain V, Patel D (2016) A GPU based implementation of the robust face detection system. *Procedia Comput Sci* 87:156–163
- Jussi M (2013) GPU accelerated face detection. Department of Computer Science and Engineering, University of Oulu, Oulu, Finland University of Oulu, 2013 ebook. <http://urn.fi/URN:NBN:fi:oulu-201303181103>
- Mohanty A, Suda N, Kim M, Vrudhula S, Seo JS, Cao Y (2016) High-performance face detection with CPU-FPGA acceleration. *IEEE Int Symp Circ Syst (ISCAS) 2016*:117–120
- Daschoudhary RN, Tripathy R (2014) Real-time face detection and tracking using Haar classifier. In: *Proceedings of SARC-IRF international conference*, 12th Apr 2014, New Delhi, India, ISBN: 978-93-84209-03-2
- Kurniawan B, Adji TB, Setiawan NA (2015) Analisis Perbandingan Komputasi GPU dengan CUDA dan Komputasi CPU untuk Image dan video processing. In: *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, vol 1, no 1
- Wai AWY, Tahir SM, Chang YC (2015) GPU acceleration of real-time Viola-Jones face detection. In: *2015 IEEE international conference on control system, computing and engineering (ICCSCE)*, pp 183–188
- Chandrashekhar BN, Sanjay HA, Deepashree KL, Ranjith N (2018) Implementation of image inpainting using OpenCV and CUDA on CPU-GPU environment. In: *International conference advances in computing, control & telecommunication technologies (ACT-2018)*. Academic Press Publishers
- Kane SN, Mishra A, Gaur A (2014) International conference on recent trends in physics (ICRTP 2014). *J Phys Conf Ser* 534(1):11001
- Patterson DA, Hennessy JL (2020) *Computer organization and design MIPS edition: the hardware/software interface*. Morgan Kaufmann, Burlington

10. Sharma S, Agrawal R, Srivastava S, Singh D (2017) Review of human detection techniques in night vision, pp 2216–2220. <https://doi.org/10.1109/WiSPNET.2017.8300153>
11. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition, CVPR 2001, vol 1, p I-I
12. Lescano GE, Santana Mansilla P, Costaguta R (2017) Analysis of a GPU implementation of Viola-Jones' algorithm for features selection. *J Comput Sci Technol* 17(1):68–73
13. Jia H, Zhang Y, Wang W, Xu J (2012) Accelerating viola-jones face detection algorithm on GPUs. In: 2012 IEEE 14th international conference on high-performance computing and communication & 2012 IEEE 9th international conference on embedded software and systems, pp 396–403
14. Chandrashekhar BN, Sanjay HA (2019) Performance framework for HPC applications on homogeneous computing platform. *Int J Image Gr Signal Process (IJIGSP)* 11(8):28–39. <https://doi.org/10.5815/ijigsp.2019.08.03>
15. Marineau R (2018) Parallel implementation of facial detection using graphics processing units
16. Putro MD, Adji TB, Winduratna B (2012) Sistem Deteksi Wajah dengan Menggunakan Metode Viola-Jones
17. Patel R, Vajani I (2015) Face detection on a parallel platform using CUDA technology. *Natl J Syst Inf Technol* 8(1):17
18. Rahmad C, Asmara RA, Putra DRH, Dharma I, Darmono H, Muhiqqin I (2020) Comparison of Viola-Jones Haar cascade classifier and histogram of oriented gradients (HOG) for face detection. *IOP Conf Ser: Mater Sci Eng* 732(1):12038



Jagadevi N. Kalshetty , V. Venkata Sree Harsha ,
and Pushpesh Prashanth 

1 Introduction

Though the whole world is moving forward and is trying to digitize in whatever field possible medicine/health care is no exception. Even though most of the government hospitals are still following the traditional protocol of pen and paper-based collection of medical data, hospitals in cities, basically private hospitals are coming up with what is called an electronic health record system. In our course of collecting information for this project, we have tried 2–3 open-source EHR tools by ourselves. EHR is very good at doing what we used to do using pen and paper but a bit more time efficiently and smartly. But again, since we are dealing with medical sciences, our target is not just to save papers, our target must be, to use such a huge amount of data in the most efficient way possible. The biggest problem in today's world of medical sciences is not the scarcity of resources, science and technologies have ensured a large amount of digital machinery that aids toward more efficient medical procedures, and the problem is the lack of knowledge, lack of communication. Scientists do not communicate directly with the patients, doctors barely communicate with other doctors, researchers are mostly privately sponsored, they do not communicate with the government, and this keeps on and on. Technology is advancing day by day giving us new ways to share and collect information and use it, but still, the idea of business and profit is what lets us have this barrier in communication which otherwise would have helped us solve very mysteries set of condition that are there in our world but remain unreported or unidentified due to this lack of communication. Also, every time one moves from one location to another has to see a different doctor or comes in an emergency case, or any other reason, the hospital or the doctor has to take all the information from the start which can be a very tedious task. Also, not everyone in the world is capable

J. N. Kalshetty (✉) · V. Venkata Sree Harsha · P. Prashanth
Nitte Meenakshi Institute of Technology, Bangalore, India
e-mail: Jagadevi.n.kalshetty@nmit.ac.in

to remember every single medical terminology regarding their diseases, conditions, allergies, etc. which again makes the new information incomplete which leads to redundancy and delay in diagnosis.

2 Problem Statement

In this project, we are going to focus on the use of technologies in data analytics in the field of Medical Science. It may seem to be a pile of data of billions of people, but if we can see the bigger picture, it will have a huge impact on Medical Science and the future of mankind. We have talked to many friends pursuing medical science and tried to understand their approach and found out a few major problems.

1. An ill-informed, uneducated, or unconscious patient is not a reliable source of medical history; hence, it affects a doctor's approach. For example, a patient was admitted from the emergency room because of acute respiratory failure. He could not verbalize his medical history, because all of his focus and efforts were upon getting enough air. In terms of his medical history, his medical team was flying blind.

The patient was a 48-year-old truck driver from out of town. No family was at the bedside, and no medical records were available. What medications was he on? What allergies did he have? Given all of the unknowns, the patient received the generic, one size fits all treatment for acute hypoxic respiratory failure. Over the next several hours, his heart rate gradually increased into the 130 s. Most likely, he was developing sepsis from an acute infection which was the cause of his initial breathing difficulties. In response, he was given antibiotics and intravenous fluids, again the generic treatment for what was the most likely cause of his condition. But it turns out his symptoms were due to something else entirely that would not have been missed if old medical records had been available. The patient just had a severe exacerbation of his chronic obstructive pulmonary disease. He did not have sepsis at all. His increased heart rate was due to beta-blocker withdrawal from not getting his routine nightly dose of metoprolol. He was eventually discharged from the hospital in good condition, back at his baseline. His hospitalization, however, was prolonged by a full day, and he received unnecessary antibiotics all because nobody knew he was on a beta-blocker. His old medical records were in Lucknow, locked up safe, and secure at his home. We, however, were in Delhi. While his medical records were secure in Lucknow, they were not useful.

2. Not only is there bad communication between the patient and the doctors, but also the medical records maintained by any particular hospital remain confined to itself resulting in a lack of network between hospitals and hence bad communication and sharing of resources between them. Before the treatment of any disease, every patient is a live human experiment of that disease in a particular geography, sub-population specific genetic makeup, etc., but due to lack of connectivity between hospitals, we lost all of these important data.

3. Adverse drug reaction (ADR)—India’s contribution to reporting ADR is negligible which is unacceptable. Currently, India’s contribution to the global safety database is near to 3%, and the completeness score is 0.93 out of 1. Given the two statistics, it is very low compared to the population that we have; it is almost equivalent to non-existent contribution [2].
4. Vague Analysis—We are incapable of doing any analysis because there is no large-scale and centralized epidemiological data. Indian population is very diverse geographically, ethnically, culturally and demands extensive research.

The above points lead to a conclusion:

- (a) Incomplete data.
- (b) Data is confined/restricted.
- (c) Data is not centralized.

Example—Consider a case: Acute pancreatitis is a severe disease with considerable morbidity and mortality. Gallstones and alcohol abuse are the most frequent causes (75% of patients). Other well-known causes are hyperlipidemia, hypercalcemia, abdominal surgery, and drugs. In 10–40% of patients, however, no cause is identified after initial diagnosis which is acute idiopathic pancreatitis. It is very important to identify the cause in these 10–40% patients as the rate of recurrence is high.

Now consider if we have well-organized and centralized data of every patient, then can’t we do the retrospective study of that medical case and similar other cases to formulate a hypothesis so as to determine its etiology and accordingly the best treatment strategy.

3 Proposed System

We are going to make use of a NoSQL database for storing patients’ medical records since most of their records will be unstructured. The idea is to make the medical record more objective as possible rather than storing PDFs and PPTs the data will be broken down and will be stored objectively. This objective data will make each and every medical detail atomic; hence, it will be easy to manage and process data. Now, this database will be connected to the two different applications for separating two types of views.

We are going to develop two applications:

1. A doctor’s interface/perspective.
2. A web-application for patients.

3.1 *Doctor's End*

This application will be for reading and updating the patient's medical records in the database.

- The hospitals, pathological laboratories, and clinics will be provided with the software where they first have to register the doctors working there by providing their license details. After this process, the doctors will be provided login credentials for authentication purposes.
- Only the registered doctors will be authorized to access a registration form, and doctors will ask the patient for his/her government id and will register the patient on the portal when a patient visit for the very first time. We will match the patient's credentials with the government database.
- Once the patient is registered, the doctor will have the responsibility to fill out the basic details like the patient's weight, height, age, blood group, etc. Doctors can also add to the past medical records of the patient if it is available. These details will be added to our centralized database.
- The patient will be assigned a collection in the database which can be accessed by the doctors under the following rights:
- **Read Access:** All the valid licensed doctors will have this access; the doctor will provide his/her login credentials, then he will be authenticated and will be directed to the portal where he can give the patient id to check his/her medical history.
- **Update Access:** This is a special access right which will be used whenever a doctor has to update the patient's medical records, and for this purpose, the doctors will require consent from the patient which can be done using the OTP authorization process. It is more of a handshake rather than an authentication which means that the doctor authenticates the patient and the patient authenticates the doctor.

These rights help in preserving confidentiality.

Confidentiality: Since only the authorized doctor and patient will have read access rights.

- Each and every read or update transaction's details such as its timestamp and the details of the person accessing it will be stored in our database for security purposes.
- Now since all these records will be stored in a digital database, it will be available to all the registered hospitals and corresponding registered doctors.
- When the patient makes the next visit to any registered hospital, the doctor assigned to the patient can access and consequently update the patient's record (Fig. 1).

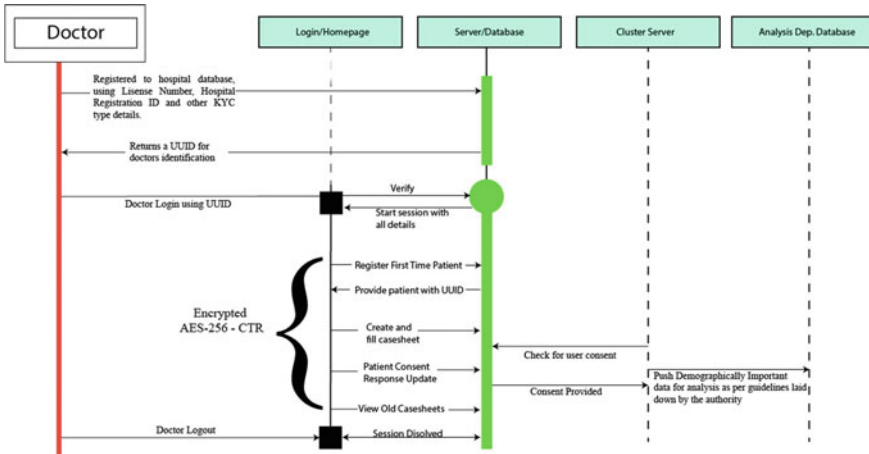


Fig. 1 Sequence diagram depicting doctor’s end

3.2 Patient’s End

- It is also necessary that the patient can keep track of his/her medical records; for this purpose, we will create a web application which will be directly connected to the centralized database.
- The patients can authorize themselves by giving their identification number and the OTP sent to their registered mobile number and finally can check their medical record but won’t be able to update it.

It will be the doctor’s responsibility to make the patient aware of the analysis program and based on their decision, on getting consent, a copy of the relevant data will be pushed for analytics.

Anonymity: of the patient will be preserved which means that the identity of the person whose medical record is chosen for the analysis process will never be disclosed.

Cloud

So as to the problem of storage and accessibility according to our research for this project, we think the use of cloud would be the best. Cloud is one of the tried and tested technologies that guarantee 24 × 7 access to data from anywhere using any device. They way any doctor can access any patient’s data at any time, given that patient has given consent to do so. This can be further be ensured through multi-factor authentication and authorization using patient and doctor unique id being logged separately at every transaction.

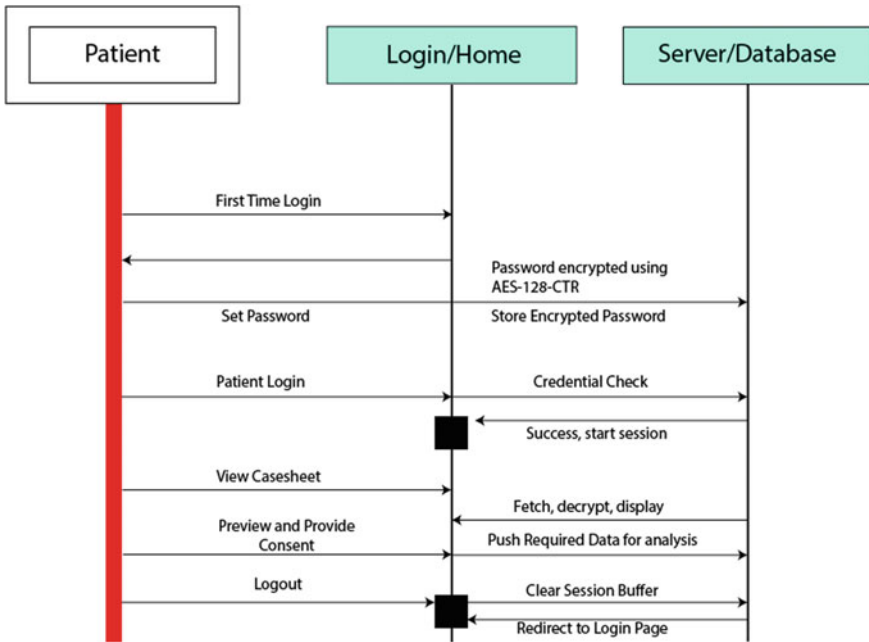


Fig. 2 Sequence diagram depicting patient’s end

Big Data Analytics

We will ensure that every data in the medical record will be objective and will include a person’s detailed family history, medical history since birth, records of all the doctor’s visits, socioeconomic background, prescription, laboratory tests, disease, previous medications, allergies, drug interactions, vaccinations, etc. Also, our doctors and hospitals will be connected to this database.

Since we have every version of a patient’s medical record at a single place hence by forming a proper community of data scientists, we can visualize these medical records for demographic analysis of diseases and various other statistical analyses that the community of doctors might require. Later in today’s day and era with big data and deep learning algorithms, using both supervised and unsupervised complex relationships, which were unknown before, might come in light. Predictive analysis can be used, which builds models to forecast any particular disease (Fig. 2).

4 Implementation

The doctor will login using his credentials and creates a session. The doctor gets three options, to register a patient, create a case sheet, and view old case sheets. During the initial phase of implementation doctor can register a patient to the system

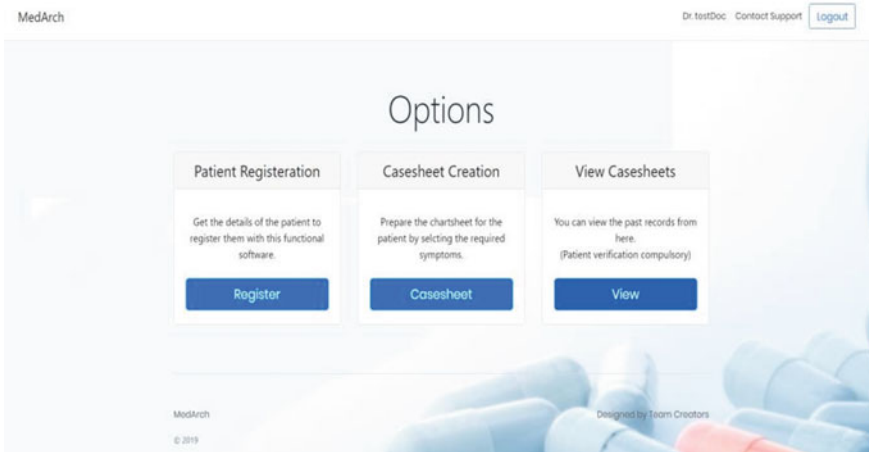


Fig. 3 Doctors' interface

which will generate a unique id for the patient, later patient can set password for his account at first login attempt through OTP verification (Figs. 3, 4, 5 and 6).

This data can only be viewed by doctors and patients through their respective portal. The data is encrypted using patient's id as key. Below capture depicts the view (Fig. 7).

Finally, once the data is stored and consent is given by the patient, some specific attributes are taken and transferred to another server where different analyses are performed. For example, we showed age versus cataract plot and realized that cataract mostly occurs in case of old peoples, but still there are exceptions (the data below is dummy data created from our domain knowledge) (Figs. 8 and 9).

Similarly, we did a demographic analysis finding the cities that are worst hit by diseases coded as 0–6.

With these small examples, we have showcased how a system like this implemented in at least government run hospital can change the picture of healthcare status of our country, which at the current moment is poor and lack behind a lot in healthcare infrastructure.

5 Conclusion

Returning back to the example, we put forth earlier in the problem statement, because of such large datasets depositing at one place diseases like acute idiopathic pancreatitis will be highlighted, and we might find the unknown reasons of it happening. Analysis like these will not only help in increasing the existing knowledge but will also enhance the doctor's approach while dealing with such cases. This system can help in increasing the efficiency as well as the standard of diagnosis in our country.

The screenshot shows a web-based medical form titled "Related Info Continued". At the top right, there is a search bar. Below the title, there are several tabs: "General", "Pain", "Visual Disturbance", "Past History", "Related Information", "Related Information 2" (which is selected and highlighted in blue), and "Drug Usage". Underneath these tabs, there are more categories: "Exposure and Family History", "Local Examination", "Lids", "Iris", and "Fundoscopy".

The form contains the following sections and fields:

- Red Eye:** A checked checkbox with a text input field containing "2 hours".
- Lacrimation:** A checked checkbox with a text input field containing "15-20minutes".
- Discharge From Eye:** A section with several radio and checkbox options:
 - Clear
 - Purulent
 - Mucopurulent
 - Drooping of Eyelid
 - Limited Movement of Eye
 - Difficulty in dark adaptation
 - Spots on eye white
 - White Reflex of Pupil
 - Eye Rubbing
 - Lump on Eyelid
 - Fever
 - Associated Factors:**
 - Chills
 - Rigor

- Malaise:** A text input field containing "2-3hrs".
- Other symptoms:** A list of checkboxes:
- Frequent Changes of Glasses
- Visual Hallucination
- Coloured Vision(Chromatopsia)
- Anisocoria:** A text input field containing "Duration if exist".

At the bottom left of the form, there is a blue "Next" button.

Fig. 4 Forms, likewise there are 12 for ophthalmology

While treating the above disease using traditional methods, since the strength of the network between hospitals is poor, hence the information or the knowledge gained by the doctors of Lucknow is unknown to the doctors working in Bengaluru and vice-versa. Hence, the doctors on a large scale are unable to share their experience and knowledge with each other; therefore, every time a new such case is recorded, they have to follow their own algorithm rather than following a hidden algorithm that has a more success rate and is used by other doctors at other places around the country.

Key	Value
<ul style="list-style-type: none"> ▼ (1) IMPORTANTS <ul style="list-style-type: none"> 🔍 _id 🔍 docname 🔍 docid 🔍 date 🔍 time > (2) GENERAL INFORMATION > (3) COMPLAINTS > (4) VITALS ▼ (5) SITE OF PAIN <ul style="list-style-type: none"> 🔍 _id 🔍 eyeache 🔍 color 🔍 onset 🔍 character > (6) ASSOCIATED SYMPTOMS > (7) TIMING > (8) VISUAL DISTURBANCE ▼ (9) CHARACTER <ul style="list-style-type: none"> 🔍 _id 🔍 visionloss > (10) VISASSOCIATEDSYMPTOMS > (11) PROGRESSION > (12) PAST HISTORY > (13) PAST HISTORY OF EYE ▼ (14) RHYTHMICAL <ul style="list-style-type: none"> 🔍 _id 🔍 Type > (15) SWELLING ▼ (16) RELASSOCIATEDSYMPTOMS <ul style="list-style-type: none"> 🔍 _id 	(5 fields) IMPORTANTS l28mrRtvJ7l+AMB+wymx7/wzMRYGj9E= GGYhWrr17ddinTCh/we9wZgFVN1EBg= CvIMj9bePIEc3/OHfKLM43468ZmmE/Eg PHCBwksMg75OpQUTJQN1sgmas5P+vNz1JK8= (9 fields) (3 fields) (10 fields) (5 fields) SITE OF PAIN b6nzZ5qUbjtzRBzTW+TWlxnDRSHNyDDmvQ= gDV30cP3ZJAGTg4F75MAAIF6g= oKoa/BgcE/ygInSCCVIMNISEZEIGR8= w3jbrRC2z51a+Jta0hu0vmzITVBQ= (3 fields) (4 fields) (3 fields) (2 fields) CHARACTER 7rcnAn4il+hQhxAydydFwOoRMO4= (3 fields) (2 fields) (5 fields) (3 fields) (2 fields) RHYTHMICAL /q3kCwiqAEDu7VvDrbKwELjixUdZJxoS (3 fields) (3 fields) RELASSOCIATEDSYMPTOMS

Fig. 5 Encrypted database

🔍 _id	ObjectId("5e930e2c80c7393774007196")	ObjectId
🔍 p_id	1001002	String
🔍 doc_id	999999	String
🔍 casesheet	c1_1001002	String
🔍 hospital_id	LCK10210	String
🔍 date	20-04-12	String
🔍 time	06:18:43pm	String

Fig. 6 Doctor access log

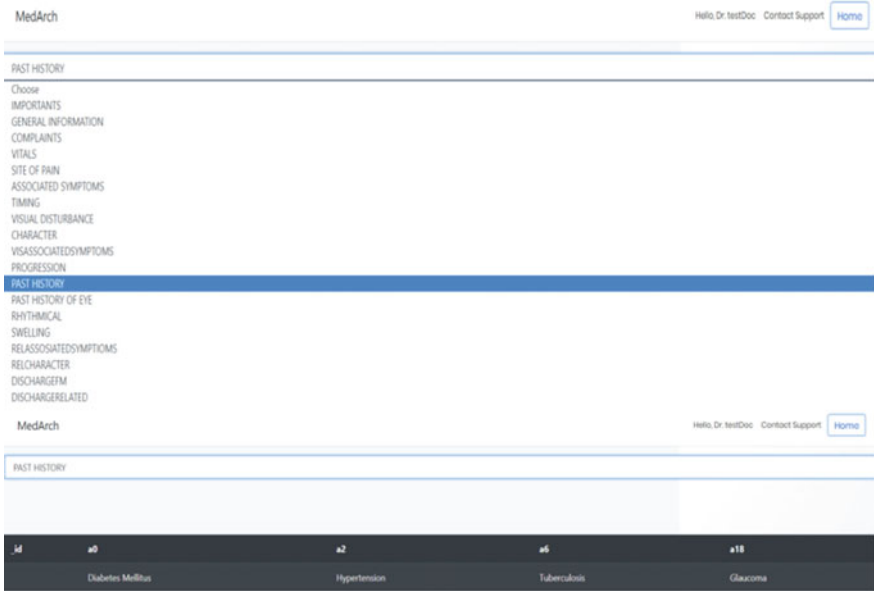


Fig. 7 View portal

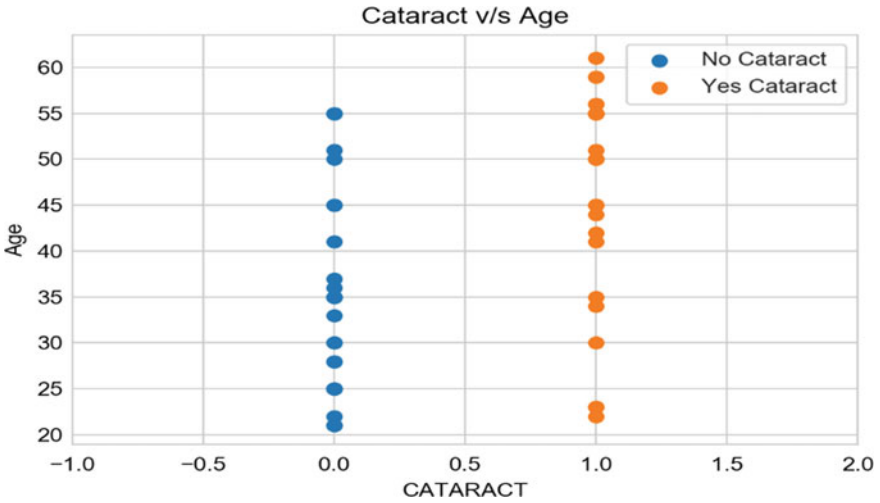


Fig. 8 Age versus cataract

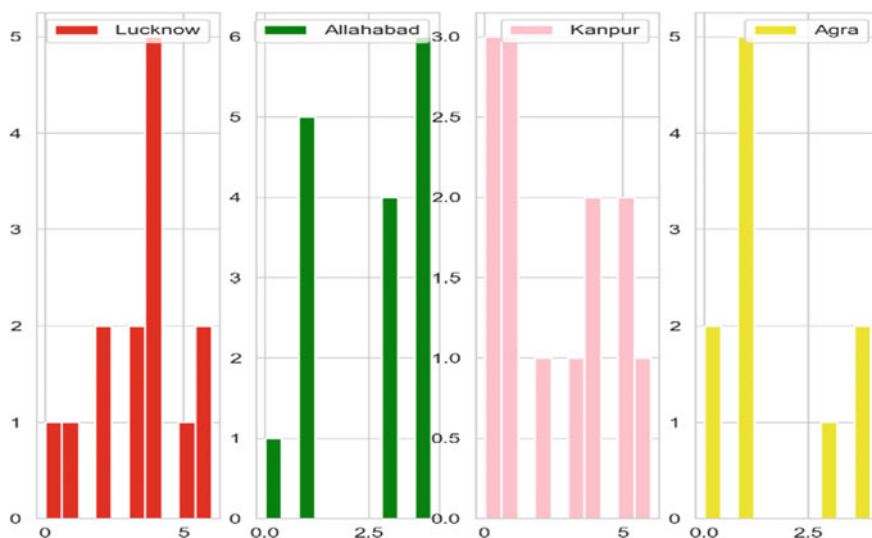


Fig. 9 Diseases versus count based on cities

References

1. Fisher N, Lack of communication is ruining healthcare. *Forbes*
2. Kalaiselvan V, Thota P, Singh GN (2016) Pharmacovigilance programme of India: recent developments and future perspectives. *Indian J Pharmacol* 48(6):624–628. <https://doi.org/10.4103/0253-7613.194855>
3. World Health Organization (2002) Constitution of the World Health Organization as adopted by the International Health Conference, New York, 19–22 June 1946; signed on 22 July 1946 by the representatives of 61 States (Official Records of the World Health Organization, no 2, p 100) and entered into force on 7 April 1948. In: Grad FP (ed) The preamble of the constitution of the World Health Organization. *Bulletin of the World Health Organization*, vol 80, no 12, p 982
4. Kumar R, Pal R (2018) A call for paradigm shift in public health discourse. *J Family Med Health Care* 7(5):841–844
5. van Brummelen SE, Venneman NG, van Erpecum KJ, van Berge-Henegouwen GP (2003) Acute idiopathic pancreatitis: does it really exist or is it a myth? *Scand J Gastroenterol Suppl* 239:117–122. <https://doi.org/10.1080/008559203100027>
6. Zhu H, Hou M (2018) Research on an electronic medical record system based on the internet. In: 2018 2nd international conference on data science and business analytics (ICDSBA), Changsha, 2018, pp 537–540. <https://doi.org/10.1109/ICDSBA.2018.00106>
7. Pirmohamed M, James S, Meakin S, Green C, Scott AK, Walley TJ et al (2004) Adverse drug reactions as cause of admission to hospital: prospective analysis of 18,820 patients. *BMJ* 329:15–19
8. Davies EC, Green CF, Taylor S, Williamson PR, Mottram DR, Pirmohamed M (2009) Adverse drug reactions in hospital in-patients: a prospective analysis of 3695 patient-episodes. *PLoS ONE* 4:e4439
9. Bates DW, Spell N, Cullen DJ, Burdick E, Laird N, Petersen LA et al (1997) The costs of adverse drug events in hospitalized patients. Adverse drug events prevention study group. *JAMA* 277:307–311

10. Wu TY, Jen MH, Bottle A, Molokhia M, Aylin P, Bell D et al (2010) Ten-year trends in hospital admissions for adverse drug reactions in England 1999–2009. *J R Soc Med* 103:239–250
11. Runciman WB, Roughead EE, Semple SJ, Adams RJ (2003) Adverse drug events and medication errors in Australia. *Int J Qual Health Care* 15(Suppl 1):i49–59
12. Moore N, Lecointre D, Noblet C, Mabile M (1998) Frequency and cost of serious adverse drug reactions in a department of general medicine. *Br J Clin Pharmacol* 45:301–308
13. Patel KJ, Kedia MS, Bajpai D, Mehta SS, Kshirsagar NA, Gogtay NJ (2007) Evaluation of the prevalence and economic burden of adverse drug reactions presenting to the medical emergency department of a tertiary referral centre: a prospective study. *BMC Clin Pharmacol* 7:8
14. Khan FA, Nizamuddin S, Najmul H, Mishra H (2013) A prospective study on prevalence of adverse drug reactions due to antibiotics usage in otolaryngology department of a tertiary care hospital in North India. *Int J Basic Clin Pharmacol* 2:548–553
15. Wanbin W (2011) Design and implementation of electronic medical record management system. *Chin Comput Commun* 7:26–28
16. Bo L (2017) Design and research of hospital electronic medical record management system based on B/S architecture. *Electron Des Eng* 25(5):46–49
17. Lazarou J, Pomeranz BH, Corey PN (1998) Incidence of adverse drug reactions in hospitalized patients. A meta-analysis of prospective studies. *JAMA* 279:1200–1205
18. Wester K, Jonnson AK, Sigset O, Druid H, Hagg S (2008) Incidence of fatal adverse drug reactions: a population based study. *Br J Clin Pharmacol* 65:573–579
19. Pirmohamed M, James S, Meakin S, Green C, Scott AK et al (2004) Adverse drug reactions as a cause of admission to hospital: prospective analysis of 18,820 patients. *BMJ* 329:15–19
20. Winterstein AG, Sauer BC, Hepler CD, Poole C (2002) Preventable drug-related hospital admissions. *Ann Pharmacother* 36:1238–1248

Pneumonia Prediction Using Deep Learning



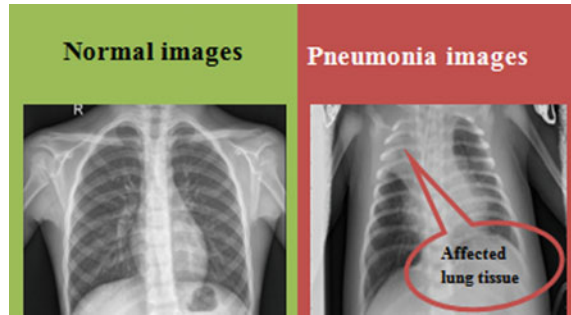
B. G. Mamatha Bai and V. Meghana

1 Introduction

Pneumonia is an inflammatory disorder of the lung that mainly affects the tiny air sacs recognized as alveoli. As pneumonia is an inflammatory disorder of the lung that mainly affects the tiny air sacs recognized as alveoli. Symptoms typically include a mixture of productive or dry cough, chest pain, fever and breathing difficulties. There is a variable of severity. Pneumonia is generally caused by virus or bacterial infection and is less frequently caused by other microorganisms, certain medicines and circumstances such as autoimmune diseases. Risk factors include other lung diseases such as cystic fibrosis, chronic obstructive pulmonary disease (COPD), asthma, diabetes, heart failure, smoking history, bad cough capacity such as a stroke or a weak immune system. The size of lungs changes with the growth of human being. As the disease is observed in all the ranges of age, it is important to know the structure of lungs in all ages.

Lungs of a healthy person consist of hard tissue evenly present on outer layer. The color of the lungs is evenly present all over the surface. A report released in India Today stated nearly 1.7 million children will die due to pneumonia by 2030 in India. A convolutional neural network in deep learning is a category of profound neural networks, most frequently used for visual imaging analysis. CNNs are multilayer perceptron regularized variants. Multilayer perceptron generally refers to fully linked networks, that is, the next layer connects each neuron in one layer to all neurons. These networks' full connectivity makes them susceptible to information overfitting. Typical forms of regularization include adding to the loss function some sort of weight measurement of magnitude (Fig. 1).

B. G. Mamatha Bai (✉) · V. Meghana
Department of CSE, Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: mamathamane@gmail.com

Fig. 1 Lungs image

A convolutional neural network comprises of a layer of input and output, as well as several concealed parts. There are many CNN architectures out of which we are using VGG 16-layer architecture as the number of parameters obtained is more. The networks used are deeper, and it yields good accuracy and error rate.

2 Related Work

Nogues et al. [1] have used three significant but earlier understood variables in this article to use profound convolutionary neural networks for computer-aided identification issues. They have investigated and assessed various CNN architectures for the first time. The designs researched comprise parameters of 5 thousand to 160 million and differ in number of layers, assess the efficiency impact of the scale of the dataset and the spatial picture background. Liet al. [2] give a technique relying on 1D neural network convolution which is suggested in this document to classify ECG signals. In addition to the input layer and the output layer, the proposed CNN model consists of five layers, that is, two convolution layers, two downsampling layers and a full connection layer, extracting the effective features from the original data and automatically classifying the features.

Albawi et al. [3] have given CNN's efficiency in machine learning issues that are outstanding. In particular, applications dealing with image data, such as the largest image classification dataset, computer vision and the results achieved in natural language processing were very amazing. In this document, we will clarify and describe all the CNN-related aspects and significant problems and how they operate. They also specify the parameters that affect the effectiveness of CNN.

Jmour et al. [4], this article discusses a teaching strategy centered on instruction for a traffic sign classification scheme in convolutionary neural networks (CNN). It also provides the outcomes of the preliminary evaluation of using this CNN to know characteristics and lists the assignment of RGB-D pictures. To determine the suitable architecture, we investigate the transfer teaching method called the "good tuning method" of reusing layers practiced on the ImageNet dataset to provide an alternative for a fresh information set's four-class classification. Bhandare et al.

[5], they provide an extensive overview of CNN's applications in computer vision and natural language processing in this article. We delineate how CNN is used in computer vision, primarily in face recognition, scene labeling, image classification and recognition of movement, measurement of human posture and evaluation of documents. We also explain how CNN is used for natural language processing in the domain of speech recognition and text classification.

Ding et al. [6], their purpose of this article is to provide data on a clearer comprehension of the activation function's growth, attributions and suitable decisions. Recent developments in deep learning methodologies have significantly increased speech recognition scheme efficiency. Much advancement has been made in the creation of fresh activation functions and appropriate initializations recently. In this research, for the assignment of speech recognition, we explored the efficacy of different activation functions. Krishna et al. [7] recent developments in deep learning methodologies have significantly increased speech recognition scheme efficiency. Much advancement has been made in the creation of fresh activation functions and appropriate initializations recently. In his research, for the assignment of speech recognition, we explored the efficacy of different activation functions. We regarded various activation functions such as ReLU, LReLU, PReLU, ELU and PELU during the research. During the research, it is noted that the output of ReLU networks is inferior.

Wang et al. [8], their paper describes the records of one patient, the time when the volume of the heart reaches maximum or minimum varies. In transferable teaching, model optimization is crucial. The similarity between our information and the information in which VGG is trained is difficult to quantify. This means that distinct types of convolution layers need to be checked. There is also a trade-off between minimizing the loss of practice and combating overfitting. Regularization power relies on the information. Although end-systolic and end-diastolic models have the same amount of data, there are different choices about the strength of regularization. Kumar et al. [9], cancer is essentially an unusual cell growth. Nowadays, breast cancer is growing rapidly. Breast cancer is one of the most important causes of women's mortality in the globe. It is the world's most common disease among females. Cancer is essentially an unusual cell growth. Nowadays, breast cancer is growing rapidly.

Sasikala et al. [10], their article uses CNN to detect lung cancer based on chest CT pictures. Lung areas are obtained from the CT image in the first stage, and each slice is segmented to get tumors in that region. A sample picture was supplied to the educated model as an input, and the model can say the existence of cancer at this point and identify the cancer place in the sample picture of a lung cancer. Haryanto et al. [11], his purpose of the study is to identify the two cancer statuses using CNN on gland pictures. On this research, the training process for six, eight and ten layers has been exploited. In this architecture, the use of four convolutionary layers and maxpooling affects the time consumption on the automatic removal of features. Type 3 has the best performance in terms of accuracy in classifying histopathological images compared to another type of CNN in our experiments. Guo et al. [12], this method of sampling is equal to filtering fuzzy. The pooling layer has the impact of extracting the secondary function; it can decrease the function maps size and boost

the processing robustness of the function. Usually, it is put between two parts of the convolution. The size of feature maps is determined by the moving step of the kernels in the pooling layer. Average pooling and maxpooling are the typical pooling activities.

Powell et al. [13], their study proposes a new and robust machine learning model based on a convolutionary neural network to classify single cells automatically as either infected or uninfected in thin blood smears on standard microscope slides. The average accuracy of our new 16-layer CNN model is 97.37% in a tenfold cross-validation based on 27,578 single-cell pictures. Only 91.99% of the same images are achieved by a transfer learning model. Antinl et al. [14], they have to train a model using the dataset out below to assist doctors make X-rays chest pneumonia diagnoses. Chest X-ray pneumonia alone is a challenging job requiring understanding of disease pathology and human anatomy. From the inspection of the dataset, it is evident that it poses a difficult issue: The ribs often obscure regions of concern, and other illnesses appear visually comparable to pneumonia in the dataset [15, 16].

Yu et al. [17], the proposed algorithm he used combines unsupervised features from the saliency map with supervised features from convolutionary neural networks, which are fed to an SVM to automatically detect high-quality retinal images versus poor quality. On a large retinal image dataset, we demonstrate the superior performance of our proposed algorithm and the method could achieve greater accuracy than other methods. Krismono et al. [18], classification of the retinal eye picture is an exciting computer vision issue with broad medical apps. For example, understanding retinal image is very important for ophthalmologists to evaluate eye diseases such as glaucoma and hypertension; if left untreated, visual impairment and blindness can result from these diseases. Cross-validation utilizes leave-one-out method in these studies.

Liu et al. [19], they used 16-layer and 19-layer mode of Oxford, however, even deeper GoogleNet has verified that depth is the most critical factor leading to elevated D-CNN results. On the gigantic ImageNet datasets, however, nearly all the very profound convolutionary neural networks were taught. In reality, we lack marked information, and in the globe, there is only one ImageNet dataset. Sandova et al. [20], their work introduces a new two-stage approach to classifying images with the aim of improving the accuracy of style classification. The suggested strategy splits the input picture into five patches at the first point and uses a profound convolutionary neural network for individual training and classification of each patch. Pasupa et al. [21], the network size is a crucial consideration, and owing to the amount of stacked layers, feature concentrations can be enriched. If the VGG 16 design is expanded in depth, this can lead to a gradient disappearing issue that contributes to a greater learning mistake [22, 23]. The ResNet-50 model, on the other hand, is deeper than the VGG 16 model, but it has a function of identity that can preserve the gradient resulting in a more accurate model.

3 Proposed Model

Pneumonia is a lung disease which is seen in 880,000 children under the age of 2 died in the year 2016. A report released in India Today stated nearly 1.7 million children will die due to pneumonia by 2030 in India. Here, we are helping doctors by making their work easier in predicting the disease through X-ray images. So we are collecting X-ray images and extracting features from them to perform correct diagnosis. Here, we are passing those images into CNN model and making the model to predict by training the model with huge amount of data. By building automatic prediction system, we can make predictions easily so that we can lesser the time constraints instead of wasting doctors time in identification of a disease we can rather use this system. From this now doctors can check more number of patients, patients do not have to wait much longer. As sooner the patient gets treatment, recovery rate also gets faster; by this, we can help more people to live. Proposed flow diagram is shown in Fig. 2.

Here, we are passing those images into CNN model and making the model to predict by training the model with huge amount of data. By building automatic prediction system, we can make predictions easily so that we can lesser the time constraints instead of wasting doctors time in identification of a disease we can rather use this system. From this now doctors can check more number of patients, patients do not have to wait much longer. As sooner the patient gets treatment, recovery rate also gets faster by this we can help more people to live. This PPDL model is divided into two parts as Level 1 and Level 2 to enable two-step verification. In the first part, we are training the model and checking the model accuracy on the new test dataset, whereas in the second part, we are predicting the disease when given a new image. The above model consists of subsequent steps as shown below:

Step 1: Here, we are collecting dataset from online resources. Dataset consists of chest X-ray images which include both pneumonia and healthier people. Further dataset is divided into training and testing dataset.

Step 2: Images are converted into grayscale images. Then, image resizing is applied on each images followed by reshaping them.

Step 3: We are using CNN architectures to perform classification of images. There are different architectures like AlexNet, VGGNet, ResNet, GoogleNet and so on.

Step 4: In this step, we are using 2 architectures VGGNet and AlexNet. VGGNet is made of 16 layers, and AlexNet is of 9 layers.

Step 5: Training images—Images are allowed to pass through the model, and as they pass, all the features are extracted. As longer the model, so many more features are extracted. The process starts learning each images and starts to collecting and learning about it more in a deeper fashion. Training is similar to both architectures.

Step 6: Testing images—This step is used to check the accuracy of the model. Once training is completed, we now pass the other set of images which are not trained.

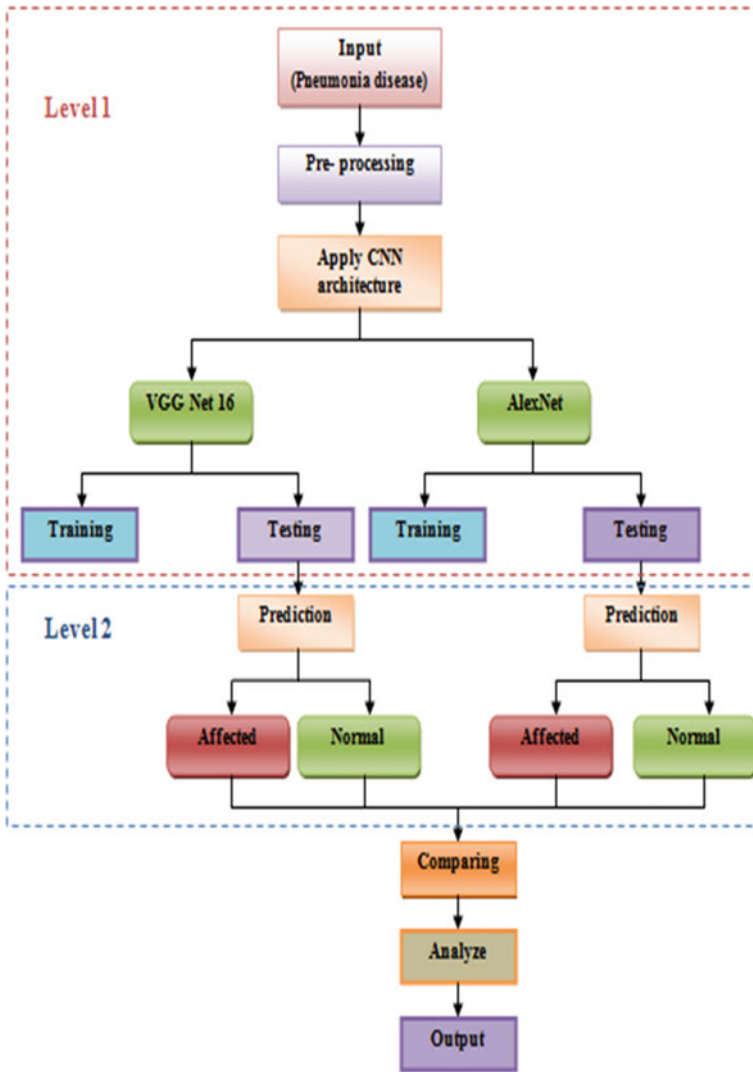


Fig. 2 Proposed PPDL model

If the test error rate decreases and the accuracy increases, then it is said that the model is working good, else we have to work more on the model in order to get more approximation. Testing is similar to both architectures.

Step 7: Prediction of images—In this step, we are passing a new image in order to check the capacity of the model to distinguish between normal image and pneumonia affected image. We are using confusion matrix on the predicted output of test images so that we can clearly see how many images are classified correctly and misclassified.

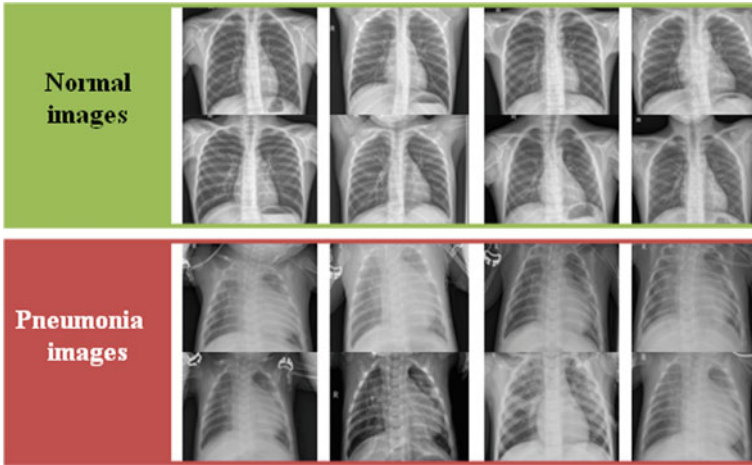


Fig. 3 Lungs dataset

Step 8: At this step, we are comparing the predicted outputs obtained from both the architectures so that we can decide which architecture is working better.

Step 9: In this last step, we perform analysis by calculating the final accuracy of both the architecture and conclude which architecture is showing best results for our dataset.

Dataset is retrieved from the Kaggle data repository [24] which consists of 5840 images as in Fig. 3, where 5216 images are taken for training and 624 images for testing.

1. Normal image—In the normal chest image, the outer thin lining of the chest is visible. As chest tissue is harder, the X-rays will not be able to penetrate through the tissue, hence color is darker.
2. Pneumonia image—In the pneumonia image, sometimes the outer lining of the chest will not be clearly visible. The affected regions of the chest are lighter when compared to the healthy region.

In our work, we are using VGG 16 and AlexNet architecture. VGG 16 and AlexNet are a convolutionary neural network educated from the ImageNet database on more than one million pictures.

3.1 VGG 16-Layer Architecture

The network is profound in 16 layers and can classify pictures into 2 classifications of images. It is improved as a network of AlexNet by replacing big kernel-sized filters one after the other with 3×3 kernel-sized filters. VGG 16 is trained with maximum

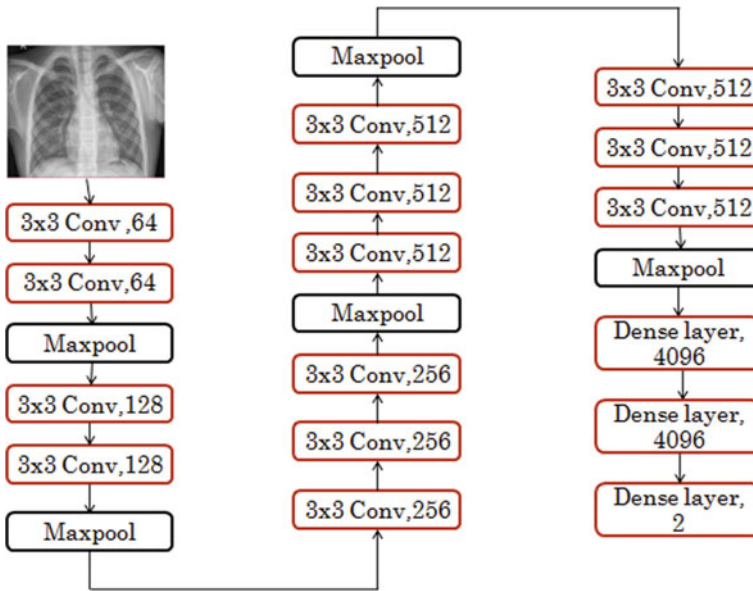


Fig. 4 VGG 16-layer architecture

number of images. The depth of filters increases as we move forward. Here, we have used only 3×3 filters as we are following VGG 16-layer architecture. Figure 4 shows the outer architecture of the VGGNet with 16 layers. The main architecture consists of:

- Convolution layer
- Pooling layer
- Fully connected layer

Convolution layer: Pneumonia images are given as an input in this layer, image size is taken as $150 \times 150 \times 3$ where 150×150 is width and height and 3 is depth of an image. Convolution neural networks are considered in the form of volumes where these layers take these volumes of activations and perform chunks of operations to produce new volumes of activations. In our case, we have depth as 3 which means this model consists of 3 channels which are said to be RGB channels. A 3×3 filter is used at convolution layer. Here, we have considered only 3×3 filters because we are using VGG 16-layer architecture. According to this architecture, using 2×2 size filters does not give good efficiency. And using larger size like 5×5 and 7×7 provides more number of parameters. When we consider 7×7 size filter, then the number of parameter obtained will be more, instead we can use 3×3 size filters simultaneously. Parameters obtained using three 3×3 filters will be lesser than that of one 7×7 filters, and network will also be deeper with ReLU activation function.

These filters convolve around the image, and these filters will extend full depth of the output volume. These filters convolve around the input volume till the full

Layer (type)	Output Shape	Param #
conv2d_79 (Conv2D)	(None, 64, 150, 150)	1792
conv2d_80 (Conv2D)	(None, 64, 150, 150)	36928
max_pooling2d_31 (MaxPooling)	(None, 64, 75, 75)	0
conv2d_81 (Conv2D)	(None, 128, 75, 75)	73856
conv2d_82 (Conv2D)	(None, 128, 75, 75)	147584
max_pooling2d_32 (MaxPooling)	(None, 128, 37, 37)	0
conv2d_83 (Conv2D)	(None, 256, 37, 37)	295168
conv2d_84 (Conv2D)	(None, 256, 37, 37)	590080
conv2d_85 (Conv2D)	(None, 256, 37, 37)	590080

Layer 1:
Input image given is of 3x150x150 size, have used 2 convolution layers with 64 3x3 filters , and a max pooling layer with 2x2

Layer 2:
In this layer I have used 2 convolution layers with 128 3x3 filters , and a max pooling layer with 2x2

Layer 3:
In this layer I have used 3 convolution layers with 256 3x3 filters , and a max pooling layer with 2x2

Fig. 5 VGG model for first 7 layers of the network

available spatial space and compute the dot product for each. After covering the spatial space of the input volume, a response obtained it considered to be the activation maps. Number of activation maps depends upon the number of filters used. In this work, we are using $16 \times 3 \times 3$ size filters in the first layer. Then, these activation maps are given as input to the next layer. This process continues to the next layers, as we go deeper the layers, the number of filters used increases accordingly as shown in Fig. 5.

Pooling layer: This layer makes the ConvNet less sensitive to small changes in the component area, in that the pooling layer yield continues even when a component is moved. There are different approaches to pooling, but most used is maxpooling. Imagine a window slipping over the component to conduct maxpooling. We grab the greatest values in the window as the window passes over the manual and dispose the remainder. So we are using maxpooling method with 2×2 size of the pane as shown in Fig. 5. Dropout is widely used to regulate profound neural networks, but it is essentially distinct to apply dropout on fully connected layers and dropout on convolutionary layers where 0.4 is set for dropout (Fig. 6).

Fully connected layer: The fully related layer has no less than three parts: an information layer, a hidden layer, and a return layer. The data layer is the former layer’s output, which is just a range of characteristics with 4096 channels. The neurons in the yield layer are related to each of the classes that the ConvNet is looking for. Like the communication between the information and the hidden layer, the yield layer takes

max_pooling2d_33 (MaxPooling (None, 256, 18, 18))	0
conv2d_86 (Conv2D) (None, 512, 18, 18)	1188168
conv2d_87 (Conv2D) (None, 512, 16, 16)	2359888
conv2d_88 (Conv2D) (None, 512, 14, 14)	2359888
max_pooling2d_34 (MaxPooling (None, 512, 7, 7))	0
conv2d_89 (Conv2D) (None, 512, 7, 7)	2359888
conv2d_90 (Conv2D) (None, 512, 5, 5)	2359888
conv2d_91 (Conv2D) (None, 512, 3, 3)	2359888
max_pooling2d_35 (MaxPooling (None, 512, 1, 1))	0

Layer 4:
In this layer I have used 3 convolution layers with 512 3x3 filters, and a max pooling layer with 2x2

Layer 5:
In this layer I have used 3 convolution layers with 512 3x3 filters, and a max pooling layer with 2x2

Fig. 6 VGG model form 8–13 layers of the network

values and their weights of comparison from the hidden layer and applies a capacity and results. Here in the yield layer, 4096 channels are reduced to 2 channels as we classify into only 2 classes. There are two categories under consideration normal and affected image as shown in Fig. 7. A Softmax activation function is used at the last layer.

flatten_6 (Flatten) (None, 512)	0
dense_19 (Dense) (None, 4896)	2181248
dropout_13 (Dropout) (None, 4896)	0
dense_20 (Dense) (None, 4896)	16781312
dropout_14 (Dropout) (None, 4896)	0
dense_21 (Dense) (None, 2)	8194

=====
Total params: 33,685,442
Trainable params: 33,685,442
Non-trainable params: 0
None

Here I have used 3 dense layers, with 2 dropout layers. At the last dense layer I have used Softmax activation function and it is finally classified into two classes

Fig. 7 VGG model for last dense layers of the network

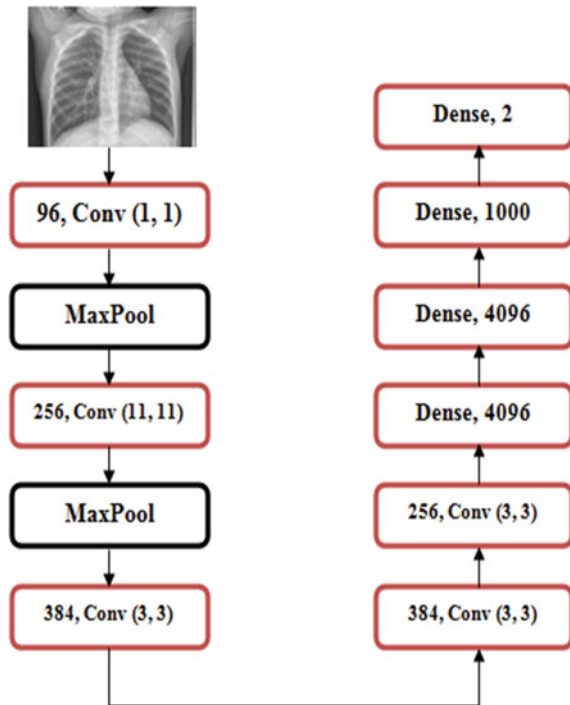
3.2 AlexNet Architecture

The network is profound in 9 layers and can classify pictures into 2 classifications of images. This architecture has no specified kernel size like in VGGNet; it is a simple architecture. Conv layers are not repeated many times, and it has 4 dense layers which help in classifying the images accurately. Figure 8 shows the outer architecture of the AlexNet with 16 layers. The main architecture consists of:

- Convolution layer
- Pooling layer
- Fully connected layer

Convolution layer: Pneumonia images are given as an input in this layer; image size is taken as $150 \times 150 \times 3$ where 150×150 is width and height, 3 is depth of an image. Convolution neural networks are considered in the form of volumes where these layers take these volumes of activations and perform chunks of operations to produce new volumes of activations. In our case, we have depth as 3 which means this model consists of 3 channels which are said to be RGB channels. Here, we are using one 1×1 , one 11×11 and three 3×3 filters, and the optimizer used is Adam optimizer.

Fig. 8 AlexNet architecture



Layer (type)	Output Shape	Param #	
conv2d_6 (Conv2D)	(None, 150, 150, 96)	384	Layer 1: Input image given is of 150x150x3 size, have used 1 convolution layers with 96 1x1 filters , and a max pooling layer with 2x2
max_pooling2d_4 (MaxPooling2)	(None, 75, 75, 96)	0	
conv2d_7 (Conv2D)	(None, 65, 65, 256)	2973952	Layer 2: In this layer we have used 2 convolution layers with 256 and 384, 11x11 and 3x3 filters , and a max pooling layer with 2x2
max_pooling2d_5 (MaxPooling2)	(None, 32, 32, 256)	0	
conv2d_8 (Conv2D)	(None, 30, 30, 384)	885120	
conv2d_9 (Conv2D)	(None, 28, 28, 384)	1327488	Layer 3: In this layer we have used 2 convolution layers with 384 and 3x3 filters , and a max pooling layer with 2x2
conv2d_10 (Conv2D)	(None, 26, 26, 256)	884992	
max_pooling2d_6 (MaxPooling2)	(None, 13, 13, 256)	0	

Fig. 9 AlexNet model for first 5 layers of the network

These filters convolve around the image, and these filters will extend full depth of the output volume. These filters convolve around the input volume till the full available spatial space and compute the dot product for each. After covering the spatial space of the input volume, a response obtained it considered to be the activation maps. Number of activation maps depends upon the number of filters used. In this work, we are using three 3×3 size filters in the third layer and 11×11 in second and 1×1 in the first layer. Then, these activation maps are given as input to the next layer (Fig. 9).

Pooling layer: The pooling layer also adds to the ConvNet ability to locate where it is in the image. In particular, the pooling layer makes the ConvNet less sensitive to small changes in the component area; it gives the ConvNet the property of translational invariance in that the yield of the pooling layer continues as before, even if a component is moved. Pooling also decreases the scope of the overview of the element, streamlining calculation in subsequent stages. Here, we are using only 2 maxpool layers overall with 2×2 kernel size; these 2 layers are seen after 1 and 3 layer of conv layer simultaneously (Fig. 10).

Fully connected layer: The fully linked layer is the layer where the last “decision” is done. At this layer, the ConvNet restores the probability of a particular kind of protest in an image. We have been speaking about the convolution neural structures to actualize something that many refer to as supervised learning. The data layer is the former layer’s output, which is just a range of characteristics with 4096 channels followed by 1000 channels. The neurons in the yield layer are related to each of the classes that the ConvNet is looking for. Like the communication between the information and the hidden layer, the yield layer takes values and their weights of comparison from the hidden layer and applies a capacity and results. Here in the yield layer, 4096 channels are reduced to 1000 channels and then to 2 channels as

flatten_1 (Flatten)	(None, 43264)	0
dense_1 (Dense)	(None, 4096)	177213440
dropout_1 (Dropout)	(None, 4096)	0
dense_2 (Dense)	(None, 4096)	16781312
dropout_2 (Dropout)	(None, 4096)	0
dense_3 (Dense)	(None, 1000)	4097000
dropout_3 (Dropout)	(None, 1000)	0
dense_4 (Dense)	(None, 2)	2002

Total params: 284,165,690
Trainable params: 284,165,690
Non-trainable params: 0

None

Layer 4:
In this layer we have used 2 dense layers with each 4096, and a drop out layer.

Layer 5:
In this layer we have used 3 dense layers with 4096 and 1000. It is followed by a drop out layer and at the end it is finally classified into 2 classes

Fig. 10 AlexNet model for last 4 dense layers of network

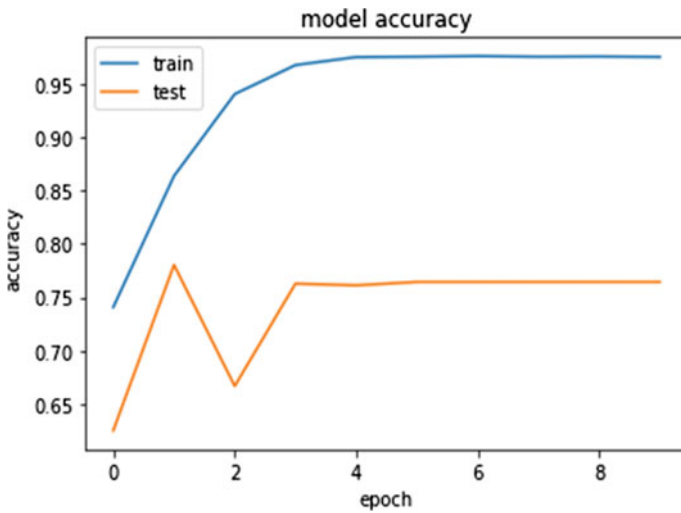


Fig. 11 Model accuracy for VGG model

we classify into only 2 classes. There are two categories under consideration normal and affected image. A Softmax activation function is used at the last layer.

Table 1 Architecture comparison between AlexNet and VGGNet

Variables	VGGNet 16	AlexNet
Number of layers	16 layers	9 layers
Kernel size used	3×3	$1 \times 1, 3 \times 3, 11 \times 11$
Optimizer	RMSprop	Adam
Loss function	Binary cross_entropy	Binary cross_entropy
Activation function	ReLU, Softmax	ReLU, Softmax
Filter size	64, 128, 256, 512	96, 256, 384
Number of dense layers	3 layers	4 layers
Pooling	Maxpooling	Maxpooling
Is kernel size fixed?	Kernel size is fixed as we use only 3×3 filter	Here, kernel size is not fixed. We can use it based on the need
Padding	Same	No padding
Dropout	0.4	0.5

3.3 Architecture Difference Between AlexNet and VGGNet

There are a lot of differences between AlexNet and VGGNet to begin with both architectures were designed for ImageNet dataset; AlexNet was introduced in 2012; and VGGNet was introduced in 2013. VGGNet architecture has 16 layers, and AlexNet has 9 layers. Kernel size is fixed for VGGnet, i.e., 3×3 and varies in AlexNet; it can be $1 \times 1, 3 \times 3, 5 \times 5, 7 \times 7$, and 11×11 . Loss function, activation remains same we are taking Binary cross_entropy as loss function and ReLU, Softmax as activation function. Here, we are using RMSprop as optimizer in VGGNet and Adam in AlexNet. When we consider networks in terms of layers then looking at Table 1, we can say that VGGNet is a deeper network than that of AlexNet. As we are performing classification in the fully connected layer, we are using Softmax activation function. Padding is required to retain same size of an image till the last layer; here in this work, we are using padding only for VGGNet where we are retaining the same size of an image. Dropout layer is used to eliminate the unwanted neurons like neurons which are not active, fault neuron, or neurons which are said to be empty. The dropout size of VGG is 0.4 and for AlexNet is 0.5.

3.4 Layer Architecture Difference Between AlexNet and VGGNet

VGGNet architecture is made up of 16 layers, and AlexNet is 9 layers where VGGNet has 2 conv layers with 64 filters, one maxpool layer with 2×2 size, and AlexNet has 2 conv layers with 96 and 256 filters, one maxpool layer with 2×2 . In the second set, VGGNet has 2 conv layers with 128 filters, one maxpool layer, and Alexnet has two

conv layers with 348 filters each. Where in the third set, we are using 3 conv layers with 256 filters each, one maxpool layer again, and AlexNet has one conv layer with 256 filters plus two dense layers with 4096 neurons each. Next fourth set in VGG has 3 conv layers with 512 filters each, one maxpool layer where Alexnet has last 2 dense layers third dense layer has 1000 neurons, and in the fourth dense layer, these 1000 neurons are shorted down to 2 in which the images are finally classified to 2 classes. In fifth set, VGG has 3 conv layers with 512 filters, one maxpool layer; the last sixth set has 2 dense layers with 4096 neurons, third dense layer with 2 seeds to classify the images into two classes. So totally VGGNet architecture is composed of 13 conv layers, 5 maxpool layers, and 3 dense layers, whereas AlexNet has 5 conv layers, 2 maxpool layers, and 4 dense layers (Table 2).

Table 2 Layer architecture comparison between AlexNet and VGGNet

VGGNet	AlexNet
<i>Set 1: 1–2 layers</i>	
Conv_1, 64	Conv_1, 64 Maxpool_1
Conv_2, 64 MAxpool_1	Conv_2, 256 Maxpool_2
<i>Set 2: 3–4 layers</i>	
Conv_3, 128	Conv_3, 348
Conv_4, 128 Maxpool_2	Conv_4, 348
<i>Set 3: 5–7 layers</i>	
Conv_5, 256	Conv_5, 256
Conv_6, 256	Dense_1, 4096
Conv_7, 256 Maxpool_2	Dense_2, 4096
<i>Set 4: 8–10 layers</i>	
Conv_8, 512	Dense_3, 1000
Conv_9, 512	Dense_4, 2
Conv_10, 512 Maxpool_3	Layer not present
<i>Set 5: 11–13 layers</i>	
Conv_11, 512	Layer not present
Conv_12, 512	Layer not present
Conv_13, 512 Maxpool_4	Layer not present
<i>Set 6: 13–16 layers</i>	
Dense_1, 4096	Layer not present
Dense_2, 4096	Layer not present
Dense_3, 2	Layer not present

4 Result Analysis

Results are obtained by training, loading of database, and testing of images, evaluation of each phase is done, and precision is achieved.

Table 3 discusses about the accuracy and loss values obtained for training and testing the images using PPDL model for VGGNet architecture.

Figure 11 illustrates the total accuracy obtained at each epoch. As we observe, the values change consistently. Training graph has gradually increased which helps the model to predict more accurately where as in testing, it is not showing much more difference. Figure 12 shows the loss function, and loss of training model is decreased which is a good criterion.

Table 4 discusses about the accuracy and loss values obtained for training and testing the images using PPDL model for AlexNet architecture.

Figure 13 illustrates the total accuracy obtained at each epoch. As we observe, the values change consistently. Training graph has gradually increased which helps the model to predict more accurately where as in testing, it is not showing much more difference.

Figure 14 shows the loss function, and loss of training model is decreased which is a good criterion (Fig. 15).

The prediction model can predict whether this person is affected or not when a new untrained image is given. So this proves that our model can say whether this person is diseased or not. Figure 7 shows correctly predicted diseased image (Table 5).

Whereas Table 1 describes the confusion matrix on test dataset which shows 383 images correctly classified as diseased affected and 139 images as healthy images. Rest 95 images are wrongly classified as false negative which is high at risk factors it should be taken care of. Remaining 7 images are misclassified as affected as per that of VGGNet model. But in AlexNet architecture, the confusion matrix on test dataset shows 385 images correctly classified as diseased affected and 142 images

Table 3 Accuracy table for VGGNet architecture

Epochs	Training accuracy	Training loss	Testing accuracy	Testing loss
1	0.7408	0.5904	0.6250	0.6157
2	0.8639	0.3105	0.7804	0.6524
3	0.9408	0.1606	0.667	1.1464
4	0.9680	0.0817	0.7628	0.8581
5	0.9755	0.0665	0.7612	0.9195
6	0.9758	0.0652	0.7644	0.9185
7	0.9764	0.0650	0.7644	0.9189
8	0.9758	0.0648	0.7644	0.9189
9	0.9760	0.0649	0.7644	0.9189
10	0.9757	0.0650	0.7644	0.9189

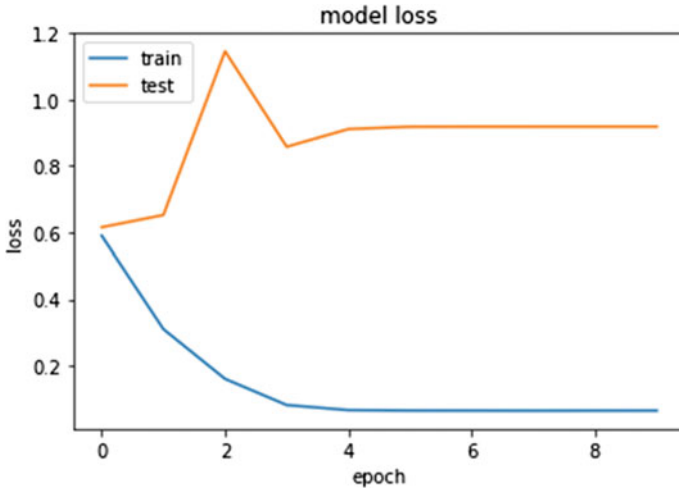


Fig. 12 Loss accuracy for VGG model

Table 4 Accuracy table for AlexNet architecture

Epochs	Training accuracy	Training loss	Testing accuracy	Testing loss
1	0.8183	0.4039	0.7708	0.5250
2	0.9404	0.1572	0.7596	0.7823
3	0.9549	0.1180	0.7676	0.8522
4	0.9582	0.1078	0.7560	0.9166
5	0.9611	0.1084	0.7580	0.9122
6	0.9595	0.1089	0.7580	0.9117
7	0.9586	0.1067	0.7580	0.9117
8	0.9595	0.1055	0.7580	0.9117
9	0.9603	0.1077	0.7580	0.9117
10	0.9594	0.1066	0.7580	0.9117

as healthy images. Rest 92 images are wrongly classified as false negative which is high at risk factors; it should be taken care of. Remaining 5 images are misclassified.

Figure 16 represents the comparison graph which is plotted between AlexNet and VGGNet where the parameters are training accuracy, testing accuracy, training loss, testing loss, time, memory, recall, and precision. By observing the graph, we notice that the accuracy obtained by VGGNet is more when compared to that of AlexNet. Time taken and memory used are more in VGGNet as it has deeper network.

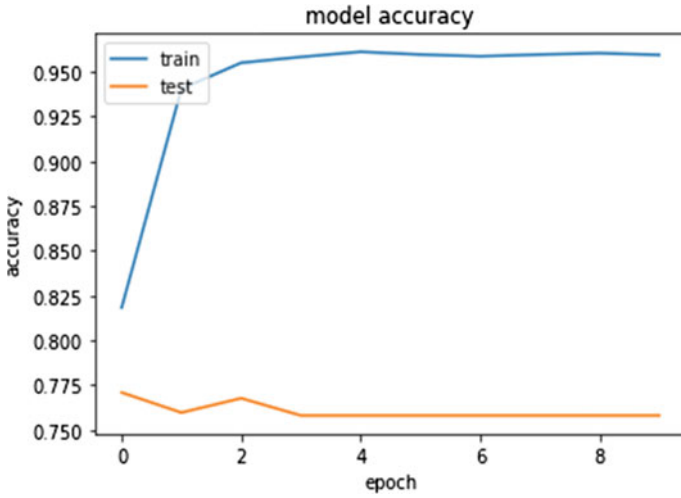


Fig. 13 Loss accuracy for AlexNet model

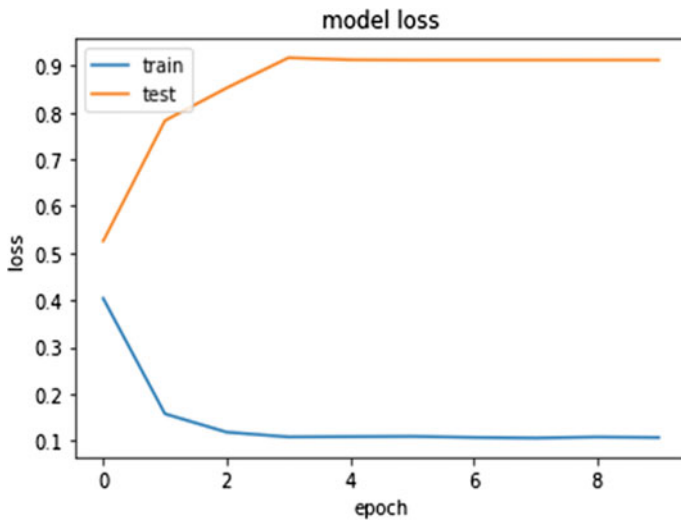


Fig. 14 Model accuracy for AlexNet model

5 Conclusion

The work is carried out is very useful for the society as maximum amount of the population is facing this problem; they can use this method to minimize the amount time, complications and also can take precautions on their daily health.

```
original_image = (944, 1416, 3)
resized_image = (150, 150, 3)
reshape_image = (1, 150, 150, 3)
[[0. 1.]]
value = [0]
This person is suffering from pneumonia
```

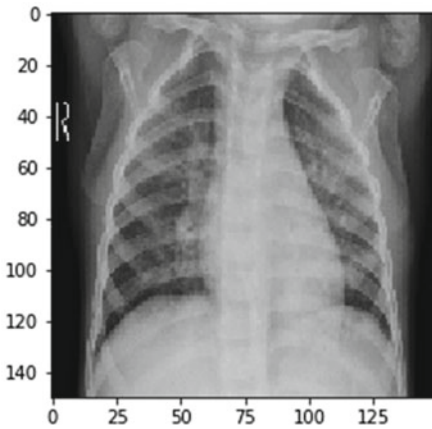


Fig. 15 Prediction model

Table 5 Comparison between AlexNet and VGGNet using confusion matrix

Confusion matrix	AlexNet	VGGNet
True negative	145	144
False positive	6	3
False negative	89	90
True positive	384	387
Precision	98.46%	99.23%
Recall	81.12%	81.13%

So here we have introduced pneumonia prediction using deep learning (PPDL) which can identify the disease through X-ray images. It is very beneficial for doctors as well as patients; this can reduce the waiting time of patients, so doctor get more time he can see more patients and treat them well. Our work is directly reducing the work load by classifying on its own. Here for this model, we have trained the model with 5216 images and tested with 624 images. Accuracy obtained for the VGG model at 10 epochs with the batch size of 250 images is 97.60% for training and 78.04% for testing by reducing loss rate to 0.0649. Precision and recall rate obtained for this model is 99.23 and 81.13%, whereas accuracy obtained for the AlexNet model at 10 epochs with the batch size of 250 images is 96.19% for training and 77.08% for

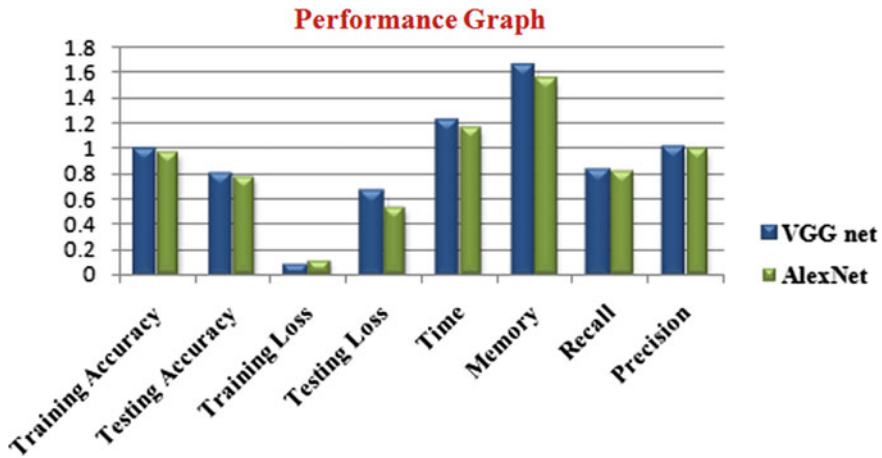


Fig. 16 Comparison graph

testing by reducing loss rate to 0.1084. Precision and recall rate obtained for this model are 98.46 and 81.12%. After analyzing the model, we can say that VGG 16 architecture is working better than that of AlexNet.

For the future work, we can improve the model by increasing the batch size and number of epochs. We can also use more other CNN models rather than VGG 16 and AlexNet and can achieve more accuracy. By training the model multiple times, we can achieve good accuracy.

Acknowledgements The authors express their sincere gratitude to Prof. N. R. Shetty, Advisor, and Dr. H. C. Nagaraj, Principal, Nitte Meenakshi Institute of Technology, for giving constant encouragement and support to carry out research at NMIT.

The authors extend their thanks to Vision Group on Science and Technology (VGST), Government of Karnataka, to acknowledge our research and providing financial support to set up the infrastructure required to carry out the research.

References

1. Isabella Noguez J, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Tans Med Imag* 35(5):1285–1298
2. Li D, Zhang J, Zhang Q, Wei X (2017) Classification of ECG signals based on 1D convolution neural network. In: 19th international conference on e-health networking, vol 5, pp 105–112
3. Albawi S, Mohammed TA (2018) Understanding of a convolutional neural network. In: International conference on engineering and technology. ISBN:978-1-5386-1948-3
4. Jmour N, Zayen S, Abdelkrim A (2018) Convolutional neural networks for image classification. In: International conference on advanced systems and electric technologies. ISBN:978-1-5386-4449-2/18

5. Bhandare A, Bhide M, Gokhale P, Chandavarkar R (2016) Applications of convolutional neural networks. *Int J Comput Sci Inform Technol* 7:2206–2215
6. Ding B, Quain H, Zhou J (2018) Activation function and their characteristics in deep neural network. In: Chinese control and decision conference. ISSN: 1947-9447
7. Vydana HK, Vuppala AK (2017) Investigative study of various activation functions for speech recognition. In: 23rd national conference on communication. ISBN: 978-1-5090-5356-8
8. Wang K, Kong Y (2017) Diagnosis of heart disease via CNNs. In: 2nd international conference on communication on electronic system. ISBN: 978-1-5090-5013-0
9. Kumar K, Chandra Sekhara Rao A (2018) Breast cancer classification of image using convolutional neural network. In: 4th international conference on recent advances in information technology. ISBN:978-1-5386-3039-6
10. Sasikala S, Bharathi M, Sowmiya BR (2018) Lung cancer detection and classification using deep CNN. *Int J Innov Technol Explor Eng* 8(2S):2278–2284
11. Haryanto T, Wasito I, Suhartanto H (2017) Convolutional neural network for gland images classification. In: International conference on information and computer technology systems. ISBN: 2338-185X
12. Guo T, Dong J, Li H, Gao Y (2017) Simple convolutional neural network on image classification. In: 2nd international conference in big data analysis, vol 8, pp 2145–2150. ISBN: 978-1-5090-3619-6/17
13. Liang Z, Powell A, Ersoy I (2016) CNN-based image analysis for Malaria diagnosis. In: International conference on bioinformatics and biomedicine (BIBM). ISBN: 978-1-5090-1610
14. Antin B, Kravitz J, Martayan E (2016) Detecting pneumonia in chest X-rays with supervised learning. *J Health Care Eng* 8
15. Amin SU, Shamim Hossain M (2019) Cognitive smart healthcare pathology detection and monitoring. In: International conference on bioinformatics and biomedicine (BIBM), vol 7. ISBN: 2169-3536
16. He K, Zhang X, Ren S (2016) Deep residual learning for image recognition, microsoft research, 2016. In: IEEE conference on computer vision and pattern recognition
17. Yu FL, Sun J, Li A, Cheng J, Wan C, Liu J (2017) Image quality classification for DR screening using deep learning. In: 39th annual international conference of the IEEE engineering in biological and medical field. ISSN: 1558-4615
18. Triwijoyo BK, Heryadi Y, Lukas, Ahmad AS (2016) Retina disease classification based on colour fundus images using convolutional neural networks
19. Liu S, Deng W (2016) Very deep convolutional neural network based image classification using small training sample size. In: 3rd IAPR Asian conference on pattern recognition. ISBN: 978-1-4799-6100-9/15
20. Sandoval C, Pirogova E (2017) Two-stage deep learning approach to the classification of fine-art paintings. ISBN: 2169-3536
21. Vatathanavaro S, Tungjitnob S, Pasupa K (2016) White blood cell classification: a comparison between VGG-16 and ResNet-50 models. In: International conference on innovative and technology study
22. Krizhevsky A, Sutskever I, Hinton GE (2016) ImageNet classification with deep convolutional neural networks. *IJERT*
23. Xiao L, Yan Q (2017) Scene classification with improved AlexNet model. In: 12th international conference on intelligent systems and knowledge engineering (ISKE)
24. Dataset: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>

An Efficient Blockchain-Based Security Framework for PUF-Enabled IoT Devices in Smart Grid Infrastructure



M. Prasanna Kumar and N. Nalini

1 Introduction

The Internet of Things (IoT) is a network infrastructure that allows multiple Internet-connected devices to be installed anywhere, from human bodies to the most remote parts of the globe, with more than 20 billion networked items expected by 2020 [1].

Smart grid systems are one of numerous areas of Industrial Internet of Things (IIoT) that have the potential to increase energy delivery dependability, flexibility, and quality [2]. However, as the system grows in size (e.g. as the number of customers grows), issues such as decreasing latency and enhancing quality of service (QoS) may arise [3]. As a result, there have been attempts to overcome these difficulties using edge computing, such as using electric vehicle charging stations as edge computing devices to make real-time decision-making easier, and as a result, improve provisioned QoS and eco-friendliness in latency-sensitive applications [4, 5].

As a non-centralised security mechanism, blockchain technology may be a viable solution for addressing high creditability and high-security concerns in IoT. The blockchain provides a secure, distributed, and autonomous framework that allows IoT devices, nodes, processes, and systems to securely connect with one another and sign transactions without the need for a third-party to process or verify transactions. Asymmetric authentication can be used by IoT devices to authenticate each other. The blockchain is a distributed ledger that maintains transactions that are made up of blocks of transactions that are cryptographically linked.

M. Prasanna Kumar (✉)
Department of ISE, SSIT, Tumakuru, India
e-mail: prasan.ctn19@gmail.com

N. Nalini
Department of CSE, NMIT, Bengaluru, India
e-mail: nalini.n@nmit.ac.in

By 2023, the global market for blockchain in energy is predicted to grow from USD 180.3 million in 2017 to over USD 5000 million. Blockchain is being used by businesses to manage data and track financial transactions and relationships. It also provides a safe avenue for corporations to manage their data. Due to their great significance, technologies such as blockchain are gaining appeal among corporations and other organisations in today's society. Operational expenses, capital expenditure, risk management, and security are all areas where blockchain can have a substantial impact. Increased automation, as well as data integrity and security, is projected to help the global economy grow. Ledgers are dispersed throughout a network of computers in blockchain, leaving no room for hackers. The system is totally transparent, with all users able to see transactions and modifications made on public blockchains if they are supplied. As a result, numerous sectors are experimenting with blockchain. The energy business can benefit greatly from blockchain, which provides new, tamper-proof techniques for authentication, authorisation, and data transmission.

For the past few years, we have heard reports about attempts to hack into electrical grids in the United States, therefore, smart grid security is vital. As a result, blockchain technology could be the key to better interconnectivity, data interchange, and permission control security requirements. Automation and remote access are key components of smart grids. These, in turn, bring with them security problems, which we are only now beginning to address.

Because blockchain includes identification security, which is achieved by a public-private key encryption with key access, anyone attempting to gain access to a system must check the credentials and their authentication before doing anything on the network. If key access codes are basically kept safe and secure, blockchain is the technology that will ensure the safety of electricity grids. Because of enhanced data collecting, intelligent electricity metres help both energy providers and end customers. However, if these smart metres are not properly secured, hackers might gain access to important customer information on a wide scale. By functioning as a decentralised transaction log, blockchain can minimise security gaps and establish the circumstances for peer-to-peer trading, in which local energy trade is made possible owing to big energy suppliers. As a result, a new decentralised security system that can meet basic security requirements including secrecy, integrity, and authentication must be proposed. Apart from that, the system must address the drawbacks of a centralised architecture.

Hence, the focus of this paper is on building a decentralised model for smart grid application. As we know, physically unclonable functions are the promising hardware security primitives for IoT devices. The advantages of physically unclonable functionalities, as well as the decentralised and distributed nature of blockchain technology, are used in our proposed framework to construct a security framework for IoT devices in smart grid infrastructure.

The outline of the paper is as follows: Sect. 2 briefs the related work on blockchain and PUF technology in building security solution in IoT environment. Section 3 describes the system model and architecture of the framework. Section 4 outlines the

implementation of the system, Sect. 5 discuss the outcomes of the work, and finally, Sect. 6 concludes work carried out in the paper.

2 Related Work

The volume of data created by IoT devices is continually increasing as the IoT business grows and the number of connected devices grows. However, IoT security and privacy issues have arisen as a result of its rapid growth. A slew of recent research has focused on blockchain and its usage in IoT security and privacy, as well as an alternate solution for IoT device identification and authorisation. The Internet of Things (IoT) is a rapidly evolving technology that consists of disruptive networked smart gadgets that are connected via the Internet without the need for human intervention to exchange sensor-based data. IoT devices are low-capacity devices (nodes) with a variety of problems, including processing, connectivity, and most importantly, security [6]. As the first gateway to the network, IoT devices require authentication and authorisation, which is one of the most important security criteria [6]. To create secure communication, these independent, networked nodes must first authenticate each other. Mutual authentication is an effective method for ensuring trust identity and safe communications by authenticating the identity of Internet-connected communicators prior to future interactions and avoiding the transmission of critical information over an open channel [7].

A number of security solutions for smart grids and edge computing systems have been developed in recent years. Tsai and Lo, for example, proposed an anonymous key distribution mechanism based on identity-based signature and encryption [8] to construct secure communication sessions. In the paper [9], He et al. provided a novel key agreement and authentication mechanism which has lower computation and communication costs than [8]. However, it was later pointed out that the protocol is subject to ephemeral secret key leaking and does not ensure the privacy of smart metre credentials, hence, an improved authenticated key agreement protocol [10] was introduced. In [11], Jia et al. proposed and explicitly verified the security of an efficient identity-based anonymous authentication mechanism for mobile edge computing. The protocol, on the other hand, does not take into account key management of communication participants.

The potential use of decentralised blockchain technology to address IoT security issues was investigated in the article [12]. Using genetic algorithms and particle swarm optimisation, a self-clustering approach for IoT networks is proposed, which clusters the network into K-unknown clusters and improves the network lifetime. To verify the proposed system, the model uses the open source hyperledger fabric blockchain platform. Wang et al. presented a blockchain-based mutual authentication and key agreement protocol for smart grid edge computing devices in [13]. Specifically, the protocol can allow efficient conditional anonymity and key management without the use of other sophisticated cryptographic primitives by leveraging blockchain. A mutual authentication-based key agreement protocol has been

designed in the paper [3]. The developed protocol takes advantage of FHMV, ECC, and the one-way hash function to provide a mutual authentication mechanism that is provably secure. In the above works, blockchain is used to implement completely decentralised security solutions in IoT systems. Permissioned blockchain, such as hyperledger fabric, has a lot of potential as an infrastructure for IoT security, credit management, and other things. In [14], EC-ElGamal-based transaction encryption and enhanced SHA-384-based block hashing are used to increase lightweight scalable blockchain for better acceptance in blockchain-based IoT applications.

Traditional cryptographic security methods are out of reach for many embedded systems and IoT applications due to a lack of resources. It is necessary to use lightweight security primitives. PUF is another option for generating low-cost keys. In restricted IoT applications, PUFs paired with other factors can give a solid authentication system. Gope and Sikdar [15] proposes a two-factor authentication strategy for Internet of Things (IoT) devices that addresses privacy and resource constraints. PUFs are one of the authentication factors in this method. The second factor is a password or a shared secret key. An authentication and key exchange system based on PUFs, Keyed Hash, and identity-based encryption has been created in [16] (IBE). The protocol removes the need for the verifier to store the PUF's challenge answer database and the need for a security method to keep it secret. However, the protocol requires resource optimisation for encrypting frames, and side channel vulnerabilities must be investigated. In [17], PUF-based key-sharing approach is presented, in which the same shared key can be created physically for all devices, allowing it to be used in a lightweight key-sharing protocol for IoT devices. In all these works, PUF emerges as an efficient mechanism to implement security in IoT ecosystem.

In our proposed framework, the benefits of physically unclonable functions and decentralised and distributed nature of blockchain technology are used to implement security framework for IoT devices in smart grid infrastructure.

3 System Model

Figure 1 shows the proposed framework for authentication and authorisation of IoT devices in smart grid infrastructure. The various components in smart grid infrastructure include.

ESP: ESP is the electricity service provider which is responsible for registering smart devices. All participants in the smart grid have confidence in it. It distributes the keys to all devices connected to it and also utilises permissioned blockchain to store the authentication information of the devices in the system. ESP uses a secure private channel for registration of smart end devices.

IES: IES is the intermediate edge server which has sufficient resources to execute security algorithm and able to communicate with the blockchain system during authentication process. IES is a part of permissioned block chain. To prevent web spoofing attacks and ensure the blockchain's proper operation, each IES joins the

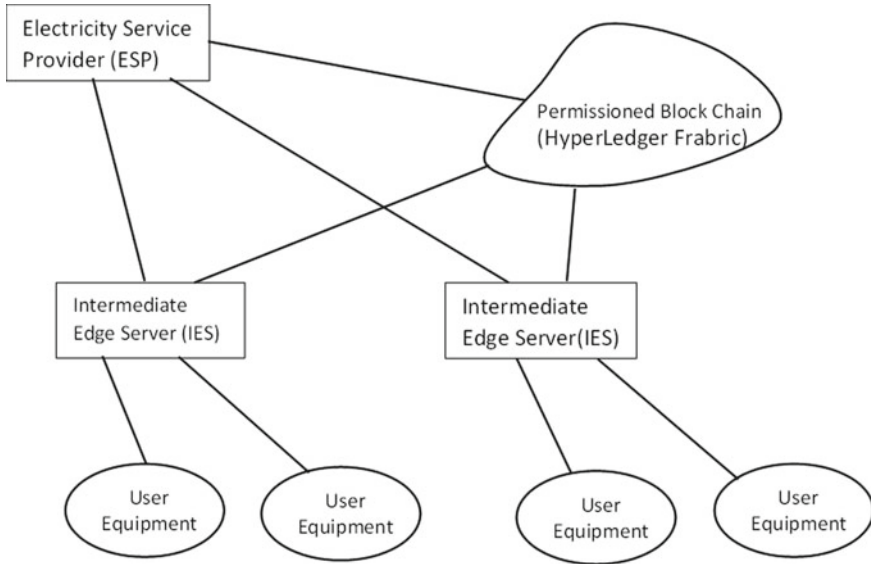


Fig. 1 Proposed system architecture

BC network [13]. The ES can also connect to a distant cloud to perform additional data analysis or long-term data storage.

UE: UE is the user equipment which is usually a smart metre. These devices update information about power consumption and associated data to an IES. Each UE connects with the nearest intermediate edge server.

PBC: The permitted blockchain (PBC) is responsible for providing decentralised and distributed database for storing end user authentication information. In the resource-constrained IOT environment, the permitted blockchain (PBC) is more efficient. The system is implemented on permitted blockchain-hyperledger fabric.

4 Implementation

Permitted blockchain, such as hyperledger fabric, has a lot of potential as an infrastructure for IoT security. A decentralised peer-to-peer network underpins the blockchain security system. Hyperledger fabric is an open source blockchain platform [18] used in our design implementation. The hyperledger platform will address concerns such as latency and decentralisation that are associated with blockchain implementation.

Transactions are used by IoT devices to interface with blockchain. Smart contracts will outline the many types of transactions that will be used to carry out various operations. The smart contracts make communication between the blockchain network

and IoT nodes easier. The IES sends various requests to the smart contract in order to perform various transactions in the blockchain network, to authenticate UE.

Initialisation of a blockchain.

To generate a blockchain, RA prepares a genesis file that includes the necessary settings. The RA then chooses a few trustworthy partners and launches the blockchain using a specific consensus process. The RA can join an existing blockchain system (e.g. hyperledger fabric) directly for simplicity.

Registration of a new device.

UE and ESP initially use a secure medium to execute the registration process of the new device. If the device is not already registered, ESP generates authentication credentials for the device and stores in the smart contract of permissioned blockchain. During registration, UE generates challenge-response pairs (CRP) and shares with the service provider(ESP). ESP stores this CRPs of the device in the block chain network. The steps are summarised as follows.

PUF-based devices produce one of a kind challenge-response pairs. The IoT devices are expected to be PUF capable in this scenario. Each device creates a challenge-response pair (CRP) that is exchanged with the server's challenge-response pair. Initially, CRP is exchanged between a device and the server. Both will select a random challenge and generate responses using PUFs in this scenario. The device produces and sends a C_d, R_d challenge-response pair to the server. In the same way, the server creates a challenge-response pair and sends it to the device. Finally, the tuple $\langle C_d, C_s, R_s \rangle$ is saved by the device in its memory, while the server stores the tuple $\langle C_s, C_d, R_d \rangle$ in the smart contract. This data is used for authentication and to set up a session for communication (Fig. 2).

Authentication and Session Establishment

UE requests for authentication with the nearest IES. IES retrieves information of the device from blockchain network and sends a challenge to the requested device.

The device has saved a tuple $\langle C_d, C_s, R_s \rangle$ containing a CRP of server and the challenge from the CRP it has shared with server during the initial CRP exchange phase. The device computes PUF output for the challenge C_d on the fly based on this information. After that, the device looks for a key $K = h(R_d R_s C_s)$ and selects a random integer R_{n1} . Following these calculations, a request message is sent to the server by the device with the parameters K, R_{n1} .

When the IES receives the request message, it looks for PUF output R_s for the challenge C_s , which is $R_s = \text{PUF}(C_s)$. It computes the value $h(R_d R_s C_s)$ and checks it with the K using R_s and (C_s, R_d) from the saved data $\langle C_s, C_d, R_d \rangle$. The device request is approved if it is valid; otherwise, it is rejected. Now, the server calculates $L = h(R_d R_s C_d)$ using a random number R_{n2} . Finally, the server responds with an L, R_{n2} message. Simultaneously, the IES computes $SK = (R_s \oplus R_{n2}) \oplus (R_s \oplus R_{n1})$.

Value L is verified by the device using its available data after getting the server answer. The server response will be accepted by the device if it matches. The session key $SK = (R_s R_{n2}) (R_s R_{n1})$ is then computed by the device. After successful session key exchange between the IES and the UE, the communication is initiated.

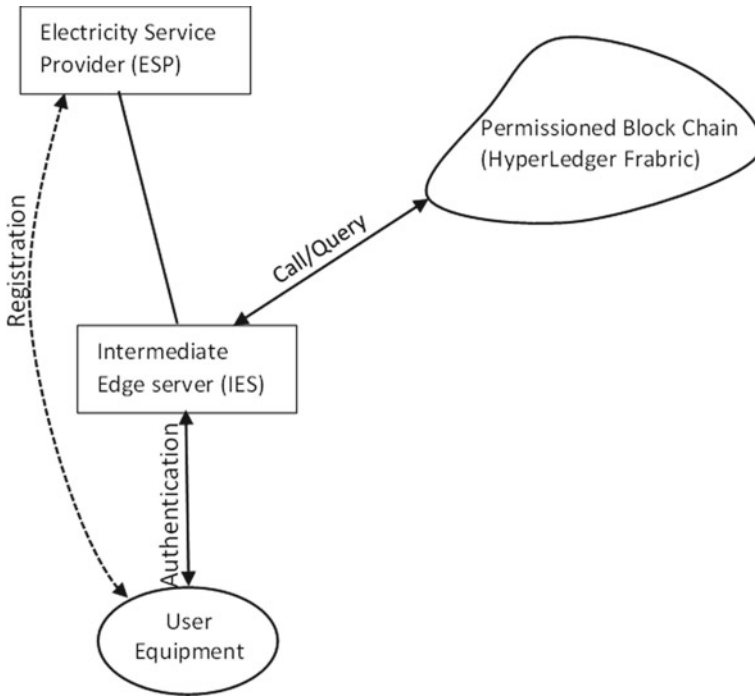


Fig. 2 Authentication and session establishment

5 Discussion

The suggested solution efficiently combines the advantages of blockchain and PUF technology to create a secure, lightweight framework for smart grid applications. Our approach is more efficient in the resource-constrained IoT ecosystem, according to the findings. The suggested strategy uses blockchain technology to allow systems to control their devices and resources without relying on a centralised authority to create trust relationships with unknown nodes. The computationally expensive elliptic-curve cryptosystem is used in all user-centric authentication schemes (ECC). Our proposed technique, on the other hand, is based on symmetric key crypto-systems that are computationally efficient like PUF, which are ideal for resource-constrained IoT devices. Permissioned networks also make good use of blockchain, such as storing data in its decentralised form. When we compared permissioned blockchains against permissionless blockchains, we found that permissioned blockchains outperform permissionless blockchains. The platform’s restricted number of nodes is the main reason behind this. This reduces the number of superfluous computations required to reach network consensus, resulting in improved overall performance. Combining PUF technology with blockchain in IoT scenario proves to be efficient.

6 Conclusion

In smart grid architecture, the ability to provide private and secure communication between end users, edge servers, and service providers is critical. A unique anonymous authentication and authorisation technique with efficient key management were introduced in this study. Unlike most existing protocols, the proposed protocol not only provides fundamental security qualities but also accomplishes additional key security properties. The protocol's main feature is that it makes use of PUF and blockchain technology to provide a more secure and simple solution to protect IoT applications and other types of data. To test the proposed system, the model uses hyperledger fabric, an open source blockchain technology. A framework for smart devices is provided by the authentication and authorisation mechanism at lower layers to communicate locally with intermediate edge servers, while a permissioned blockchain implementation is explored for the upper layer communications.

References

1. Lyu L, Nandakumar K, Rubinstein BIP, Jin J, Bedo J, Palaniswami M (2018) PPGA: privacy preserving fog-enabled aggregation in smart grid. *IEEE Trans Indus Inform* 14(8):3733–3744. <https://doi.org/10.1109/TII.2018.2803782>
2. Wang K, Yu J, Yu Y, Qian Y, Zeng D, Guo S, Xiang Y, Wu J (2018) A survey on energy internet: architecture, approach, and emerging technologies. *IEEE Syst J* 12(3):2403–2416
3. Garg S, Kaur K, Kaddoum G, Rodrigues JJPC, Guizani M (2020) Secure and lightweight authentication scheme for smart metering infrastructure in smart grid. *IEEE Trans Industr Inf* 16(5):3548–3557. <https://doi.org/10.1109/TII.2019.2944880>
4. Sarkar S, Chatterjee S, Misra S (2018) Assessment of the suitability of fog computing in the context of internet of things. *IEEE Trans Cloud Comput* 6(1):46–59. Available <https://doi.org/10.1109/TCC.2015.2485206>
5. Kumar N, Zeadally S, Rodrigues JJPC (2016) Vehicular delay-tolerant networks for smart grid data management using mobile edge computing. *IEEE Commun Mag* 54(10):60–66. Available <https://doi.org/10.1109/MCOM.2016.7588230>
6. Zhang Z, Cho MCY, Wang C, Hsu C, Chen C, Shieh S (2014) IoT security: ongoing challenges and research opportunities. In: 2014 IEEE 7th international conference on service-oriented computing and applications, pp 230–234
7. Wu L, Wang J, Choo KR, He D (2019) Secure key agreement and key protection for mobile device user authentication. *IEEE Trans Inform Forens Secur* 14(2):319–330. Available <https://doi.org/10.1109/TIFS.2018.2850299>
8. Tsai J, Lo N (2016) Secure anonymous key distribution scheme for smart grid. *IEEE Trans Smart Grid* 7(2):906–914
9. He D, Wang H, Khan MK, Wang L (2016) Lightweight anonymous key distribution scheme for smart grid using elliptic curve cryptography. *IET Commun* 10(14):1795–1802
10. Odelu V, Das AK, Wazid M, Conti M (2018) Provably secure authenticated key agreement scheme for smart grid. *IEEE Trans Smart Grid* 9(3):1900–1910
11. Jia X, He D, Kumar N, Choo K-KR (2019) A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. *IEEE Syst J* 14(1):560–571
12. Rashid MA, Pajoo HH (2019) A security framework for IoT authentication and authorization based on blockchain technology. In: 18th IEEE international conference on trust, security and privacy in computing and communications/13th IEEE international conference on big

- data science and engineering (TrustCom/BigDataSE), pp 264–271. <https://doi.org/10.1109/TrustCom/BigDataSE.2019.00043>
13. Wang J, Wu L, Choo K-KR, He D (2020) Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure. *IEEE Trans Industr Inf* 16(3):1984–1992. <https://doi.org/10.1109/TII.2019.2936278>
 14. Guruprakash J, Koppu S (2020) EC-ElGamal and genetic algorithm-based enhancement for lightweight scalable blockchain in IoT domain. *IEEE Access* 8:141269–141281. <https://doi.org/10.1109/ACCESS.2020.3013282>
 15. Gope P, Sikdar B (2019) Lightweight and privacy-preserving two-factor authentication scheme for IoT devices. *IEEE Internet Things J* 6(1):580–589. <https://doi.org/10.1109/JIOT.2018.2846299>
 16. Chatterjee U et al (2019) Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database. *IEEE Trans Depend Secure Comput* 16(3):424–437. <https://doi.org/10.1109/TDSC.2018.2832201>
 17. Zhang J, Qu G (2020) Physical unclonable function-based key sharing via machine learning for IoT security. *IEEE Trans Industr Electron* 67(8):7025–7033. <https://doi.org/10.1109/TIE.2019.2938462>
 18. Foundation L (2016) Hyperledger whitepaper. v2.0.0, pp 1–19

The Abstraction of XOR Gate Using Reversible Logic



Uttkarsh Sharma, Shruti Gatade, and N. Samanvita

1 Introduction

High consumption, low speed, and density beyond 10 nm have all hindered the usage of CMOS technology in recent years. To address these issues, a group of experts developed a solution for this traditional CMOS technology, known as quantum dot cellular automata (QCA), which is employed in high-speed applications. CMOS technology also works at nanoscale, but QCA uses quantum cells rather than transistors in circuits which gives quick results with less power dissipation compared to CMOS. Quantum cells are basic building blocks of circuit to be designed in QCA. These cells contain 4 quantum dots as shown in Fig. 1 [1–4].

A quantum dot is a semiconductor particle which can emit light of specific wavelength when applied with some energy. These quantum dots are usually represented as a core–shell structure with the size almost in nanometres [5, 6].

The rest of the paper is organized as following: In Sect. 2, basic of quantum dot cellular automata (QCA). Section 3 quantum gates. In Sect. 4 implementation and data flow. Section 5 shows the simulation results in QCA designer tool, and Sect. 6 shows the results analysis. Finally, Sect. 7 concludes the paper.

2 Basics of Quantum Dot Cellular Automata (QCA)

The quantum cell acts as a wire through which the data flows; hence, it is also termed as QCA wire. A 5-cell QCA wire is represented in Fig. 2. In this QCA wire, data transfer occurs from A to B. The data flowing through the intermediate cells are

U. Sharma · S. Gatade (✉) · N. Samanvita
Nitte Meenakshi Institute of Technology, Bengaluru, India
e-mail: Shruti.gatade@nmit.ac.in

Fig. 1 Basic quantum cell

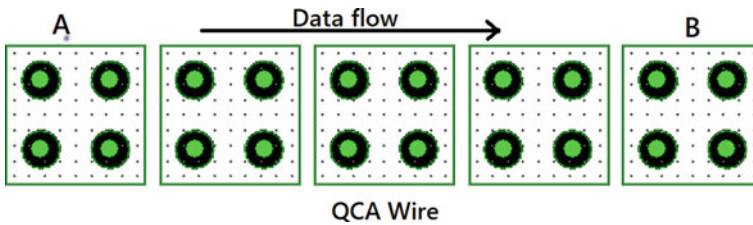
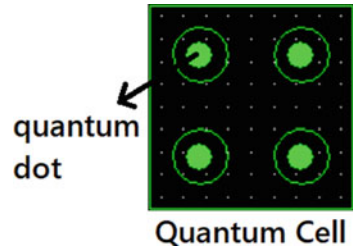


Fig. 2 QCA wire

similar to that of cell A. Similar to a classical circuit, branching can be done in a QCA wire so as to obtain same output at multiple ends [7, 8].

(a) **Majority Gate**

A majority gate is 3 inputs, 1 output gate. It is a basic gate for all the circuits in QCA. Any operation or any gate is designed using majority gate.

In Fig. 3, it can be seen that A, B, C are the inputs and Y is the output. The middle cell is the one where the processing takes place. It is also called as device cell. As the name “majority” suggests, the majority value of the input is processed and given as the output. As we are aware that quantum computing works on 0’s and 1’s, hence inputs can either have 0’s or 1’s. If the majority of the inputs is 1, then the output will be 1 and vice versa. Using this knowledge AND, OR, NOT, and many gates can be designed. But the only thing that differs is the polarization [9, 10].

(b) **Polarization**

Polarization can be understood as the constant input to any circuits. Consider Fig. 3, let the input “C” be the constant input of 0. In order to achieve this, we need to polarize the cell with input 0 which is represented as “-1 state” as shown in Fig 4.

Similarly, to give a constant input as 1 the cell must be polarized to “1 state”. Figure 5 shows different gates using polarization and majority gate [9].

(c) **Clocking**

Clocking can also be referred to as delaying of data. In general, clocking is used to co-ordinate the data flow in order to attain accurate results. In QCA, clocking has four phases: Switch, Hold, Release, and Relax as depicted in Fig. 6.

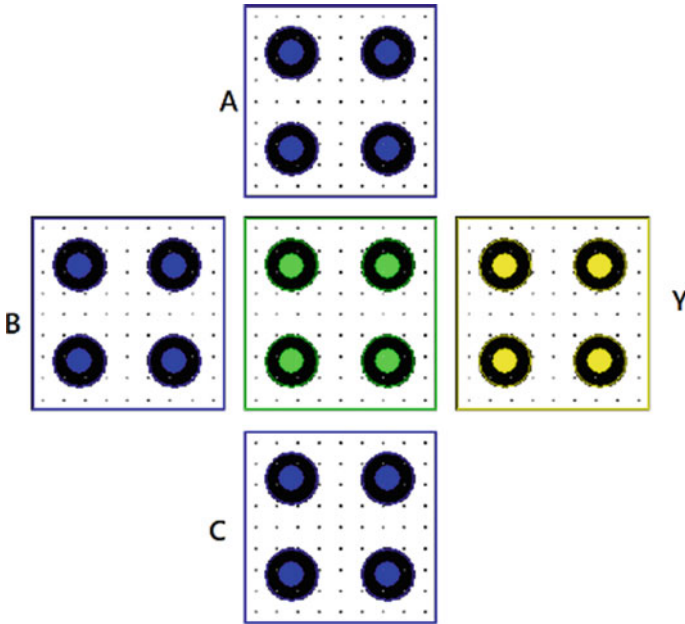
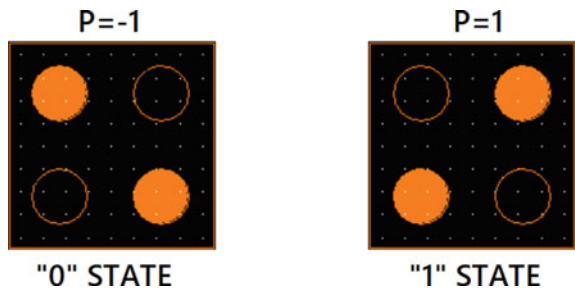


Fig. 3 Majority gate

Fig. 4 Polarization of a cell



The most important concept about clocking is the flow of data from one clock cycle to another which is an important concept while understanding the working of reversible gates (Fig. 7).

Let us assume the time taken for data to flow from one cell to another under clock 0 is 1 ns. The time taken for the data transfer from one cell to another in different clocks is 5 ns. This is because changing from one clock to another creates a delay in transfer of data.

There are four clock cycles: clock 0, clock 1, clock 2, clock 3. Delay of data can also be seen from clock 0 to 1, 1 to 2, 2 to 3, and so on [10–12].

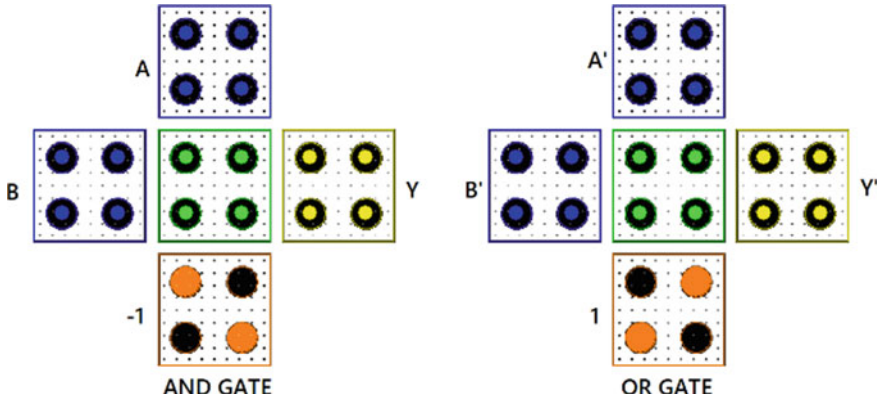


Fig. 5 Basic AND and OR gate

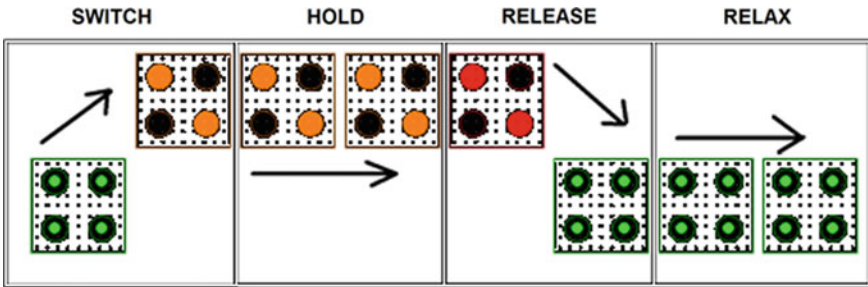


Fig. 6 Clocking phase in QCA

(d) **XOR Gate**

An exclusive EX-OR/XOR gate is a classical gate with two inputs and one output. XOR gate is made up of two “AND” gates and one “OR” gate. It gives output as “1” only if either of the input is “1” (Fig. 8).

$$Y = 1 \text{ only if } A \neq B$$

In QCA, designing of XOR can be achieved using below methods

a. **Irreversible gates**

The gate which does not have the number of inputs equals the number of outputs.

b. **Reversible gates**

The gates have the number of inputs equal to the number of outputs.

In further sections, we will get an insight on designing of XOR gate using reversible gate method.

Fig. 7 Data flow in clock cycle

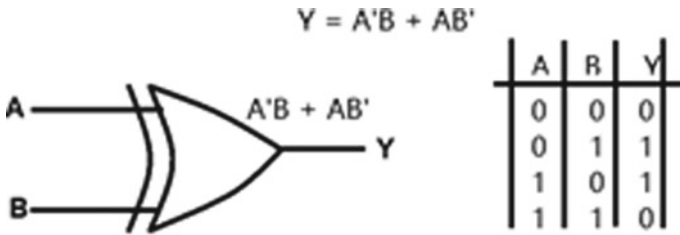
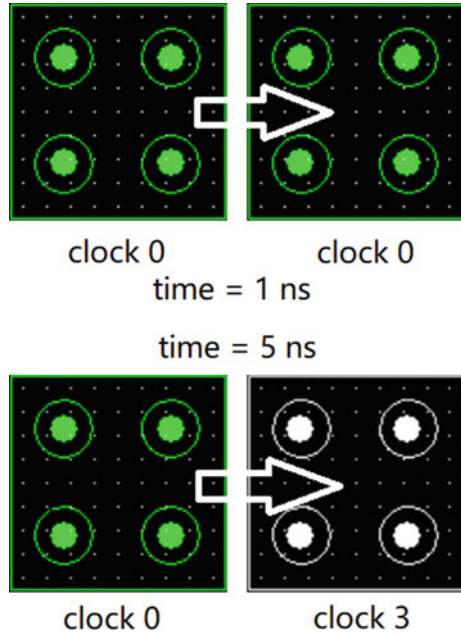


Fig. 8 Logic diagram, truth table, and equation

3 Quantum Gate

A quantum gate is a small circuit which uses some qubits for operation just as classical gates using 0's and 1's. There are different types of quantum gate. One of the quantum gate is Feynman gate.

Feynman gate is a quantum gate that uses the reversibility concept [13, 14]. This gate has two inputs A, B and two outputs P, Q.

Figure 9 shows the design of Feynman gate which also resembles another quantum gate called CNOT gate. Figure 10 depicts the output waveform of Feynman gate for the corresponding input signal which is similar to the output of XOR gate [15].

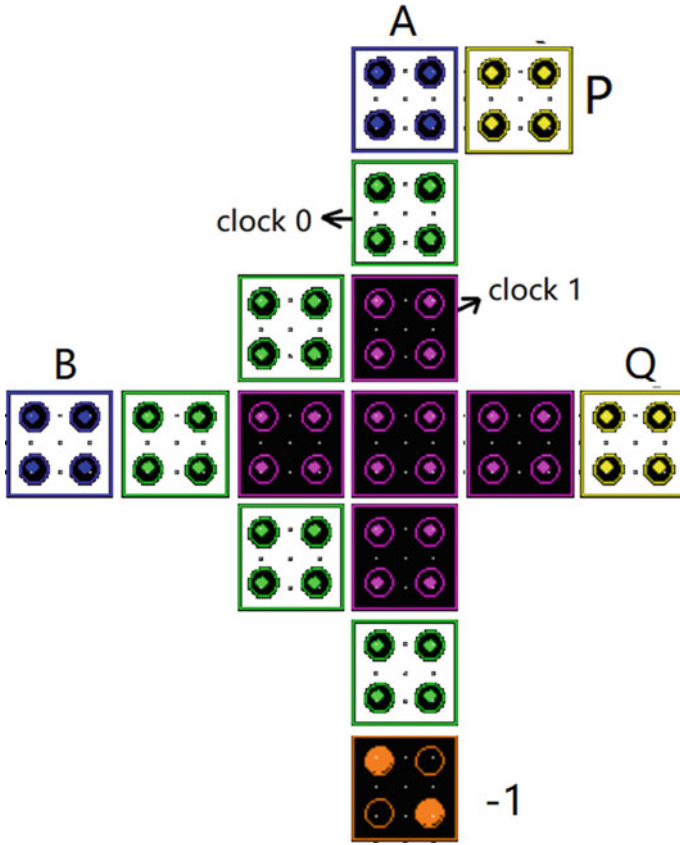
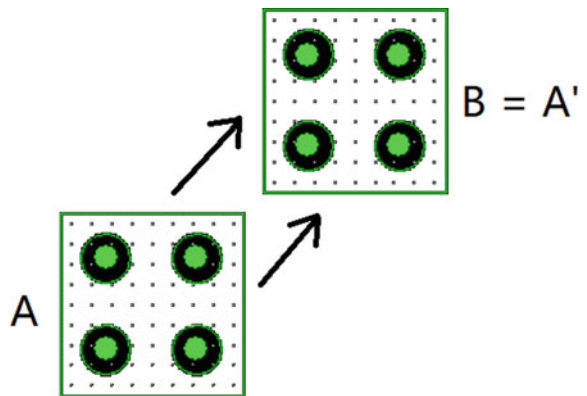


Fig. 9 Feynman gate

Fig. 10 NOT gate



4 Implementation and Flow of Data

(a) NOT Gate

Before moving on with designing the XOR gate, it is necessary to learn an important concept of NOT gate which is depicted in Fig. 10.

The input to the cell A will be inverted and obtained at the output cell B, by placing the cell B at an angle 45° upright to the cell A as shown in Fig. 10. Among the various designs of NOT gate, the above design is more efficient and less area consuming.

(b) XOR Gate Design

The actual XOR gate representation [16] is as shown in Fig. 11. Data flow is the important part in understanding the working of XOR.

Consider input A in the Fig. 12, when the data flows from input A to a normal cell of clock 0, it has two paths to go. First to clock 1 represented by a purple cell and second to clock 0 represented by green cell.

Since we have discussed time delay in the previous section, we know that the data will flow first to “clock 0” cell and then to “clock 1”. Similar is the case with input B in Fig 13.

Fig. 11 XOR gate

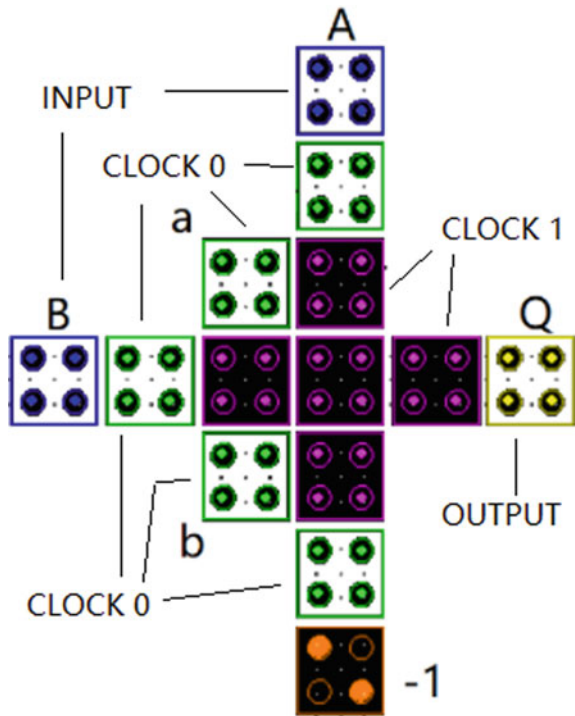


Fig. 12 Data flow

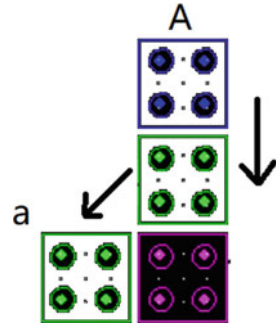
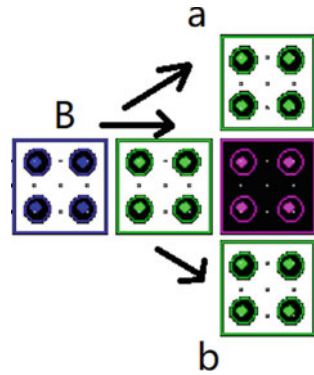


Fig. 13 Device Cell 'b'



Referring to Fig. 12, the cell “a” receives the data from both the inputs A and B. Thus, cell “a” acts as a device cell; similarly, the cell “b” shown in Fig. 13 also acts as a device cell.

The data from the device cell “a” flow in two directions shown in Fig. 14.

Fig. 14 Flow of data from device cell

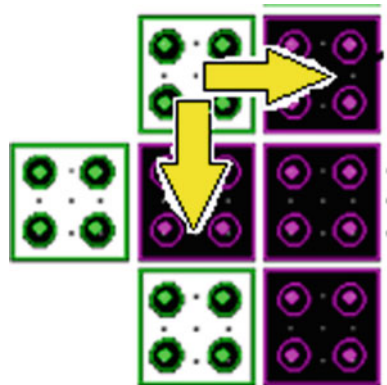
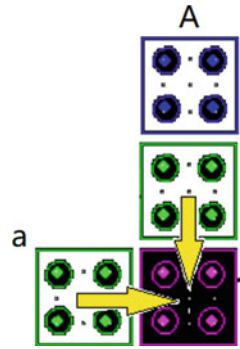


Fig. 15 Flow of data when the cell is in clock 1



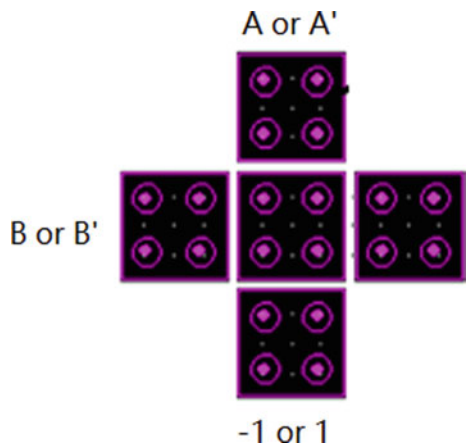
The “clock 1” cells already received data through both inputs A and B shown in Fig. 15.

The data from the inputs will be first received to the “clock 1” cell, so that data will be processed first and the data from device cell “a” will be processed in the next cycle. Similar process happens at the other two ends, input “B” and input “-1”.

If we notice, the data from device cell “a” is inverted input of A and B. So we can say that inputs A and A’ are processed alternatively. This concept is applied for the rest.

The main part of the XOR circuit is its majority gate. In this majority gate, the input of “-1 or 1” decides the operation of either “AND” or “OR” gate shown in Fig. 16. There are many possibilities of different combinations, but the valid output is only $A'B + AB'$, and rest all are garbage outputs that can lead to disturbances in output just as in output Q in Fig. 17.

Fig. 16 Majority gate working as AND/OR gate



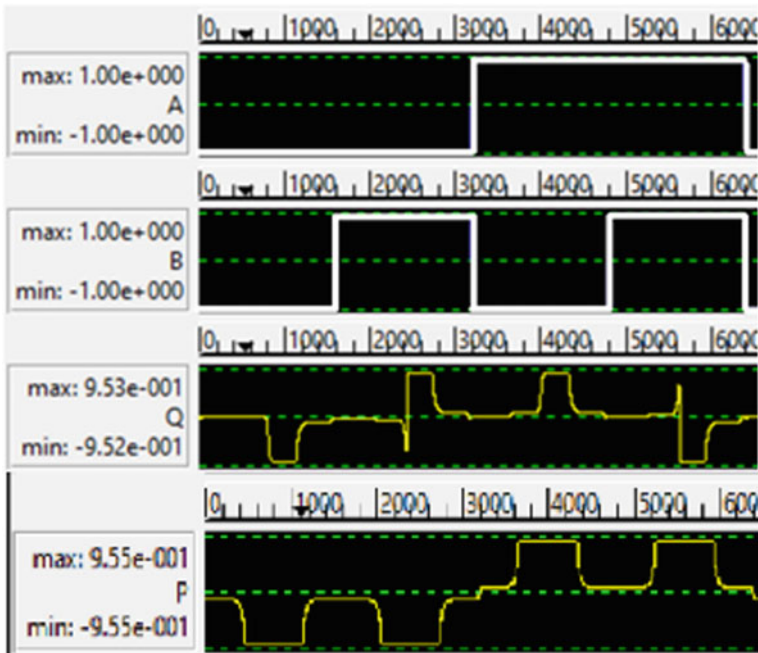


Fig. 17 Output waveform of Feynman gate

5 Simulation

(a) Feynman Gate

The wave touching the maximum value represents the binary digit of the “1” state, and the wave touching the minimum value represents the “0” state as shown in Fig. 17.

Since the output is similar to that of the XOR, the output waveform of Feynman gate can be easily matched with the truth table of the XOR gate shown in Fig. 8.

(b) XOR Gate

The final processed data in the device cell can be examined by considering the device cell as an output cell.

From Fig. 18, it can be observed that the device cell “a” stores the inverted data of both the inputs, i.e., if $A = 0$ and $B = 0$, then $a = 1$ which is indicated by a square wave.

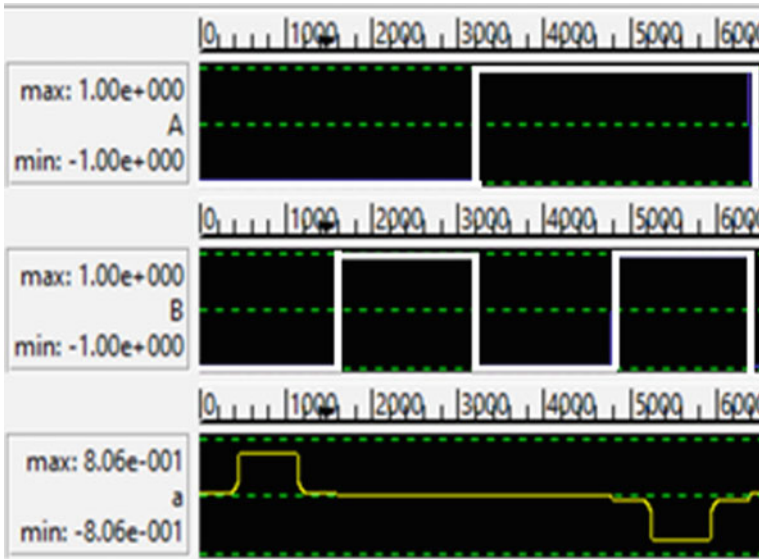


Fig. 18 Simulation output waveform

Table 1 Result analysis of Feynman and XOR gate in terms input and output cells, area, energy dissipation

Gate	No. of input cells	No. of output cells	No. of device cells	Area	Total no. of cells	Total energy dissipation	Average dissipation of energy
Feynman gate	2	2	1	0.02 μm^2	15	1.78E-2	1.62E-3
XOR gate	2	1	1	0.02 μm^2	14	1.5E-2	1.36E-3

6 Result

From Table 1, it can be seen that two gates (Feynman and XOR) have the same area, but different dissipation energies. According to the obtained stats, XOR gate is more suitable and efficient design but Feynman gate gains its advantage due to its reversibility concept.

7 Conclusion

Many complicated circuits rely on quantum gates for their implementation. The XOR gate, like the other digital gates, is an important gate in QCA. It is the only classical gate that is a replica of two or more quantum gates. The XOR and Feynman gate

circuits are developed using the QCAD designer tool in this paper, and the exact working of the XOR gate, as well as data flow via each cell, is explained. The design is evaluated in terms of input and output cells, as well as area and energy dissipation. In the future, more attention will be placed on how device cells perform and how they process data. This will provide more information on how reversible circuits and a few sophisticated circuits function.

References

1. Lent CS, Douglas Tougaw P, Porod W, Bernstein GH (1993) Quantum cellular automata. *Nanotechnology* 4(1):49–57
2. Lent CS, Douglas Tougaw P (1997) A device architecture for computing with quantum dots. *Proc IEEE* 85(4):541–557
3. Sheikhaal S, Angizi S, Sarmadi S, Moaiyeri M, Sayedsalehi S (2015) Designing efficient QCA logical circuits with power dissipation analysis. *Microelectron J* 46:462–471. <https://doi.org/10.1016/j.mejo.2015.03.016>
4. Singh G, Sarin RK, Raj B (2016) A novel robust exclusive-OR function implementation in QCA nanotechnology with energy dissipation analysis. *J Comput Electron* 15:455–465. <https://doi.org/10.1007/s10825-016-0804-7>
5. Abdullah-Al-Shafi M, Shifatul M, Bahar AN (2015) A review on reversible logic gates and its QCA implementation. *Int J Comput Appl* 128(2):27–34
6. Tripathi D, Wairya S (2021) A cost efficient QCA code converters for nano communication applications. *Int J Comput Digit Syst*
7. Lent CS, Douglas Tougaw P (1993) Lines of interacting quantum-dot cells: a binary wire. *J Appl Phys* 74(10):6227–6233
8. Lent CS, Douglas Tougaw P, Porod W (1994) Quantum cellular automata: the physics of computing with arrays of quantum dot molecules. In: *Proceedings workshop on physics and computation, PhysComp'94*, pp 5–13. IEEE
9. Majeed AH (2017) A novel design binary to gray converter with QCA nanotechnology. *Int J Adv Eng Res Dev* 4(9)
10. Sridharan K, Pudi V (2015) *Design of arithmetic circuits in quantum dot cellular automata nanotechnology*. Springer International Publishing, Cham
11. Goswami M, Mondal A, Mahalat MH, Sen B, Sikdar BK (2019) An efficient clocking scheme for quantum-dot cellular automata. *Int J Electron Lett* 6:1
12. Liu W et al (2014) A first step toward cost functions for quantum-dot cellular automata designs. *IEEE Trans Nanotechnol* 13(3):476–487
13. Bahar A, Waheed S, Habib A (2014) A novel presentation of reversible logic gates in quantum-dot cellular automata (QCA). In: *2014 international conference on electrical engineering and information communication technology (ICEEICT)*, pp 1–6
14. Bahar N, Waheed S, Hossain N, Saduzzaman M (2017) A novel 3-input XOR function implementation in quantum dot-cellular automata with energy dissipation analysis. *Alex Eng J* 56:1–9. <https://doi.org/10.1016/j.aej.2017.01.022>
15. Feynman RP (1985) Quantum mechanical computers. *Opt News* 11(2):11–20
16. Bahar AN et al (2018) A novel 3-input XOR function implementation in quantum dot-cellular automata with energy dissipation analysis. *Alexandria Eng J* 57(2):729–738
17. Balakrishnan L, Godhvari T, Kesavan S (2015) Effective design of logic gates and circuit using quantum cellular automata (QCA). In: *2015 international conference on advances in computing, communications and informatics (ICACCI)*, 10 Aug 2015, pp 457–462. IEEE
18. QCADesigner 2.0. <https://qcadesigner.software.informer.com/2.0/>

VisionX—A Virtual Assistant for the Visually Impaired Using Deep Learning Models



Akula Bhargav Royal, Balimidi Guru Sandeep, Bandi Mokshith Das, A. M. Bharath Raj Nayaka, and Sujata Joshi 

1 Introduction

One of the beautiful gifts to the humans are their vision which plays an important role. It is crucial to us which helps in our daily life. But by thinking about the fact given by the World Health Organization (WHO), in 2018 about 1.3 billion people suffer from vision problems globally. Amongst them, approximately 39 million human beings are blind, and more or less 246 million humans have mild visual limitations.

The facts these days presented by means of [1] and world health organization proves that around 1.3 million people (1.96%) earth's total 7.7 billion people is visually impaired consequently there may be a bigger need to resolve for such ultimatum and this is justified by South African records. Because of visual impairment, one needs to depend on others for their daily needs or sometimes compromise because of that illness. But with the Technology and industrial revolution happening and Artificial Intelligence for automation we can develop an equipment which can help blind to do their daily tasks without depending on others [2]. Around the globe, there are 135 million visually impaired people out of which 45 million are blind people, visual disability have a great impact on one's life since they can't see anything and has to depend on others for doing their daily tasks [3]. In these days, modern high-tech world the need of independent dwelling is important for visually impaired people. They may live in their daily environment, but in strange and new environments they can't live easily without any manual aid [4].

Can you imagine? Taking help of others for doing using simple tasks. Some people can't walk without taking help from others [5]. They should depend on others for doing basic tasks as well. This problem is increasing too fast in many people, so researchers are developing in new technologies to assist and help such people. This

A. B. Royal · B. G. Sandeep · B. M. Das · A. M. Bharath Raj Nayaka · S. Joshi (✉)
Nitte Meenakshi Institute of Technology, Bangalore 560064, India
e-mail: sujata.joshi@nmit.ac.in

work aims at developing an assistive glass for helping this people which takes input from camera on user command and process it and give it in the form of audio response to the user.

2 Literature Survey

Even though there are apps for helping visually impaired it didn't completely solve the problem. So researchers came with idea of using camera integrated on glasses to ease the usability but it also didn't completely ease so we wanted to integrate a voice assistant which will ease them to do their jobs like reading, knowing about their environment classifying objects and identifying people. This helps visually impaired for doing their daily jobs without depending on others.

The authors in [6] proposed a system where user can capture images in the smart phone which are then processed by the OCR model and extracts text from the image which is the given out in the form speech using a text to speech module.

The system user interface consists of a login page, registration page and also displays the text extracted from the image [7]. There are many devices for helping visually impaired for having perception around environment using touch or sound but the text reading systems are still in development. The authors developed an OCR system for supporting visually impaired people and the system consists of an image scanning module which takes image and feature extraction module which extracts features by binarizing, segmentation which helps in segmenting different parts of the image into different segments and then the relevant features are extracted and finally the recognition system to perceive the textual content within the pics.

By considering the advancements in the recent technologies the authors developed an intelligent assistant chatbot e-book reading [8]. Here, the rest API's are used for analysing the different images which is powered by google cloud vision engine. The images then are classified into various classes like landmark detection, emblem detection, express content detection. They used cloud speak API for speech to text.

Feature extraction is the main step in object detection and text detection [9, 10]. The two ways of feature extraction methods, namely, scale invariant features transform technique (SIFT) and speeded up sturdy features (SURF) in which the sooner one was designed to fit images or objects of various scenes [11, 12]. The paper aims in developing a system that restores crucial function of vision which is perceiving about the surrounding environment. The system uses the object detection using feature extraction which is scale invariant feature transform (SIFT). The main idea is to generate a local gradient patch around the key point. Many stages of Gaussian filtered picture are subtracted to construct the difference of Gaussian image. For further calculation variations of Gaussian (DoG) for every octave locate extrema from DoG is considered.

The blind can receive the information about surroundings with help of sound where they implemented an algorithm for image recognition by sonar. In this, edge detection is the main and first step of algorithm which is used for preprocessing [13].

To reduce noise associated with the images, applying a Gaussian blur to a picture is just like convolving the photograph with a Gaussian function. It is because of the reality that a Fourier transform of a Gaussian is also another Gaussian, hence making use of a Gaussian blur has the impact of lowering the picture's excessive-frequency components.

With a purpose to extract text from then photo it is critical to recognize properties of the image and additionally textual content in it [14, 15]. We will see that textual content extraction from synthetic or document photograph is less complicated than scene image because scene image can be blended with noise and blur. Textual content detection techniques can be divided as area-based, connected-component based and texture-based techniques wherein edge based techniques use exceptional mask like Sobel, Roberts, Prewitt and Canny to discover edges from photograph. Connected component-based method uses graph algorithm, wherein text is taken into consideration as mixture of related additives.

In another work proposed by the authors, a smart walking stick is designed which uses the Raspberry Pi 3b+ as a principal micro controller, global Positioning system (GPS) and ultrasonic sensors [16, 17]. The GPS system is used for navigation and guidelines for his destination whilst ultrasonic sensors have been used for locating barriers within the environment via soundwaves. The output is given through headset linked via Bluetooth.

3 Proposed Method

The proposed system helps visually impaired by detecting different objects and text present in the surroundings. It helps visually impaired in reading books as well as different texts present in the images.

The architectural design of the system is shown in Fig. 1, where Esp32 camera module is used for capturing images and the proposed system is based on a client-server model. In the proposed method the Esp32 is made as Bluetooth server and the mobile application is made as client.

Here, the images are taken from the camera module and are then passed onto to mobile application, where the processing is made using object recognition model and pytesseract for OCR, and the result is given as speech.

3.1 Hardware

Esp32

Esp32 is a low-cost integrated board system on chip (SoC) microcontroller. It is an upgraded version of ESP8266 SoC and is shown in Fig. 2.

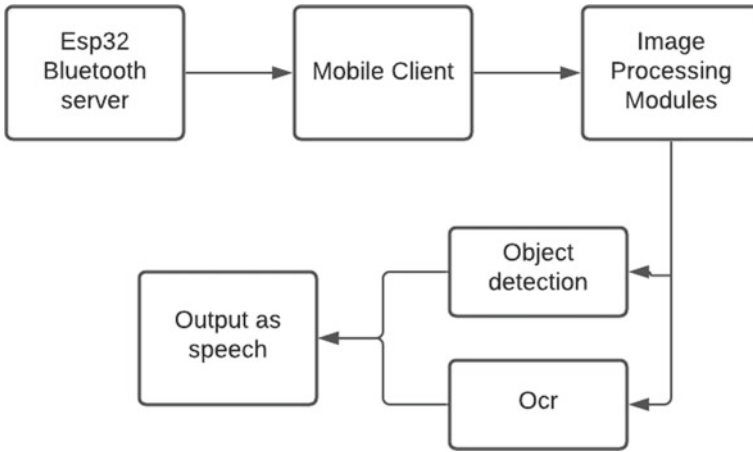
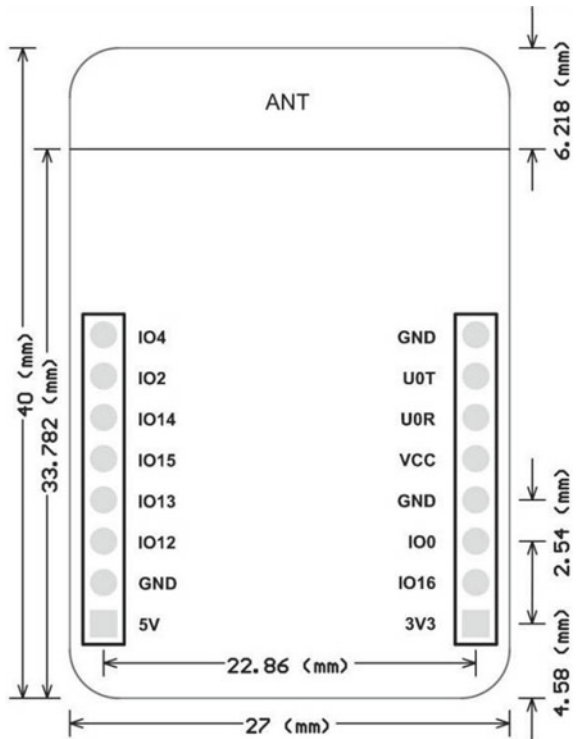


Fig. 1 Architectural design of the proposed system

Fig. 2 Block diagram of Esp32-CAM



It has dual core microprocessor with integrated Bluetooth and Wi-Fi. Esp32 board supports code dumps to make hardware work as it is expected to and can be programmed using Arduino IDE, Platform IO IDE, MicroPython and other such IDE's. Esp32 has a secure boot and flash encryption making it a bit more secure than its previous versions.

Esp32 Camera Module

The Esp32-CAM is a totally small digicam module which has the Esp32-S chip which costs approximately \$10 besides the OV2640 camera, and numerous GPIOs to attach peripherals, it additionally functions a microSD card slot is very useful here since we can store the required photographs concerned with the digicam or to keep documents to serve to clients.

FTD1

The FTDI USB to TTL serial converter module is a universal asynchronous receiver-transmitter (UART) board used for TTL serial communication. It is a breakout board for the FTDI FT232R chip with a USB interface, can use 3.3 or 5 V DC and has Tx/Rx and different breakout factors. The block diagram in Fig. 3 shows FTD1 module.

FTDI USB to TTL serial converter modules are used for popular serial applications. It is popularly used for communication to and from microcontroller development boards consisting of ESP-01s and Arduino micros, which do not have USB interfaces.

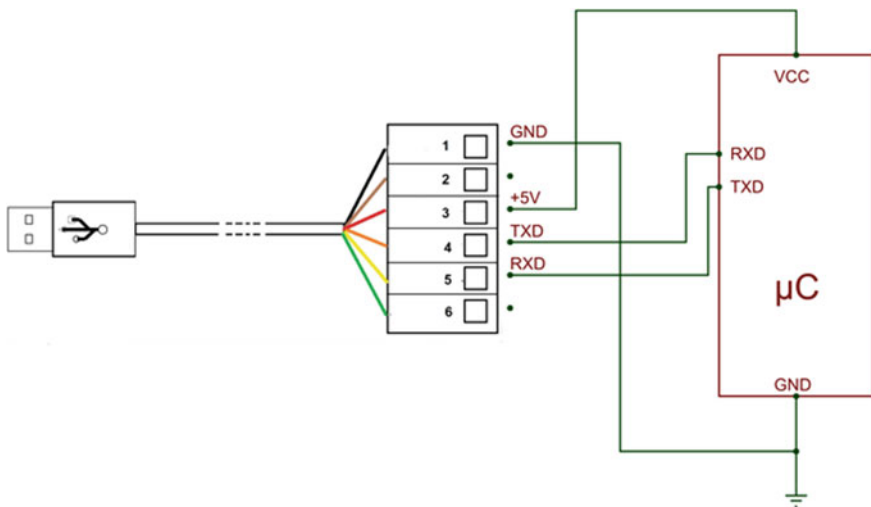


Fig. 3 Block diagram of FTD1

3.2 Dataset

Microsoft Common Objects in Context (COCO) is the dataset used inside the object detection segment. The primary version of MS COCO dataset was launched in 2014. It includes 164 K images cut up into training (83 K), validation (41 K) and test (41 K) units. In 2015 extra test set of 81 K pics became launched, along with all of the previous test pics and 40 K new pictures. Primarily based on community remarks, in 2017 the training/validation split became changed from 83 K/41 K to 118 K/5 K. The new break up makes use of the equal pictures and annotations. The 2017 take a look at set is a subset of 41 K pictures of the 2015 test set. Moreover, the 2017 launch consists of a new unannotated dataset of 123 K pictures.

3.3 Methodology

3.3.1 Input

The input of the system is taken from the Esp32 and with Bluetooth it is taken to the mobile application. Here, we are creating a server in the Esp32 and creating a client in the mobile along with an interface to access the image files on request. The problem with flutter and Esp32 is flutter takes images in the form of bytes, so we need to synchronize and keep delays for loading the image so that we can store image in the directory structure.

Here, the Esp32 camera module is made as a Bluetooth server which is responsible for capturing images and then passes onto mobile client where the various models are run on the mobile where the output is given as speech.

Various models in the system are as follows.

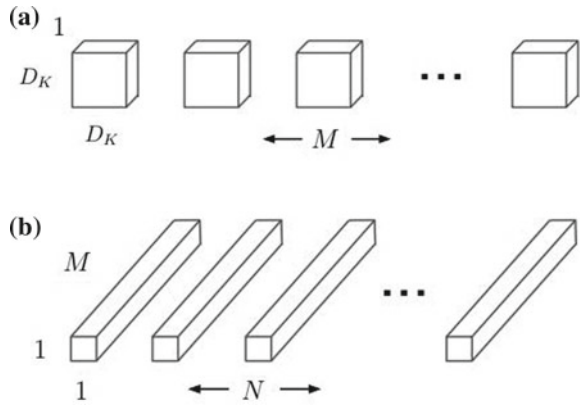
3.3.2 Object Detection

MobileSSDNet

The mobile net structure has two layers depth-wise convolutions and point-wise separable convolutions. The depth-wise separable convolutions are a shape of factorized convolutions. This model standardizes convolutions into depth-wise convolutions while 1×1 convolutions into point-wise convolutions. In those sort of neural nets depth-wise convolution applies a single filter to every input channel. The Pointwise convolution then applies a 1×1 convolution to mix the outputs. The depth-wise convolutions split this into two layers: a separable layer for filtering and a separable layer for combining. This factorization has the effect of notably decreasing the computation and model length.

The depth-wise convolutions unlike traditional convolutions splits the multiplication operation into sub operations like $1 \times n$ and $n \times 1$ from $n \times n$ thus reducing

Fig. 4 **a** Depthwise convolution. **b** Pointwise convolution



significant amount of computations (Fig. 4a). The main operation $n \times n$ is split into $1 \times n$ and $n \times 1$ which is reduced. This operation does not have significant change on a small dataset but for a large dataset containing more features and more rows it has reduced more costly computations. For example consider a computation involving 3×3 filter the depth wise separable convolutions splits into 3×1 and 1×3 filters thus reducing the computations from 9 operations into 6 operations.

The point-wise convolutions in Fig. 4b, are a type of convolutions which uses a 1×1 kernel, a kernel that iterates through every point. It has a depth which is equal to the number of channels the input is having. It is used along with depth-wise convolutions to form a depth-wise separable convolutions. The depth-wise separable convolutions can then be used as layers for the mobile net convolutions which are then used for object detection since images have large number features this can reduce the cost of computations.

OCR

Optical character recognition (OCR) is an ability of a computer to detect different characters present in the images or video scenes. In the last few decades, many researchers have developed many different techniques to perform this task. The OCR is already in use in many sectors like banking like automatic filling of forms etc. The OCR has different steps involved as the text is scattered into many segments of the images. The segmentation is required since the characters are widely spread in an image it is necessary to group them into meaningful chunks which makes the detected characters into a meaningful text. In the recent years, many companies are providing various OCR engines for OCR and Pytessarct by Google is the most efficient and most accurate OCR engine present.

Optical character recognition (OCR) is a method that is used to discover varies characters present within the photograph. Optical character recognition or optical character reader is the digital or mechanical conversion of pictures of typed, hand-written or published textual content into device-encoded textual content, whether or not from a scanned record, an image of a file, a scene-picture or from subtitle textual

content superimposed on a picture. In the proposed system, the OCR is performed using tesseract engine by google which is the most accurate OCR engine.

Speech to Text

Speech to text is an ability of a computer or an electronic device such as mobile, tab, laptop where in the device can convert the text into the form of speech so that the users can give commands which are then converted into text and are then used in our system to perform object detection based on the received commands.

The commands that are used in the system are used to connect the mobile application with external hardware, i.e., Esp32 Camera module which then captures image and then passes the image to application in the form of bytes and are then saved into file after taking the image. Then we have a command to recognize the image which is used to process the image and it is sent to both the object detection and OCR modules and if the object is detected it is given I the output else it return nothing and similarly the OCR module does the same. A Flutter library that exposes tool specific speech recognition capability. This plugin incorporates a set of classes that make it smooth to apply the speech recognition abilities of the underlying platform in Flutter. It helps Android, iOS and web. The target use cases for this library are instructions and brief phrases, no longer non-stop spoken conversion or always on listening.

Text to Speech

Text to speech is an ability of a computer to convert text into speech. This text to speech is used in various contexts like voice assistants and many more applications. In our system, this module helps in converting the received text into speech.

This flutter tts plugin used to interact with local capability. Underneath the hood, it uses text to speech for Android, and A speech synthesiser for IOS platform. In this, we're exploring the strategies of flutter_tts plugin. To check what we will achieve by means of this plugin.

4 Results

The proposed system uses Esp32 Cam for taking images and uses deep learning models and tesseract for performing object detection and Optical character Recognition.

The model used in the research is mobile SSD net which has an accuracy of 97% on training set and 88% on test set when Sigmoid and Tanh activation functions were used and is shown in Fig. 5. A comparison is also made with the Sigmoid and RELU activation functions as shown in Fig. 6.

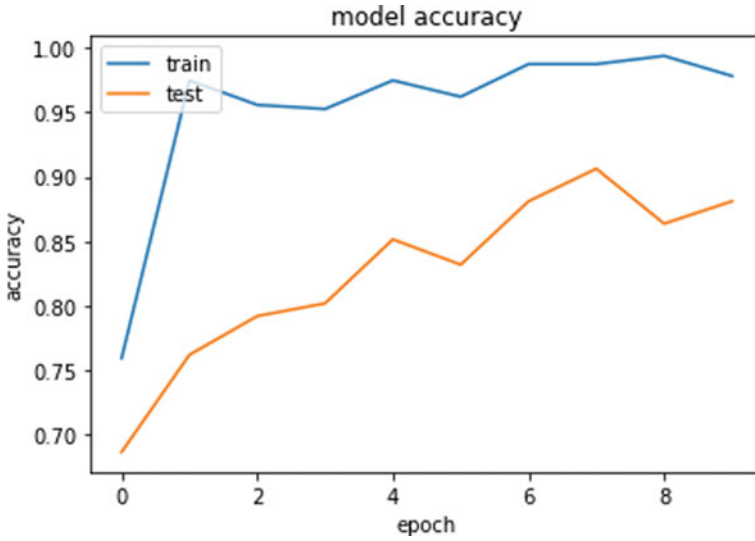


Fig. 5 Model accuracy using sigmoid and tanh activation

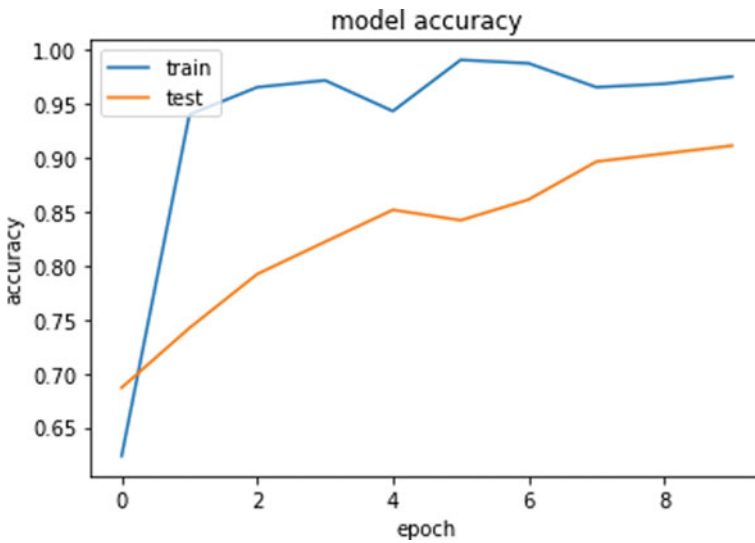


Fig. 6 Model accuracy using sigmoid and RELU activation

5 Conclusion

The research is mainly focused on helping visually impaired using deep learning algorithms and Pytesseract OCR Engine. The proposed system consists of an Esp32

camera module and also has a voice assistant which takes input in form of images and gives output in the form of speech.

The research focuses on helping visually impaired by identifying different objects and also helps in reading different text present in the images and also identifying objects in real time.

References

1. Mathur A, Pathare A, Sharma P, Oak S (2019) AI based reading system for blind using OCR. In: 2019 3rd international conference on electronics, communication and aerospace technology (ICECA), pp 39–42. IEEE
2. Gopinath J, Aravind S, Chandran P, Saranya SS (2015) Text to speech conversion system using OCR. *Int J Emerg Technol Adv Eng* 5(1)
3. Felix SM, Kumar S, Veeramuthu A (2018) A smart personal AI assistant for visually impaired people. In: 2018 2nd international conference on trends in electronics and informatics (ICOEI), pp 1245–1250. IEEE
4. Jabnoun H, Benzarti F, Amiri H (2014) Object recognition for blind people based on features extraction. In: International image processing, applications and systems conference, pp 1–6. IEEE
5. Jabnoun H, Benzarti F, Amiri H (2015) Object detection and identification for blind people in video scene. In: 2015 15th international conference on intelligent systems design and applications (ISDA), pp 363–367. IEEE
6. Gopala Krishnan K, Porkodi CM, Kanimozhi K (2013) Image recognition for visual impaired people by sound. In: International conference on communication and signal processing
7. Panchal AA, Varde S, Panse MS (2016) Character detection and recognition system for visually impaired people. In: 2016 IEEE international conference on recent trends in electronics, information & communication technology (RTEICT), pp 1492–1496. IEEE
8. Shandu NE, Owolawi PA, Mapayi T, Odeyemi K (2020) AI based pilot system for visually impaired people. In: 2020 international conference on artificial intelligence, big data, computing and data communication systems (icABCD), pp 1–7. IEEE
9. Yi C, Tian Y (2014) Scene text recognition in mobile applications by character descriptor and structure configuration. *IEEE Trans Image Process* 23(7):2972–2982
10. Chinchole S, Patel S (2017) Artificial intelligence and sensors based assistive system for the visually impaired people. In: 2017 international conference on intelligent sustainable systems (ICISS), pp 16–19. IEEE
11. Saeed NN, Salem MAM, Khamis A (2013) Android-based object recognition for the visually impaired. In: 2013 IEEE 20th international conference on electronics, circuits, and systems (ICECS), pp 645–648. IEEE
12. Gaudissart V, Ferreira S, Thillou C, Gosselin B (2004) SYPOLE: mobile reading assistant for blind people. In: 9th conference speech and computer
13. Bourbakis NG, Kavvaki D (2001) An intelligent assistant for navigation of visually impaired people. In: Proceedings 2nd annual IEEE international symposium on bioinformatics and bioengineering (BIBE 2001), pp 230–235. IEEE
14. Oak SA, Vidhate A (2016) Improved duplicate address detection for fast handover mobile IPv6. In: The international conference on computing communication, control and automation, IEEE section (ICCUBEA)
15. Koustriava E, Papadopoulos K, Koukourikos P, Barouti M (2016) The impact of orientation and mobility aids on way finding of individuals with blindness: verbal description vs. audio-tactile map. In: International conference on universal access in human-computer interaction, pp 577–585. Springer, Cham

16. Varma M, Zisserman A (2003) Texture classification: are filter banks necessary? In: Proceedings of 2003 IEEE computer society conference on computer vision and pattern recognition, vol 2, p II-691. IEEE
17. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: European conference on computer vision, pp 818–833. Springer, Cham

Analyzing the Performance of Novel Activation Functions on Deep Learning Architectures



Animesh Chaturvedi, N. Apoorva, Mayank Sharan Awasthi, Shubhra Jyoti, D. P. Akarsha, S. Brunda, and C. S. Soumya

1 Introduction

Deep learning is becoming increasingly popular in a variety of fields, including medicine, the military, and aerospace. Deep Learning delivers the highest accuracy when trained on large amounts of data, which contributes to its widespread appeal. It has a significant benefit, and this plays a major role in explaining its widespread popularity. Now, the “Big Data Era” [1] presents a plethora of options to develop new technologies and promote its appeal. From efficient inference on small data [2], Deep learning evolved in a manner comparable to the human nervous system, operating on huge data sets, solving a range of problems from estimation and forecasting [3] to classification that includes Self-driving vehicles, translation, and revolutionary medical treatments. These are testimony of deep learning capabilities and its broad range of exploration possibilities. The unique activation functions are being utilized and investigated on various data sets for exoplanet classification [4, 5]. When compared to the standard ReLU and Sigmoid Activation Functions, these activation functions SBAF parabola, AReLU, SWISH, and LReLU performed incredibly well on Vanilla Neural Networks and provided close to 99% accuracy on various datasets. It will be fascinating to observe if these activation functions perform similarly well for Deep Learning architectures such as CNN [6], DenseNet, Imagenet, and so on.

A. Chaturvedi · N. Apoorva · M. S. Awasthi · S. Jyoti · D. P. Akarsha · S. Brunda ·
C. S. Soumya (✉)
Nitte Meenakshi Institute of Technology, Bangalore, India
e-mail: soumya.cs@nmit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
N. R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication
and Applications*, Lecture Notes in Electrical Engineering 928,
https://doi.org/10.1007/978-981-19-5482-5_76

903

2 Activation Functions

Activation functions serve as a link between the data sent to the input layer and the neuron presently in use, as well as the outcomes sent to the final output layer. Neuron activation is determined by computing the weighted sum of activation functions and then adding bias to the total [7]. Neurons with similar information are triggered. The activation of neurons is governed by a set of principles. A set of rules governs the firing of neurons. The main purpose of activation functions is to bring non-linearity into the system. Forward propagation passes the results of activation functions to the next layer. An error is computed if the output value differs considerably from the true value. The process is then referred to as reverse propagation. Consider the neural network shown in Fig. 1.

For linear activation functions, the above network layers can be pictured as-Layer1:

$$\implies Y(1) = W(1)M + b(1)(1)c(1) = Y \tag{1}$$

where

$Y(1)$ is the output of layer 1.

$W(1)$ denotes the weights matrix for input to hidden layer neurons, i.e., $w_1, w_2, w_3,$ and w_4 .

M denotes the inputs i_1 and i_2 .

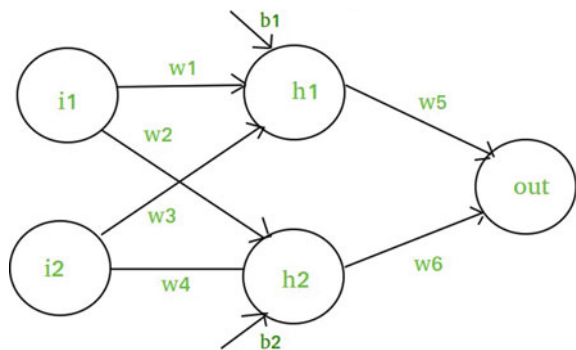
b denotes the vectored bias assigned to hidden layer neurons, i.e., b_1 and b_2 .

$c(1)$ is the vectored sort of any linear function.

Similarly Layer 2:

$$\implies Y(2) = W(2)c(1) + b(2)(2)c(2) = Y \tag{2}$$

Fig. 1 Example of neural network



Now, substituting (1) in (2) we get,

$$\begin{aligned} \implies Y(2) &= (W(2) * [W(1)M + b(1)]) + b(2) \\ &= [W(2) * W(1)] * M + [W(2) * b(1) + b(2)] \end{aligned} \tag{3}$$

Let $[W(2) * W(1)] = w$ and $[W(2) * b(1) + b(2)] = b$, $[W(2) * W(1)] = w$ and $[W(2) * b(1) + b(2)] = b$,

$$\therefore Y(2) = w * M + b$$

The result obtained is a linear function. If activation functions do not appear to be applied, then the equation of weights and bias must be linear. Solving linear equations is straightforward, however it has a limited capability in determining complex difficulties that require complex understanding. Any neural network that lacks an activation function represents a linear regression model.

It is advantageous to employ non-linear activation functions, which aid in the analysis of complicated data, computation, and learning, as well as the generation of proper outputs. Despite the fact that there are many layers in the network, using a linear activation function yields another linear equation since concatenating them yields another linear equation, which is insufficient to enhance the model. Because the derivative of the linear activation function yields a constant, it is not feasible to go back and checkout neurons. There are a number of activation functions used in shallow and deep neural network architectures. Few of them are discussed below.

2.1 Rectified Linear Unit (ReLU)

ReLU seems to be a linear activation function, yet it allows the network to converge fast. However, because there is a derivative, backpropagation is attainable. Because a model that utilizes it is quicker to train and generally produces higher performance, it has become the default activation function for many types of neural networks.

$$f(x) = x^+ = \max(0, x) \text{ where } x \text{ is the input to a neuron.}$$

This is also known as a ramp function.

2.2 Approximation to Rectified Linear Unit (AReLU) [8]

AReLU is the generalized version of the activation function ReLU used in this study. It is further shown analytically and empirically that the approximation to ReLU, designated as AReLU, is constant and discrete at the “knee-point,” and that AReLU

does not quite require significant parameter adjustment. Extensive experience demonstrates unambiguously that the insights derived from the mathematical theory are supported by performance metrics that outperform ReLU. AReLU is written as:

$$y = kx^n$$

2.3 Saha-Bora Activation Function (SBAF) [8]

SBAF is the generalized version of the Sigmoid activation function in this study. Furthermore, Banach space theory and contraction mapping have been used to demonstrate that SBAF may be viewed as a solution to a first order differential equation. This is analyzed by comparing system utilization metrics such as runtime, memory, and CPU consumption. The function has acceptable analytical properties and does not appear to have any local oscillation problems. Saha-Bora Activation Function is formulated as:

$$y = \frac{1}{1 + kx^\alpha(1 - x)(1 - \alpha)}$$

2.4 Swish [9]

The first thing that comes to mind while looking at this layout is that it looks quite a bit like ReLU, with one exception: the domain about 0 is not the same as ReLU. It is a non-obtrusive function. That is, unlike ReLU, it does not abruptly shift direction near $x = 0$. However, it bends smoothly from 0 and higher and then back upwards. As a result, unlike ReLU and the other two activation functions, it does not remain steady or move in one direction and is also non-monotonic. According to the authors, it is this trait that distinguishes Swish from most other activation functions, which share this uniformity. Swish Activation Function is uninterrupted at all points. It is termed as-

$$y = \text{swish}(x) = x\sigma(\beta x)\sigma(x) = 1/(1 + \exp(-x))$$

here $\sigma(x)$ is the sigmoid function and β can either be a constant defined prior to the training or a parameter that can be trained during training time. The derivative of the Swish Activation function is

$$f'(x) = \beta f(\beta x) + \sigma(\beta x)(1 - \beta f(\beta x))$$

2.5 Scaled Exponential Linear Unit (SeLU)

SeLU causes neural networks to have a self-normalizing feature. The activations of neurons converge toward a zero mean and unit variance. Because SeLUs have such a high level of self-normalization, we no longer have to be concerned with disappearing gradients. It is formulated as

$$\begin{aligned} \text{if } a > 0 & : \text{ return scale} * a \\ \text{if } a < 0 & : \text{ return scale} * \alpha * (\exp(a)) \end{aligned}$$

Here, alpha and scale are constants where alpha = 1.673 and scale = 1.0507. The values for alpha and scale are chosen in such a manner that the mean and variance of the values supplied as input are maintained throughout all layers and the inputs are appropriately big.

2.6 Exponential Linear Unit(ELU)

ELU, which stands for Exponential Linear Unit, is an activation function with comparable functionality to ReLU but less differences. ELU conducts resilient deep network training, resulting in higher classification precision. In comparison to other activation functions, ELU includes a saturation function for dealing with the negative section. When the unit is disabled, the activation function is reduced, causing ELU to execute quicker in the presence of noise. ELU does not have the issue of disappearing and bursting gradients. It is uninterrupted and distinguishable at all places.

$$\begin{aligned} y = \text{ELU}(x) &= \exp(x) - 1; \quad \text{if } x < 0 \\ y = \text{ELU}(x) &= x; \quad \text{if } \geq 0 \end{aligned}$$

2.7 Mish

Mish is a self-regularized non-monotonic activation function inspired by Swish's self-gating feature. In Long Short-Term Memory (LSTM)s and Highway Networks, self-gating is the use of the sigmoid function. It is an uninterrupted, smooth, non-monotonic activation function. It can be formulated as:

$$\begin{aligned} f(x) &= x \tanh(\text{softplus}(x)) \\ &= x \tanh(\ln(1 + e^x)) \end{aligned}$$

$$f'(x) = \frac{e^x \omega}{\delta^2}$$

where

$$\omega = 4(x + 1) + 4e^2x + e^3x + e^x(4x + 6) \text{ and } \delta = 2e^x + e^2x + 2$$

3 Deep Neural Network Architectures

The design of neural network is similar to human nervous system. It comprises one input layer, two or more hidden layers, and finally one output layer. There are various types of neural networks like feed-forward, CNN, RNN, etc. Different types of network have their own business purpose.

3.1 Convolution Neural Networks

CNN is the most successful model in the field of image processing. It has accomplished great results in image classification, recognition, semantic segmentation, and machine translation, and can freely learn and extract features of images. CNN has solved or partially solved the issues of low performance, lack of actual images, and segmented operation of traditional machine learning methods. The important advantage of CNN model is that they can extract features without applying segmented operation while obtaining satisfactory performance. Features of an object are consequently extracted from the original data. Kunihiko Fukushima introduced the Neocognitron in 1980, which propelled CNNs. The development of CNNs has made the innovation progressively productive and automated.

The first and foremost step is to feed the dataset into a CNN classifier to complete the classification task. It uses different activation functions referenced in the review. Our CNN architecture includes two convolutional layers (one convolutional operation and one max pooling operation), one fully connected layer and one output layer.

The CNN architecture shown in Fig. 2, includes one input image, two convolutional pooling layers, one fully connected layer, and one output layer. The sizes of the convolutional filters and of the pooling operation are 5×5 and 2×2 , respectively. The first layer has three convolutional filters, the subsequent layer has five convolutional filters, and the fully connected layer has 50 units. In all classification experiments, we set the learning rate to 0.01 and the size of a batch is 100. The same architecture is run on different datasets and activation functions. The CNN architecture is chosen to limit the computational complexity and retain the classification accuracy.

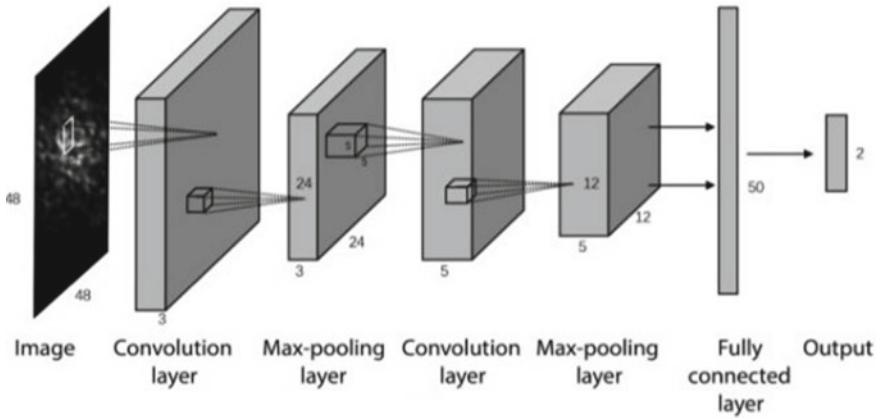


Fig. 2 CNN Architecture

3.2 DenseNet

Densely Connected Convolutional Networks, i.e., DenseNet, are the new choice for many of the datasets. When we try to analyze, we find that CNN has flaws. Because the route for information to go from the initial input layers to the final output layer (and for the gradient in the other direction) is longer and larger, they disappear before reaching the other side of the layer. Residual Network (ResNet) also a type of Convolution Neural network architecture that makes it possible to construct networks with up to thousands of convolutional layers, which outperform shallower networks. DenseNet is similar to ResNet but achieves more precision. In other networks, the results of the preceding layer are linked to the next layer via a sequence of processes. Convolution operations on pooling layers, batch normalization, and an activation function are often included in the collection of procedures. DenseNet, do not combine the layer’s feature maps with the input features of other layers, but rather concatenate them. The equation reshapes again into:

$$x_l = H_l(x_0, x_1, x_2, \dots, x_{l-1})$$

DenseNet are seen as groups of DenseBlocks, with the number of filters changing but the size of the maps remaining constant. Transition layers are the layers that exist between these blocks. Normalizing layers by re-centering and re-scaling inputs stabilizes neural networks. The preceding layers’ results are normalized by executing 1×1 convolution and 2×2 pooling layers. We concatenate the outputs of all layers here, thus dimensions are increased at all levels. The generalization at the k th layer following H_1 that produces ‘ m ’ features every time is known as:

$$m_l = m_0 + mX(k - 1), \text{ heremisgrowthrate.}$$

The following layers add additional functionality to the layers that came before them. Transition blocks execute 1×1 convolution with filters, followed by 2×2 pooling with a stride of 2, resulting in a 50% decrease in feature maps and size. As we progress through the network, the number of levels increases ‘ m ’ times. Initially, 1×1 convolution is conducted using 128 filters, followed by 3×3 convolution. This is accomplished through the usage of 32 feature maps. This is known as the growth rate.

Finally, each layer in each denseblock performs the identical set of operations on the input, and the resulting outputs are concatenated. Each layer contributes fresh information to the ones that came before it.

4 Dataset

We have used a combination of computer vision datasets in this paper. These benchmark datasets are obtained from UCI Machine Learning Repository. We have experimented on Intel I5 2nd generation 8250U Processor (1.6–3.4 GHz Turbo Boost) 8 GB RAM and 1 TB Hard Drive, with 2 GB Dedicated AMD Graphics Card. The following datasets were used-

Cifar 10: The CIFAR-10 Dataset, as the name implies, contains images from ten distinct categories. There are 60,000 images in 10 distinct categories, including Airplane, Automobile, Bird, Cat, Deer, Dog, Frog, Horse, Ship, and Truck.

The images are all 32×32 pixels in size. There are 50,000 train images and 10,000 test images in all.

Cifar 100: CIFAR 100 is identical to CIFAR 10, except it has 100 classes with 600 images in each class. There are 500 training images and 100 testing images in each class. CIFAR 100 consists of 100 classes divided into 20 super classes. Each picture has a “fine” label and a “coarse” label. Fine labels define the class to which they belong, whereas coarse labels identify the superclass to which they belong.

MNIST: MNIST stands for Modified National Institute of Standards and Technology. This collection includes 60,000 tiny square 28×28 pixel grayscale pictures of handwritten single numerals ranging from 0 to 9.

Fashion MNIST: Fashion MNIST is a dataset that contains 60,000 examples of the training set and 10,000 examples of the test set, both of which are photographs from Zolando’s articles. Each example is a 28×28 grayscale picture with a label from one of ten classifications.

5 Methodology

The goal of this research is to evaluate the performance of bespoke activation functions on various deep neural architectures such as CNN and DenseNet. Because these activation functions worked remarkably well on vanilla neural networks, it will be interesting to anticipate the model's loss and accuracy using these activation functions on deep neural architectures. The proposed system consists of the subsequent steps:

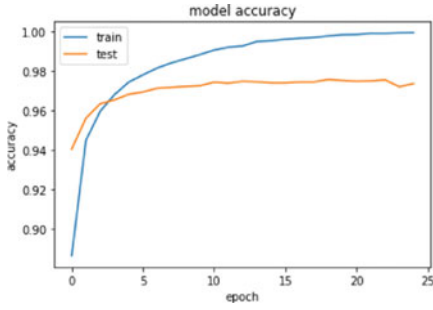
- Step 1: Determine the datasets on which the model must be trained.
- Step 2: Create train-test data partitions.
- Step 3: Perform one hot encoding on train and test datasets.
- Step 4: Create a predictive model with the CNN and DenseNet architectures.
- Step 5: Test the model and tune hyper-parameters.
- Step 6: Use categorical CrossEntropy as loss function and Adam as optimizer.
- Step 7: Loss and accuracy values are plotted.

Model Architecture: The model essentially recreates a network that functions similarly to neurons in our brain. The network is taught to make predictions based on previously collected data. In this study, we investigated the performance of various activation functions using deep neural networks such as CNN and DenseNet. Categorical CrossEntropy, a robust and popular loss function utilized in most of the classification problems, was computed during each execution. There are various objectives to complete, one of which is that an example can only belong to one of the potential categories. The model determines the category it is in here.

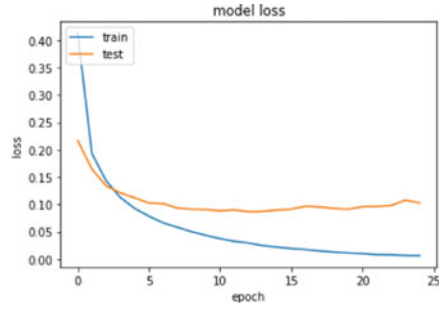
6 Performance Analysis

All of the following activation functions have been evaluated on various networks such as CNN and DenseNet, as well as on various datasets such as MNIST, Cifar10, Fashion MNIST, and so on. We have a variety of activation functions, and it is difficult to select one that is appropriate for all test situations. Many aspects come into play, including whether or not it is differentiable, how quickly a neural network with a particular activation function converges, how smooth it is, whether it fits the constraints of the universal approximation theorem, and whether or not normalization is retained.

As we have seen that we have used different datasets on CNN and DenseNet architectures with multiple activation functions. Here are a few plots that show the performance of the ReLU and AReLU activation functions (Figs. 3, 4, 5 and 6).

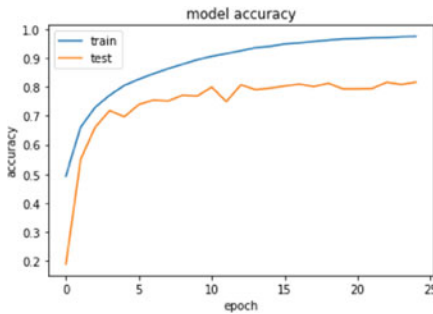


(a) CNN MNIST ReLU accuracy

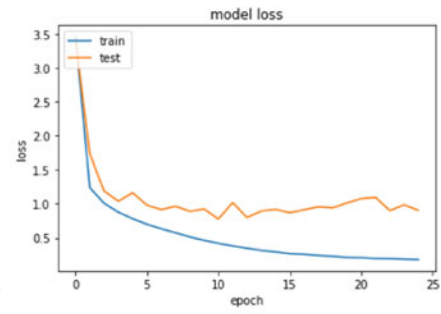


(b) CNN MNIST ReLU loss

Fig. 3 ReLU was passed on CNN over MNIST dataset, with an accuracy of 0.9734 and had loss of 0.1025

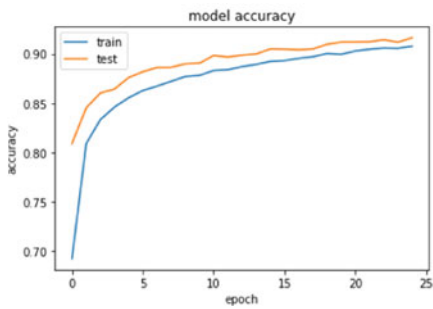


(a) DenseNet CIFAR10 ReLU accuracy

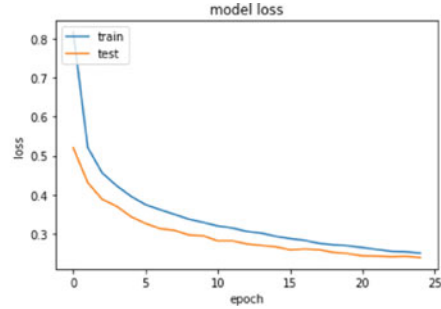


(b) DenseNet CIFAR10 ReLU loss

Fig. 4 ReLU was passed on DenseNet over CIFAR10 dataset, with an accuracy of 0.8083 and had loss of 0.9868

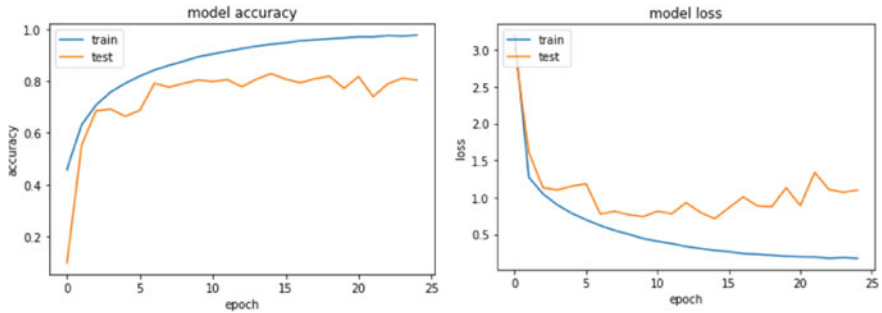


(a) CNN FashionMnist AReLU accuracy



(b) CNN FashionMnist AReLU loss

Fig. 5 AReLU accuracy showed upto 0.9162 and a loss of 0.2386 on fashion MNIST dataset



(a) DensNet cifar10 AReLU accuracy (b) DensNet cifar10 AReLU loss

Fig. 6 AReLU was tested on DenseNet over CIFAR10 dataset, and accuracy was upto 0.7769 and had loss of 1.24

7 Conclusion

The purpose of this research is to demonstrate the performance of various activation functions and compare their performance across datasets and architectures. It may be summed up as follows: Because ReLU is not differentiable at $X = 0$, the Gradient will be 0. During Back-Propagation, we compute gradient as well as the gradient of Activation Functions. If the derivative equals zero, the gradient equals zero, there will be no weight update, and the network will finally stop learning. A-derivative ReLU's is not equal to 0 and is differentiable at $x = 0$. AReLU is more tenacious than ReLU. AReLU was performing similarly to ReLU, and in some cases outperformed ReLU (Table 1).

ReLU showed up better performance with DenseNet rather than CNN architecture. Ideally for all the datasets and activation functions, DenseNet architecture showed better performance compared to CNN. While comparing activation functions with different datasets, Mish Activation function worked well on MNIST, ReLU worked well on CIFAR10 dataset, for the CIFAR100 dataset none of the activation functions performed well but among all ELU gave a better result. All the activation functions could perform better on Fashion MNIST dataset with average accuracy of 92.27% except for SBAF and SBAF Parabola activation functions.

Table 1 Performance analysis of activation functions over CNN and DenseNet

Performance parameters	Architecture			CNN			DenseNet		
	Activation function	MNIST	CIFAR-10	CIFAR-100	FashionMNIST	MNIST	De	CIFAR-100	FashionMNIST
Accuracy	ReLU	0.9734	0.7809	0.481	0.9239	0.9939	0.9051	0.4762	0.9249
	AReLU	0.9746	0.7867	0.4686	0.9162	0.9927	0.8028	0.4607	0.9269
	Swish	0.9735	0.7612	0.4855	0.9191	0.9878	0.8164	0.4924	0.9273
	Mish	0.9738	0.7711	0.4535	0.9194	0.9934	0.8139	0.4781	0.9207
	ELU	0.9765	0.7761	0.4457	0.9164	0.9192	0.8152	0.5135	0.9113
	SELU	0.973	0.7492	0.3973	0.9031	0.9898	0.8061	0.4935	0.9251
	SBAF	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN
	SBAF parabola	0.9759	0.1	0.01	0.7229	0.1009	0.1	0.01	0.1
	Absolute function	0.9754	0.7709	0.3363	0.8976	0.4994	0.5268	0.5193	0.8878
	ReLU	0.1025	0.7136	2.0615	0.2155	0.0719	0.8169	2.6083	0.3698
Loss	AReLU	0.2015	0.7615	2.062	0.2386	0.0686	1.1031	3.399	0.3656
	Swish	0.108	1.0686	2.4678	0.2332	0.1106	1.0618	3.2292	0.4038
	Mish	0.1013	1.0312	2.7021	0.2341	0.0793	1.1285	3.3057	0.5011
	ELU	0.0953	0.9142	2.7263	0.2417	0.0808	0.8714	2.6364	0.442
	SELU	0.1071	0.9346	2.6897	0.2729	0.104	0.8873	2.5875	0.3161
	SBAF	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN
	SBAF parabola	0.1654	2.5871	4.679	0.6913	3.0135	2.9098	5.0057	3.0092
	Absolute function	0.1119	0.6761	2.5586	0.2825	2.6472	2.1403	2.4566	0.4565

References

1. Ginge G et al (2015) Mining massive databases for computation of scholastic indices: model and quantify internationality and influence diffusion of peer-reviewed journals. In: Proceedings of the 4th national conference of institute of scientometrics, SIoT, pp 1–26
2. Anisha RY et al (2017) Early prediction of LBW cases via minimum error rate classifier: a statistical machine learning approach. In: IEEE international conference on smart computing (SMARTCOMP), pp 1–6
3. Saha S et al (2016) DSRS: estimation and forecasting of journal influence in the science and technology domain via a lightweight quantitative approach. *Collnet J Sci Inf Manage* 10(1):41–70
4. Safonova M et al (2021) Quantifying the classification of exoplanets: in search for the right habitability metric. *Euro Phys J Spec Top* 230(10):2207–2220
5. Basak S et al (2020) CEESA meets machine learning: a constant elasticity earth similarity approach to habitability and classification of exoplanets. *Astron Comput* 30:100335
6. Ravikiran M et al (2018) TeamDL at SemEval-2018 Task 8: cybersecurity text analysis using convolutional neural network and conditional random fields. *SEMEVAL
7. Hebbar PA et al (2022) Theory, concepts, and applications of artificial neural networks. In: *Applied soft computing*. Taylor & Francis, p24
8. Saha S, Mathur A, Bora K, Basak S, Agrawal S (2018) A new activation function for artificial neural net based habitability classification. In: 2018 international conference on advances in computing, communications and informatics (ICACCI), 2018, pp 1781–1786. <https://doi.org/10.1109/ICACCI.2018.8554460>
9. Ramachandran P et al (2017) Swish: a self-gated activation function. *Neural Evol Comput: n pag*. arXiv
10. Basak S, Mathur A, Theophilus AJ et al (2021) Habitability classification of exoplanets: a machine learning insight. *Eur Phys J Spec Top* 230:2221–2251. <https://doi.org/10.1140/epjs/s11734-021-00203-z>
11. Mohapatra R et al (2021) AdaSwarm: augmenting gradient-based optimizers in deep learning with swarm intelligence. In: The IEEE transactions on emerging topics in computational intelligence. <https://doi.org/10.1109/TETCI.2021.3083428>
12. Yedida R, Saha S (2021) Beginning with machine learning: a comprehensive primer. *Euro Phys J Spec Top* 230:2363–2444. <https://doi.org/10.1140/epjs/s11734-021-00209-7>
13. Prashanth T et al (2021) LipGene: Lipschitz continuity guided adaptive learning rates for fast convergence on Microarray Expression Data Sets." *IEEE/ACM transactions on computational biology and bioinformatics*; <https://ieeexplore.ieee.org/document/9531348>
14. Saha S et al (2021) DiffAct: a unifying framework for activation functions. In: International joint conference on neural networks (IJCNN), pp 1–8
15. Mediratta I et al (2021) LipARELU: ARELU networks aided by Lipchitz acceleration. In: 2021 international joint conference on neural networks (IJCNN), pp 1–8
16. Sarkar J et al (2014) An efficient use of principal component analysis in workload characterization-a study. *AASRI Procedia* 8:68–74
17. Yedida R, Saha S (2019) A novel adaptive learning rate scheduler for deep neural networks. *ArXiv*, abs/1902.07399
18. Makhija S et al (2019) Separating stars from quasars: machine learning investigation using photometric data. *Astron Comput* 29:100313
19. Sridhar S et al (2020) Parsimonious computing: a minority training regime for effective prediction in large microarray expression data sets. In: 2020 international joint conference on neural networks (IJCNN), pp 1–8
20. Saha S et al (2018) A new activation function for artificial neural net based habitability classification. In: 2018 international conference on advances in computing, communications and informatics (ICACCI), pp 1781–1786

Hybrid Model for Stress Detection of a Person



A. C. Ramachandra, N. Rajesh, K. Mohan Varma, and C. R. Prashanth

1 Introduction

Stress is the body's response to any Physical, Mental or Emotional pressure. Stress has negative effects on human body and is classified into two types Acute stress and Chronic stress. Acute stress generally lasts for short interval of time with high intensity of stress whereas chronic stress effects the person over a long period of time. During the Period of stress body releases adrenaline and hormone called cortisol also known as stress hormone. Adrenaline is responsible for increase in the blood pressure, heart rate and Cortisol is responsible for increase in blood sugar levels. The response system for stress is usually self-limiting and when the body access the threat has passed hormone levels drop and blood pressure and heart rate comes to baseline. But when stressors are always present the body constantly feels under attack and long-term activation of stress response and high cortisol levels will lead a person to be diabetic as the body is unable to produce high insulin levels continuously.

Long term elevation in Blood Pressure is harmful and may lead to Heart failure. Apart from these it also leads to feelings like anger, sad, fear or depression and can push a person into complete mental illness such as BPD. Which causes mood swings and unstable behavioral patterns in some stressful events like traveling in public transport, cope with the work pressure in stressful working environments, etc. A clinical study conducted by Oka, Department of psychosomatic medicine from Kyushu University Japan. They have conducted a clinical study [1] where

A. C. Ramachandra (✉) · N. Rajesh
Nitte Meenakshi Institute of Technology, Bengaluru 560064, India
e-mail: ramachandra.ac@nmit.ac.in

K. Mohan Varma
L&T Technology Services, Vadodara, India

C. R. Prashanth
Dr. Ambedkar Institute of Technology, Bangalore 560056, India

they mainly concentrated on how stress effects core body temperature. During their study on lab rats whenever these rats are placed in some unfamiliar environments or in some of the stressful events, they have noticed the raise in body temperature which they have termed as Psychogenic fever which is stress related fever. Some people who are highly stressed their body temperature reached 41 °C. They have concluded that psychological stress increases the body temperature of a person. A public dataset collected by the Alessio Rossi. In this dataset [2] they have collected various Physiological responses from 22 People who have volunteered to provide data. Time period for collecting the data was 24 h and parameters included heartrate, sleep quality, physical activities and emotional status (anxiety status and stressful events). In this study most of the people is having higher heart rate variation during the time interval they have stated that they have undergone high workload comparatively with that of not working conditions.

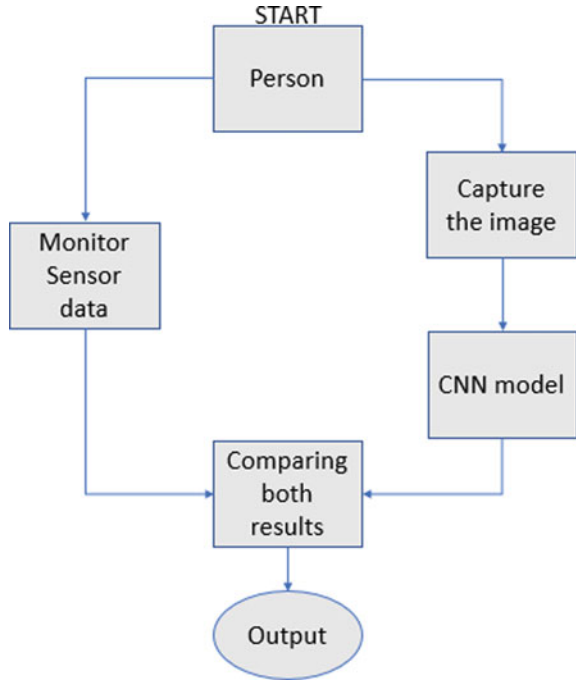
Consistent stress due to workload will have adverse effect on a person’s mental well-being. It also contributes for unstable behavior and physical breakdown amid people of various ages. High stress due to workload increases fatigue and chances of cancer. Many people lack in identifying stress as it is very challenging to detect. By the time people decide to take medical help from doctors they might be affected very badly with noticeable ailments. So, to avoid this there is requirement to detect stress in early stages. Many methods [3, 4] use a questionnaire-based surveys to detect stress of an individual in this type of surveys the person has to spend needful time to take such surveys. Some systems used only Facial emotion-based stress detection which is not reliable [5–7]. Thus, these systems are not suitable for the Modern-day approach and sometimes even these surveys might not be effective in determining stress [8–11]. There is a need to develop a system which is reliable in predicting stress of a person by taking into account the parameters which are more likely to be affected in a stressful condition.

The aim is to make a Hybrid system that monitors some of the parameters which can be used to predict stress such as Temperature, Heartbeat and facial Emotion of a person. So, the objective of the proposed system will be collecting and monitoring the Temperature, heartbeat of a person with the help of Microcontroller and detecting the facial emotion of that person using pre-trained CNN model and Table 1 shows the list of Emotions taken for training.

Table 1 List of emotions taken for training

Emotion	Images
Angry	3993
Fear	3205
Happy	7164
Neutral	4982
Sad	4938

Fig. 1 Conceptual flow of the proposed system



Total of 24,282 image of 5 emotion classes will be used to train the model. Finally, by comparing both the outputs from Microcontroller and CNN model the system will decide whether a person is stressed or not stressed as shown in Fig. 1.

2 Proposed System Architecture

The Overall architecture of the system is represented in Fig. 2. Hardware unit consists of Temperature sensor and Pulse sensor for monitoring the corresponding temperature and heartbeat of a person. Both the sensors are connected to a microcontroller board based on AT mega 328P. In Software unit we are using python [12–15] for training a model which is capable of predicting stress based on emotion. The training model will run in Software unit which is used to capture a frame from the video stream and using the trained model and input sensor data from the microcontroller the proposed system predicts the person is stressed or not stressed. Open CV is being used for processing the image captured by the camera.

Using built in camera an image is captured, processing of the captured image is done by using Python. Machine learning algorithms and Computer vision are the strategies utilized by the checking framework for computations and used for real-time

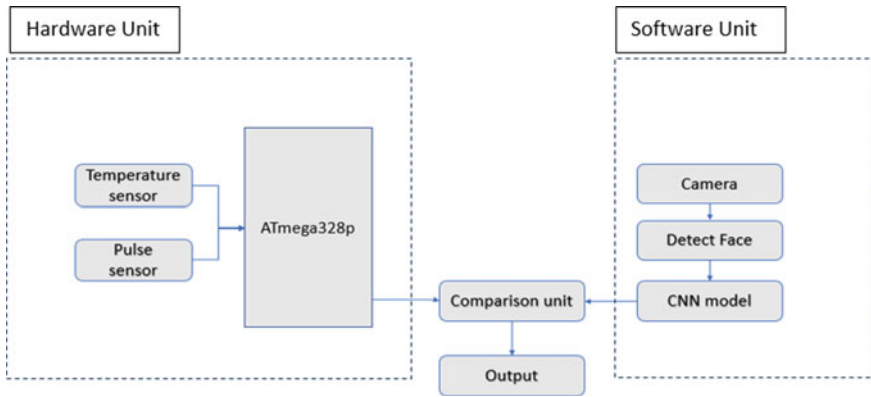


Fig. 2 System architecture

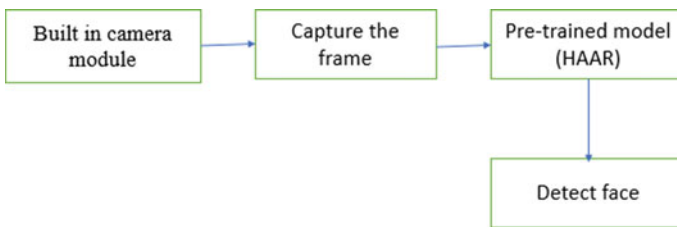


Fig. 3 Face detection overview

application. Haar [16] feature-based cascade classifiers are used for Face Detection [17, 18]. Figure 3 shows the different stages of face Detection system.

Detected face will act a Region of Interest for the CNN model which detects the expression of the person and decide whether a person is stressed or not. When eyes are wide open the person is in fear, when eyebrows shrink with eyes slightly closed the person is angry, when a person head position is slightly downwards with closed lips, he is sad. All these Emotions are used to detect stress. As only facial emotion of a person is not reliable in deciding whether the person is stressed or not parameters such as temperature of a person and heartrate of a person is taken into account. Temperature sensor used is TMP36 contact type sensor, it is a low voltage precision centigrade temperature sensor produces a voltage output that is linearly proportional to temperature. The accuracy of the sensor is typically ± 2 °C with operating range of 2.7–5.5 V. It uses the property of the diodes and measures the small changes and outputs an analog voltage between 0 and 1.75 V DC. TMP 36 has an offset of 500mv which is equal to 0 °C. For 10 mv change in voltage = 1 °C change in Temperature.

- Temperature = (mv – 500)/10

The pulse sensor operating voltage range is between 3.3 and 5 V which contains an LED and a photo diode for detecting the heartbeat. This sensor has two surfaces,

first surface the light-emitting diode and light sensor is connected. On the second surface, the circuit is connected which is responsible for the noise cancelation and amplification.

Calculation of Heart rate from Inter Beat Interval (IBI):

The pulse sensor will calculate the heart rate based on the interval time between two successive peaks in the heart rate R–R. The unit of IBI can be both millisecond (ms) and in second (s). Since we need heart rate per minute, we convert the milliseconds to seconds ($60 \times 100 = 60,000$ ms) for example if IBI from R–R is 650 ms.

- Heart rate = $(60,000/IBI)$
- Heart rate = $(60,000/650)$
- Heart rate = 92 beats/min.

Microcontroller receives the data from the sensors. Camera is used to capture a frame from the video stream and face is detected using HAAR classifier and detected face will now act as a Region of Interest (ROI). The CNN model predicts the facial emotion the label generated from the Microcontroller which is read by the Python using serial UART protocol. The result is compares based on the outputs from the microcontroller and CNN model.

3 Proposed System Flow

The hardware side of the system and the software side of the system will start simultaneously as shown in Fig. 4. These sensors are attached to the index finger of a person.

Hardware side consisting sensors are connected to microcontroller for monitoring the temperature and heartbeat of a person. The threshold value for heartbeat is set as 80 beats/min and Temperature as 38 °C. Whenever the threshold value is reached the system generates a label 'X' which will be read by the software part using UART protocol and stores in a data variable. Similarly, the software part uses Anaconda and python with OpenCV for capturing the frame, resizing the image as a part of pre-processing. The image is given to pre-trained model to predict a person is stressed or not stressed based on the emotion of a person. If the label 'X' is generated and stress is predicted by the CNN model the output is given as stress has been detected.

4 Implementation

For the classification of emotion in a photo we use a convolution neural network. Here we implement CNN using pre-trained Mobile net [19] neural network architecture using Tensor Flow and Open CV in python platform as shown in Fig. 5.

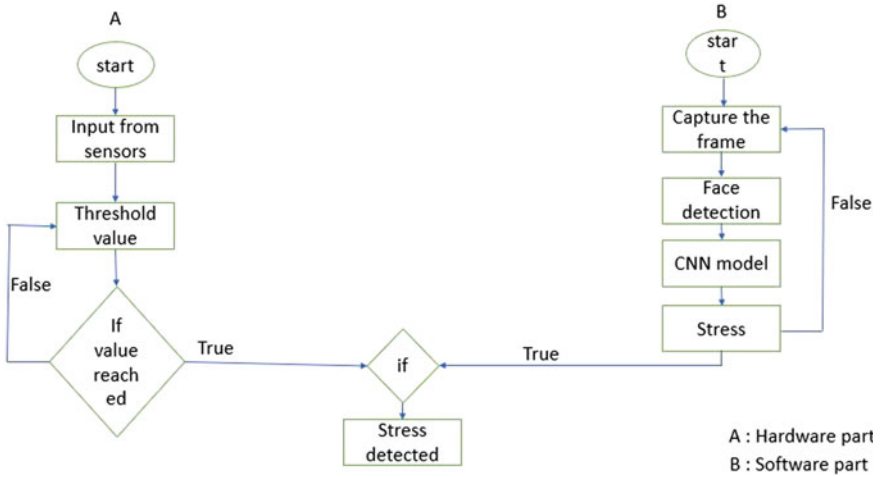


Fig. 4 System flow

```
for layer in MobileNet.layers:
    layer.trainable = True

# Let's print our layers
for (i,layer) in enumerate(MobileNet.layers):
    print(str(i),layer.__class__.__name__,layer.trainable)

def addTopModelMobileNet(bottom_model, num_classes):
    """creates the top or head of the model that will be
    placed ontop of the bottom layers"""

    top_model = bottom_model.output
    top_model = GlobalAveragePooling2D()(top_model)
    top_model = Dense(1024,activation='relu')(top_model)

    top_model = Dense(1024,activation='relu')(top_model)

    top_model = Dense(512,activation='relu')(top_model)

    top_model = Dense(num_classes,activation='softmax')(top_model)

    return top_model

num_classes = 5

FC_Head = addTopModelMobileNet(MobileNet, num_classes)

model = Model(inputs = MobileNet.input, outputs = FC_Head)

print(model.summary())
```

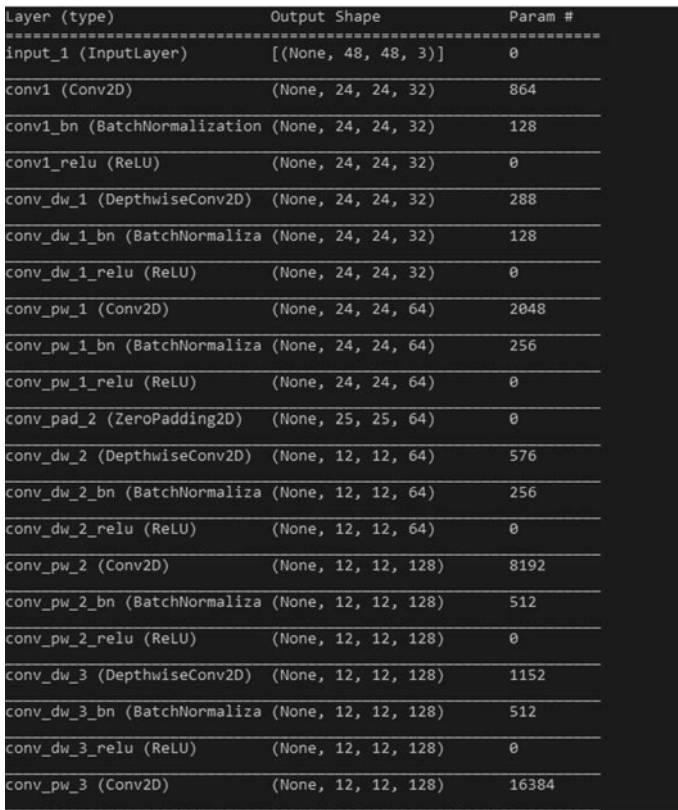
Fig. 5 Adding mobile net layers

Mobile Net is a convolutional neural network that tends to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depth wise convolutions to filter features as a source of non-linearity. As a whole, the architecture of Mobile Net contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers. Mobile Net layers are shown in Fig. 6.

The presented CNN learning methodology revolves around the Kaggle fer_2013 dataset. It contains different emotions, separated into training and testing dataset. The training dataset consists of 24,282 images and testing dataset consists of 5937 images of 5 different classes which are angry, fear, happy, neutral and sad.

Steps in training:

- The dataset is imported from the system, which contain various emotions and is broken into two parts, i.e., training and validation.



Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 48, 48, 3)]	0
conv1 (Conv2D)	(None, 24, 24, 32)	864
conv1_bn (BatchNormalization)	(None, 24, 24, 32)	128
conv1_relu (ReLU)	(None, 24, 24, 32)	0
conv_dw_1 (DepthwiseConv2D)	(None, 24, 24, 32)	288
conv_dw_1_bn (BatchNormaliza)	(None, 24, 24, 32)	128
conv_dw_1_relu (ReLU)	(None, 24, 24, 32)	0
conv_pw_1 (Conv2D)	(None, 24, 24, 64)	2048
conv_pw_1_bn (BatchNormaliza)	(None, 24, 24, 64)	256
conv_pw_1_relu (ReLU)	(None, 24, 24, 64)	0
conv_pad_2 (ZeroPadding2D)	(None, 25, 25, 64)	0
conv_dw_2 (DepthwiseConv2D)	(None, 12, 12, 64)	576
conv_dw_2_bn (BatchNormaliza)	(None, 12, 12, 64)	256
conv_dw_2_relu (ReLU)	(None, 12, 12, 64)	0
conv_pw_2 (Conv2D)	(None, 12, 12, 128)	8192
conv_pw_2_bn (BatchNormaliza)	(None, 12, 12, 128)	512
conv_pw_2_relu (ReLU)	(None, 12, 12, 128)	0
conv_dw_3 (DepthwiseConv2D)	(None, 12, 12, 128)	1152
conv_dw_3_bn (BatchNormaliza)	(None, 12, 12, 128)	512
conv_dw_3_relu (ReLU)	(None, 12, 12, 128)	0
conv_pw_3 (Conv2D)	(None, 12, 12, 128)	16384

Fig. 6 Mobile net layers implimented

- We have imported the features of mobilenetv2 model for emotion classification using TensorFlow. We have built our model and added in front of the pre-trained Mobile net (model.h5).

Training model flow has been shown in Fig. 7. Training model accuracy and loss plot have been shown in Figs. 8 and 9.

In hardware we have sensors connected to microcontroller. The connections are shown in Fig. 10. Pin 1 of Temperature Sensor (TMP 36) is connected to 5v in microcontroller, Pin 2 to A1 (Analog Pin 1) and Pin 3 to GND (Ground). Pin 1 of pulse sensor is connected to GND, Pin 2 to 5 V and Pin 3 to A0 (Analog Pin 0).

The sensor values from the Temperature sensor and Heartbeat sensor are continuously monitored by the Microcontroller and when threshold value for Temperature (> 38) and heartbeat (> 80) is reached the microcontroller generates a label 'X' as shown in Fig. 11.

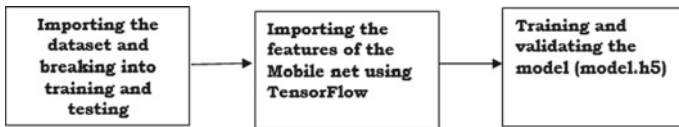


Fig. 7 Training model flow

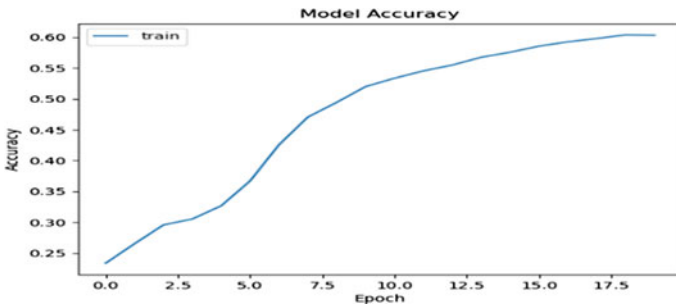


Fig. 8 Accuracy plot with batch size of 40 and 20 epochs

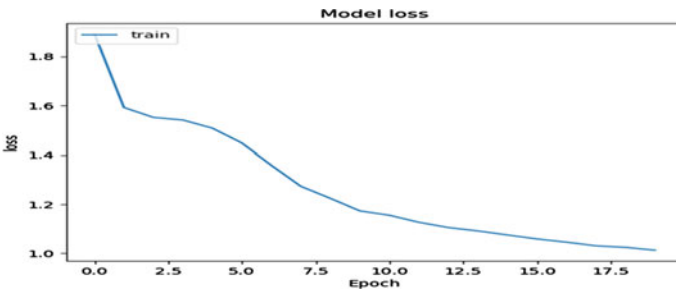


Fig. 9 Loss plot with batch size of 40 and 20 epochs

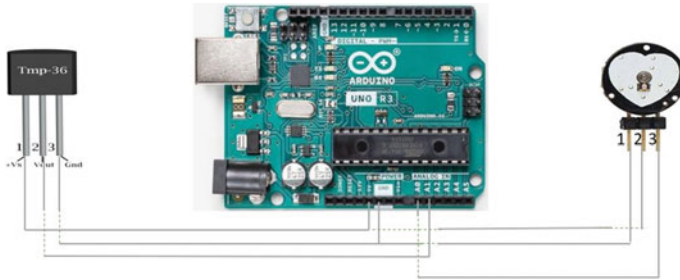


Fig. 10 Sensor connections

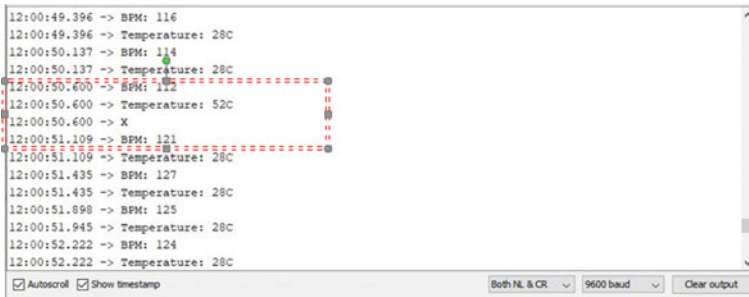


Fig. 11 Sensor values

In main Program the following steps will be done:

- The pre-trained model for face detection and emotion detection will be loaded and all the necessary libraries are imported.
- Capturing the Frame from the Video stream, resizing to 48 * 48 image and the image is given as an input to pre-trained Face detection algorithm there the detected face will become the region of interest (ROI).
- Reading from the serial port for the label and store the label in a variable *ch*.
- Comparing label from the serial port and emotion prediction, if emotion is angry, sad or fear and the label generated by the microcontroller is 'X' it gives the output as stress detected.

5 Results

During the time of 8:00 am to 8:30 am the person 1 and person 2 was resting and the corresponding body temperature and heartbeat was noted. During the time of 8:30 am to 11:00 am the person 1 and person 2 was working in a stressful environment and the corresponding body temperature and heartbeat was noted as shown in Tables 2 and 3.

Table 2 Tabulated sensor values of person 1

Time	Temperature (°C)	Heartbeat (per min)
8:00:00	36	65
8:15	34	66
8:30	32	67
8:45:00	34	66
9:00	35	71
10:00	38	82
10:15:00	39	80
10:30	38	86
10:45	39	70
11:00:00	38	90

Table 3 Tabulated Sensor values of person 2

Time	Temperature (°C)	Heartbeat (per min)
8:00:00	37	59
8:15	33	55
8:30	32	67
8:45:00	34	66
9:00	36	70
10:00	40	83
10:15:00	39	89
10:30	38	90
10:45	40	92
11:00:00	42	87

It is noted that the variation of body temperature and heart rate is less when a person is not working in the stressful environment and during working the body temperature and heartbeat was comparatively high than that in resting as shown in the graph Figs. 12 and 13.

Fig. 12 Variation in body temperature and heartbeat of person 1

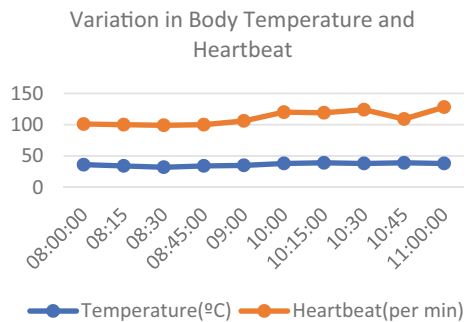
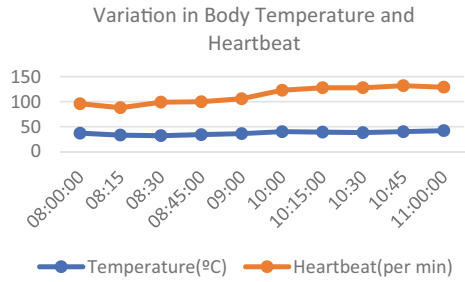


Fig. 13 Variation in body temperature and heartbeat of person 2



In Fig. 14 we can clearly differentiate the period where the person is experiencing stress due to excess of workload in stressful environment. When Temperature exceeds $> 38\text{ }^{\circ}\text{C}$ and Heartbeat > 80 beats/min label 'X' is generated by the microcontroller. Simultaneously when the CNN model Predicts emotion which are predefined as stressful emotions the system will display the output as Stress Detected as shown in Fig. 15.

- When CNN model predicts Emotion as Angry with Label 'X' generated from Microcontroller. The frame shows the person is stressed and related emotion as Shown in Fig. 16.
- When CNN model predicts Emotion as Fear with Label 'X' generated from Microcontroller. The frame shows the person is stressed and related emotion.
- When CNN model predicts Emotion as sad with Label 'X' generated from Microcontroller. The frame shows the person is stressed and related emotion.

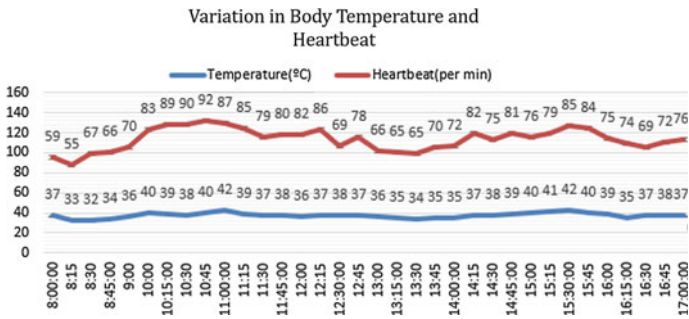


Fig. 14 Variation in body temperature and heartbeat throughout the day



Fig. 15 Display of the result



Fig. 16 Images taken for testing the algorithm

- When CNN model predicts Emotion as Happy with Label 'X' generated from Microcontroller. The frame shows NO-stress with related emotion.
- When CNN model predicts Emotion as Neutral with Label 'X' generated from Microcontroller. The frame shows NO-stress with related emotion.
- When CNN model predicts Emotion as Angry without Label 'X' generated from Microcontroller. The frame shows NO-stress with related emotion.

6 Conclusion

The proposed system is able to identify the emotion present in the image based on training labels. The system is accurately comparing the results from the Microcontroller and the CNN model. Further the system is able to display whether a person is stressed or not. The validation accuracy is around 69%. More data in dataset in future will help the classifier to produce the more accurate results as well as accuracy.

References

1. Oka T, Oka K, Hori T (2001) Mechanism and mediators of Psychological stress-induced rise in core temperature. *Psychosom Med* 63(3)
2. Rossi A et al (2020) A public dataset of 24-h multi-levels psycho-physiological responses in young healthy adults. *Data* 5(4):91
3. El-Samahy E et al (2015) A new computer control system for mental stress management using fuzzy logic. In: 2015 IEEE international conference on evolving and adaptive intelligent systems (EAIS). IEEE, 2015

4. Sandulescu V, Dobrescu R (2015) Wearable system for stress monitoring of firefighters in special missions. In: 2015 E-health and bioengineering conference (EHB). IEEE, 2015
5. Rachakonda L (2019) Stress-lysis: a DNN-integrated edge device for stress level detection in the IoMT, IEEE Trans Consum Electron 65(4):474–483
6. Ghosh A, Danieli M, Riccardi G (2015) Annotation and prediction of stress and workload from physiological and inertial signals. In: 2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, 2015
7. Sano A, Picard RW (2013) Stress recognition using wearable sensors and mobile phones. In: 2013 Humaine association conference on affective computing and intelligent interaction. IEEE, 2013
8. Rachakonda L et al. (2018) A smart sensor in the IoMT for stress level detection. In: 2018 IEEE international symposium on smart electronic systems (iSES)(Formerly iNiS). IEEE, 2018
9. Elias K et al (2019) Stress-log: An IoT-based smart system to monitor stress-eating. In: 2019 IEEE international conference on consumer electronics (ICCE). IEEE, 2019
10. Deshmukh RS, Jagtap V, Paygude S (2017) Facial emotion recognition system through machine learning approach. In: 2017 international conference on intelligent computing and control systems (ICICCS). IEEE, 2017
11. Rosa RL, Rodriguez DZ, Bressan G (2015) Music recommendation system based on user's sentiments extracted from social networks. IEEE Trans Consum Electron 61(3):359–367
12. Phuc LT, Jeon H, Truong NT, Hak JJ (2019) Applying the Haar-cascade algorithm for detecting safety equipment for safety management systems for multiple working environments. Electron J 8(10):1079
13. Kamencay P (2017) A new method for face recognition using convolutional neural network. Dig Image Process Comput Graph 14:663–672
14. Kuster PP (2018) Face detection and face recognition in python programming language. In: Proceedings of the 7th international conference on informatics and applications (ICIA2018), Japan, 2018
15. Manikandan J, Lakshmi Prathyusha S, Sai Kumar P, Jaya Chandra Y, Umaditya Hanuman M (2020) Face detection and recognition using open CV based on fisher faces algorithm. Int J Recent Technol Eng 8(5)
16. Li C et al (2017) Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. In: 2017 13th IEEE international conference on electronic measurement and instruments (ICEMI). IEEE, 2017
17. Pai VK, Balrai M, Mogaveera S, Aeloor D (2018) Face recognition using convolutional neural networks. In: IEEE conference record: # 42666; IEEE Xplore. ISBN: 978-1-5386-3570-4
18. Wasnik P et al (2016) Presentation attack detection in face biometric systems using raw sensor data from smartphones. In: 2016 12th international conference on signal-image technology and internet-based systems (SITIS). IEEE, 2016
19. Saxen F et al (2019) Face attribute detection with mobilenetv2 and nasnet-mobile. In: 2019 11th international symposium on image and signal processing and analysis (ISPA). IEEE, 2019
20. McEwen BS, Stellar E (1993) Stress and the individual: mechanisms leading to disease. Arch Intern Med 153(18):2093–2101

Machine Learning-Based Social Distance Detection: An Approach Using OpenCV and YOLO Framework



Deepthi Shetty, H. Sarojadevi, Onkar Bharatesh Kakamari, Savitha Shetty, Saritha Shetty, Radhika V. Shenoy, B. N. Rashmi, M. S. Sneha Dechamma, and G. Tanmaya

1 Introduction

Globally, the coronavirus disease (COVID-19) has spread rapidly, making social separation an increasingly important preventative measure. This paper focuses on building a surveillance system that combines deep learning, OpenCV, and computer vision to ensure pedestrian safety, avoid overcrowding, and maintain a safe distance between pedestrians.

Large crowds at the locations may exacerbate the existing scenario. Recently, all countries around the world had been, and still are, under lockdown, forcing inhabitants to stay at home. However, because this encourages people to visit more public areas, religious sites, and tourist locations, this technique of measuring social separation will be good all over the world in certain conditions. Limiting contact between

D. Shetty (✉) · H. Sarojadevi · O. B. Kakamari · R. V. Shenoy · B. N. Rashmi · M. S. Sneha Dechamma · G. Tanmaya
Department of CS and E, NMIT, Bengaluru, India
e-mail: deepthi.shetty@nmit.ac.in

H. Sarojadevi
e-mail: sarojadevi.n@nmit.ac.in

O. B. Kakamari
e-mail: 1nt19cs132.onkar@nmit.ac.in

S. Shetty
Department of CSE, NMAM Institute of Technology, Karkala, India
e-mail: shettysavi1@nitte.edu.in

S. Shetty
Department of MCA, NMAM Institute of Technology, Karkala, India
e-mail: shettysaritha1@nitte.edu.in

infected individuals and healthy individuals, or between populations with high transmission rates and populations with low transmission rates, COVID-19 transmission is reduced or disrupted by social distancing.

The rest of the paper is organized as follows. Section 2 is providing the system architecture. Section 3 focuses on design. Section 4 describes the implementation. Section 5 details the process and approaches. Section 6 lists a few test cases. Section 7 presents the results. Section 8 provides the impact of the work presented and the future scope. Lastly, Sect. 9 draws conclusions.

2 System Architecture

The crowd behaviour is examined using three tasks in the proposed FOB congestion control system: Object identification, tracking of the object itself and object movement tracking. Here, the object is a human head. [1]

The faster R-CNN architecture helps to achieve the objective of head detection by deploying the input frame on the pre-trained CNN model, such as GoogleNet's inception architecture [2]. "The region proposal network" (RPN) identifies the areas in the feature map that might contain the objects by creating the score and bounding boxes for the object proposals. Based on RPN's recommended regions and CNN's feature maps, the "region of interest (RoI)" pooling layer is then utilized to extract feature maps. Finally, using fully linked layers the bounding boxes are categorized and fine-tuned using the output feature map. If the closeness criterion is not satisfied, the item is handled as a novel object in crowd tracking. Every subsequent frame shows a gradual movement of a certain object, i.e., object tracking in crowds. This is demonstrated in Fig. 1.

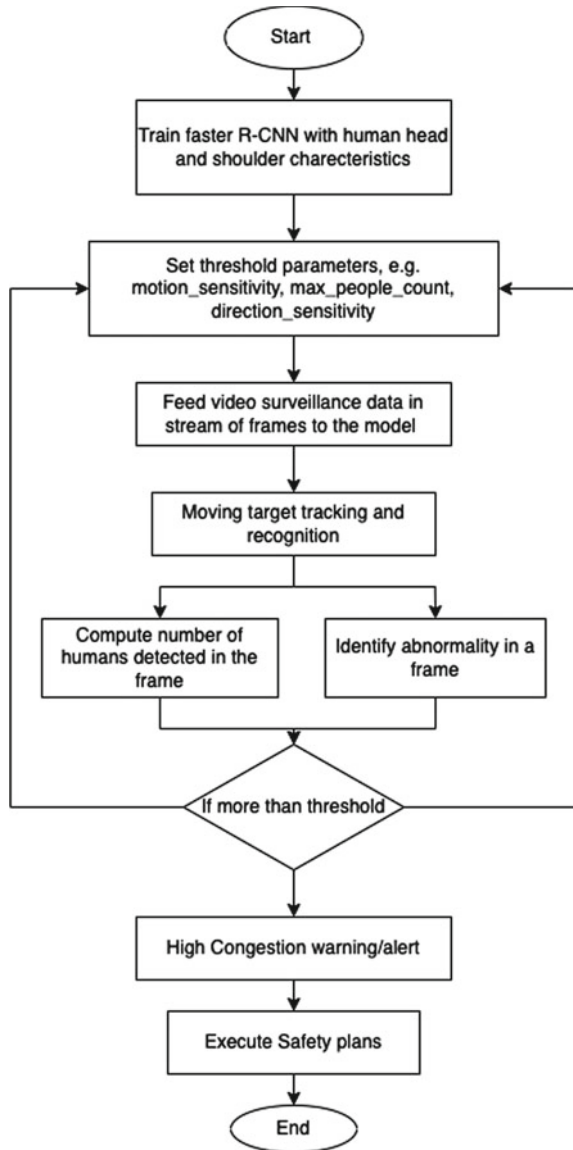
3 Design

The proposed notion is built using Python 3, OpenCV, and the Caffe framework in this study. Based on a segmented ROI flowchart, people can be detected for social distance and safety violation alerts. The framework for detecting objects is the most essential feature of this study. The primary goal of this system is to analyse collected footage filmed for human detection and then analyse it for social distancing violations. Techniques for image processing and the OpenCV Library are used here.

This study's object detection model is run using a deep learning model framework [3]. Because of the low execution time, the mobile net SSD model was chosen [4]. This is visually depicted in Fig. 2.

Parallelism in threads is exploited in this implementation. Using threading is a good idea to reduce the amount of time it takes for each frame to process object detection in this investigation. The frame will be run while the object detection is processed using a multithreading technique [5–7]. To determine the distance between

Fig. 1 System design for congestion control



two bounding boxes, the following strategy is adopted. The bounding box's centre point is used in this study to determine the distance between a pair of box's bounds locations. To calculate a bounding box's centre point, the centre of the moment at which the bounding boxes come together is used to tell the difference between a pair of two distinct bounding box positions [8].

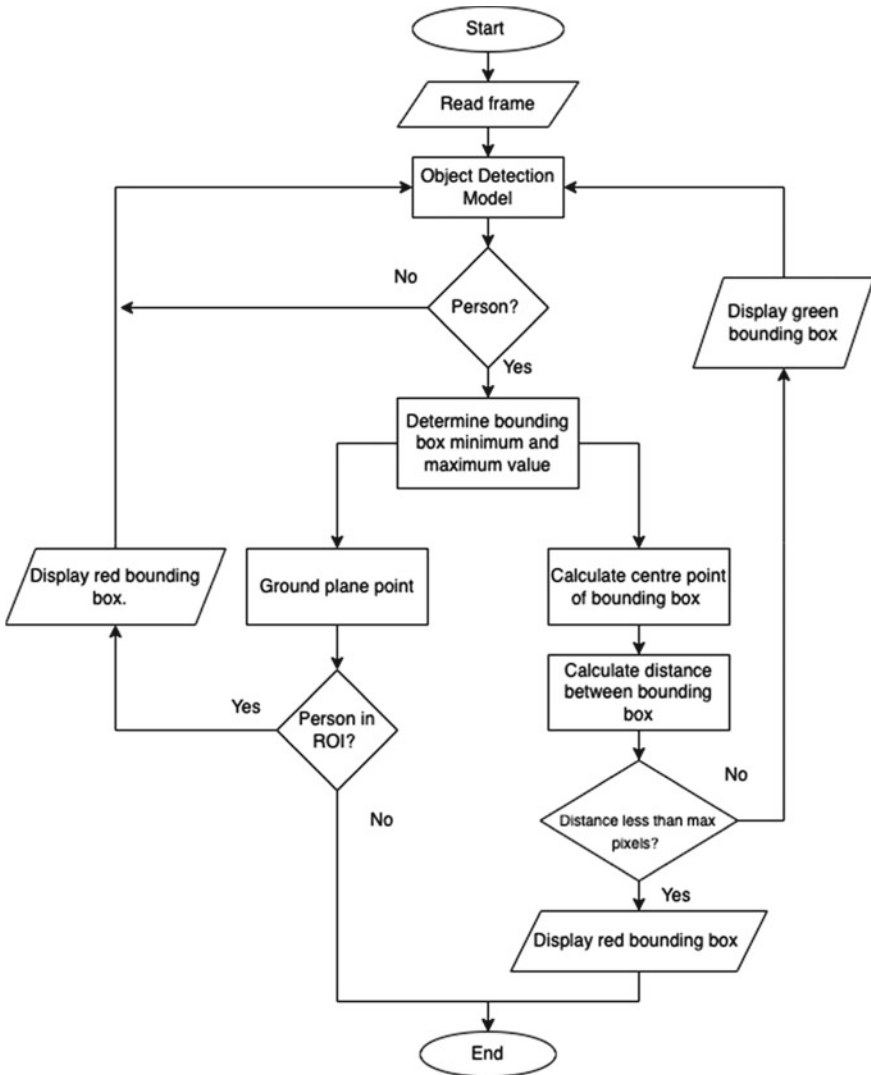


Fig. 2 Detection of social distance and safety violations based on the flow chart segmented by the region of interest

4 Implementation

4.1 Methodology

Using deep learning and computer vision algorithms along with OpenCV and the Tensor flow library. In the proposed framework, the focus is on recognizing individuals in video or image streams and whether or not social distance is preserved [9, 10].

YOLOv3 is used to detect people within the video. This calculates the gap between every person within the frame that has been observed. This follows a list showing how many people are considered high, low, and not in danger. To introduce social distancing detectors, deep learning and computer vision is used. To make a social distancing detector, below mentioned procedures are followed:

- Identify all the people in the video stream using object detection [11]. Calculate the pairwise distances between all of the individuals who are identified.
- Check whether any two individuals are N pixels apart. For object detection, YOLO is employed, using which a bounding box around the objects (people) are drawn once they have been identified.
- Calculate the distances between the boxes using the centroid of the boxes.
- The Euclidean distance is used to calculate the space. A box is coloured red if it is unsafe, and green if it is safe [12].

4.2 Description of the YOLO Framework

You only look once (YOLO) is a real-time convolutional neural network that can be used to detect objects in real time. Before segmenting the image into sections and estimating bounding boxes and probabilities for each, to process the entire image, the technique employs a single neural network.

A unique feature of YOLO is its ability to run in real time and its excellent accuracy. To produce predictions, the method merely performs one forward propagation across the neural network, so it “only looks at the image once. “It then outputs recognized objects along with bounding boxes after suppression.

In the software, the picture is divided into areas and crop boxes and the likelihood for each is determined using a single neural network. Usually, the boxes that define the boundaries are weighed using determined likelihood. With YOLO, a single CNN can forecast a few boxed boundaries and complexity probability, as well as shooting crate, all at the same time. This optimizes detection efficiency by training on complete images.

This model has a few distinct advantages over other models when it comes to detecting objects:

- Because YOLO has seen the big picture throughout preparing and assessment, it records qualitative information in relation to groups in addition to their look data.
- Because this method of learning is applicable for a wide range of situations when trained on genuine images, even the most advanced detection algorithms currently available are outperformed by YOLO.

5 Description of the Process and Approach

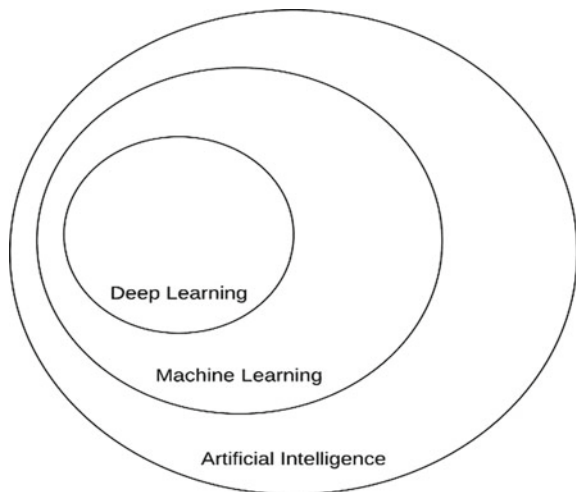
5.1 Basics of Deep Learning

The interrelation between deep learning, machine learning, and artificial Intelligence is depicted visually in the Venn diagram in Fig. 3. AI’s main goal is to create a set of algorithms and approaches that can be utilized to tackle a difficulty that people can handle in a normal and almost automatic manner however that computers struggle with.

A wonderful example of an AI challenge is elucidating and recognizing the parts of a picture which is a job that a person could complete with less effort, however, machines find this a quite tough job. Pattern recognition and data learning are two areas where the machine learning subfield is particularly involved.

Deep learning is part of the ANN algorithm family, and the two concepts can be used interchangeably in most situations. Deep learning has existed for more than sixty years, under numerous titles and personification dependent on research patterns, accessible technology, and repositories. The emphasis on the basics of deep learning, including what makes a neural network “deep” and the idea of “hierarchical learning”

Fig. 3 Deep learning is a subfield of machine learning, which is then a subfield of AI



has contributed to deep learning is one of the most common machine learning and computer vision applications today.

5.2 Image Fundamentals and OpenCV

Pixels are the fundamental elements of any image. Every picture is made up of a series of pixels. The pixel is the smallest granularity that can be used. A pixel is commonly thought of as the “colour” or “intensity” of light visible in a specific area of our picture. When viewed as a grid, each square in an image contains a single pixel.

Each square in an image contains a single pixel if we consider it as a grid. The following depicts an image with a resolution of 1000×750 pixels in Fig. 4, which implies it is 1000 pixels wide and 750 pixels in height. A (multidimensional) matrix can be used to describe an image. In this scenario, our matrix comprises 1000 columns and 750 rows (the height). The total number of pixels in our picture is $1;000 \times 750 = 750;000$.

There are two ways to represent most pixels: Monochrome/single channel or colour channel.

A scalar value between 0 and 255 represents each pixel in a grayscale image with zero representing “black” and 255 representing “white”, with values that are closer to zero being those that are darker and nearer to 255 being brighter. A grayscale gradient of darker pixels on the left and lighter pixels on the right is shown in Fig. 5.

In RGB colour space, a pixel is made up of three values: one for each of the red, green, and blue elements, instead of just a single scalar value, as in a grayscale picture. In the RGB colour model, to determine colour, all that is required is the amount of red, green, and blue contained in a single pixel. For a total of 256 “shades,” each red,

Fig. 4 This image has a width of 1000 pixels and a height of 750 pixels, for a total of 750,000 pixels

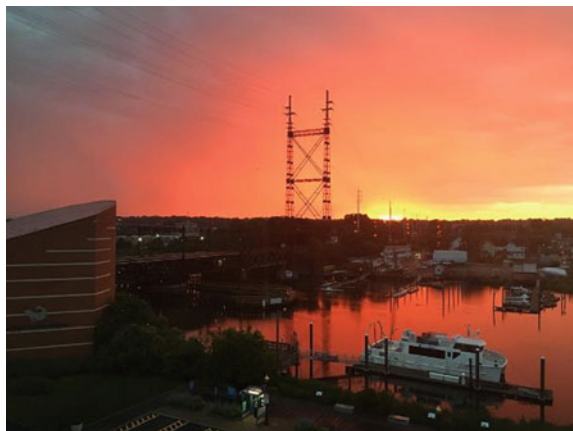
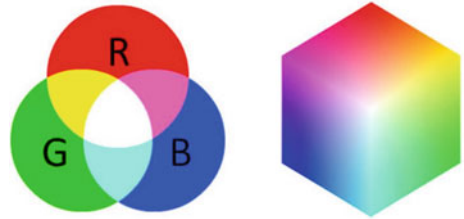


Fig. 6 RGB colour space. RGB cube is on the right



Fig. 5 Pixel value gradient from black (0) to white is shown in this picture (255)



green, and blue channel can have values in the range [0; 255] where 0 denotes no representation, while 255 denotes maximum representation.

The strength of it is usually reflected with 8-bit unsigned integers since the range [0; 255] is required for pixel value. Along with this mean subtraction or scaling on the image is also been done, which entails converting the image to a floating-point data sort.

An RGB tuple is made by combining the three red, green, and blue values (red, green, and blue). This tuple is a representation of a colour in the RGB colour space. An additive colour space, such as the RGB colour space, is an example in which the pixel brightens and approaches white as the amount of each colour is increased.

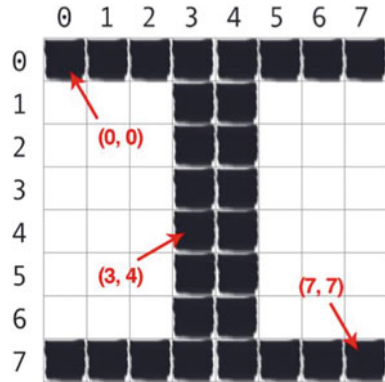
In Fig. 6, the RGB colour space (left) is visible. Adding red and green produces yellow, which is evidenced by the fact that. Combining red and blue results in pink. White is made up of three primary colours red, green, and blue.

Consider the following colour “white.” Each of the red, green, and blue buckets will be filled and as follows: (255, 255, and 255) is a three-digit number. Then, since black is the absence of colour, each of the buckets (0, 0, 0) are emptied. To make a pure red colour, the bucket is filled in red (except the red bucket): (255, 0, 0). A cube is another typical representation of the RGB colour space (right-hand side of the following diagram) because an RGB colour is a three-valued tuple with values ranging from 0 to 255, maybe imagined as the $256 \times 256 \times 256 = 16,777,216$ cube of different colours, based on the amount of each bucket contains three colours: RGB.

The coordinate system for images a picture is depicted as a grid of pixels. To appreciate this issue as an example, consider the grid to be a sheet of graph paper. The genesis point of the image is in the upper-left corner. (0; 0) on this graph document. The *x* and *y* values both increase as we step down and to the right. This “graph paper” representation is depicted in Fig. 7. The letter “I” has been written on a scrap of graph paper. It can be observed that this is a 64-pixel 8 8 grid. It’s worth noting that instead of one, it starts from zero. Because Python is a zero-indexed programming language, it always starts at zero.

RGB pictures are represented as multidimensional NumPy arrays in image processing software such as OpenCV and scikit-image (height, width, and depth).

Fig. 7 A graph paper with the letter “I” scribbled on it. Pixels are accessed using their (x; y)- coordinates. (Python is zero-indexed)



This representation also confuses readers who are new to image processing libraries: Although we usually think of a picture in terms of width first, height second, why does the height arrive before the width? Matrix notation explains the solution. When defining the dimensions of a matrix it is often written as rows \times columns. An image’s height is determined by the number of rows, while its width is determined by the number of columns. The quantity of information of the depth, however, will not change. Seeing a NumPy array’s shape as (height, width, and depth) can seem perplexing at first, but it makes intuitive sense when it comes to the construction and annotation of a matrix.

RGB channels are saved in reverse order by OpenCV. OpenCV keeps pixel data in BGR order despite our tendency of thinking in terms of RGB. This is done by OpenCV for a purpose. Simple history explains why this is so. RGB ordering was employed at the time by camera manufacturers and other software developers, the OpenCV library’s early developers chose the RGB colour format.

Simply said, the order of the RGB values was determined and it is done because of the past, as well as we must at this time respect that decision. It is a minor point to remember while dealing with OpenCV, but it is crucial.

5.3 Basics of Convolutional Neural Networks

A CNN’s layers each apply a different group of filters, usually tens of thousands and then aggregate the outcome before transmitting the next layer’s output. The values of these filters are invariably learned by a CNN at the time of preparation.

A CNN might be able to:

- In the first layer, find edges from raw pixel data in the context of picture categorization, then use these edges to recognize forms in the next layer (or “blobs”).

Fig. 8 Kernel is represented in the diagram above as a small matrix that moves around a larger image from left to right and top to bottom

131	162	232	84	91	207
104	-1	109	+1	237	109
243	-2	202	+2	135	26
185	-15	200	+1	61	225
157	124	25	14	102	108
5	155	16	218	232	249

- These forms can be used to recognize higher-level elements including face a framework, auto elements, and so on in the network’s topmost layers. These higher-level features are used by the final layer in a CNN to make assumptions about the image’s contents [13].
- An (image) convolution is a multiplication of two matrices element by element followed by a number in deep learning.

Take a picture with a large matrix and a kernel with a tiny matrix (in any case in the relation to the source “big matrix” picture shown in the following Fig. 8. The kernel (red region) is slid along the original image from left to right and from top to bottom. The neighbourhood of pixels centred at the kernels’ image centre is examined for each of the original image’s (x; y)-coordinates. The kernel is then convolved with this neighbourhood of pixels to produce a unique output value. The value of output is recorded the same (x; y)-coordinates as the input image in the output image of the kernel’s centre [14].

This is done to ensure an unusual kernel size that the image’s centre has a valid (x; y) coordinate (Fig. 8). A 3 × 3 matrix is found on the left. There are zero-indexed coordinates, and the origin is located in the matrix’s top-left corner. The centre of the matrix is the x = 1; y = 1 point. However, on the right, there is a 2 × 2 matrix. x = 0.5; y = 0.5 is the centre of this matrix. But, as it is known, pixel position (0.5; 0.5) cannot exist without interpolation, as a result, integers must be used for the pixel coordinates. We use odd kernel sizes because of this: make certain that the centre of the kernel always has an (x; y)-coordinate that is correct.

Now that the fundamentals of kernels are understood, let’s look at the convolution operation and see an example of it in action to help us solidify our understanding. A convolution in image processing has three parts:

1. A picture to use as input.
2. A kernel matrix to be applied to the input image.

3. An output image for the image convolved with the kernel's output. Convolution (i.e., cross-correlation) is very easy.

A CNN's layers each apply a different group of filters, usually tens of thousands and then aggregate the outcome before transmitting the next layer's output. The values of these filters are invariably learned by a CNN at the time of preparation. CNN might be able to find edges in the first layer from raw pixel data in the context of picture categorization, then use these edges to recognize forms in the next layer (or "blobs").

These forms can be used to recognize higher-level elements including faces, auto elements, and so on in the network's topmost layers. These higher-level features are used by the final layer in a CNN to make assumptions about the image's contents. An (image) convolution is a multiplication of two matrices element by element followed by a number in deep learning. Take two matrices and multiply them together (which both have the same dimensions). Multiply each factor individually. (Note that this is not a dot product, but rather a simple multiplication.)

Layer Types in CNN

Convolutional neural networks are built using a variety of layers, but the ones you are most likely to see are:

- Activation (ACT)
- Pooling (POOL)
- Fully connected
- CONV (convolutional) is a term used to describe a type of convolute on (ACT or RELU, where the actual activation function is the same) (FC)
- Batch normalization (BN) is the process of converting one batch of data into another batch of (DO).

A CNN is constructed by stacking these layers in a specific order. To illustrate a CNN, we frequently utilize simple text diagrams:

"CONV => RELU => FC => SOFTMAX INPUT => CONV => RELU => FC => SOFTMAX INPUT => CONV => RELUCNN: INPUT => CONV => RELU => FC => SOFTMAX" [15].

5.4 The Four-Step Process of Creating a Deep Learning Model

Step 1. Gathering the data

The first step in creating a deep learning network is to collect our first dataset. The photographs will be required, as well as the labels that go with them. Only a few categories should be utilized for these labels. Furthermore, the number of photos in each group should be comparable (i.e., the same number of examples per category).

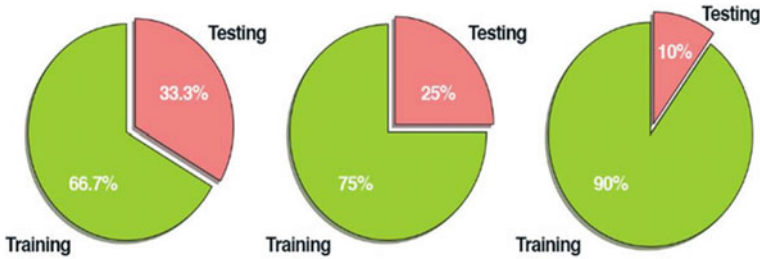


Fig. 9 Examples of common training and testing data splits

The algorithm used would be biased towards overfitting these disproportionately represented groups if there had been twice as many cat photographs as dog images and multiple times as many panda pictures as cat photos [16]. A class imbalance is a common problem in machine learning that impacts a wide range of applications. There are techniques for dealing with this, but the best way to minimize learning problems caused by class imbalance is to abolish them altogether.

Step 2. Dividing the dataset

Our basic dataset will be divided into two parts: (1) a set of drills and (2) a set of scenarios to test.

The classifier here learns how each category appears by making predictions on the input data and then correcting itself when the predictions are erroneous, thanks to a training set. The performance of the classifier on a test set can be evaluated after it has been conditioned. The training and research sets must be different and must not overlap [17].

The training and testing sets must be distinct and not overlap. The classifier will have an unfair advantage if the testing data is included in the training data because it has previously viewed and “learned” from the testing examples. So, they are kept separately as demonstrated in Fig. 9.

Step 3. Train the network

The photo library for training can now be utilized to train the network. To be helpful, the network must start recognizing each of the categories in our named data. When an error is made by the model, it learns from it and improves. The gradient descent method must be used in general [18].

Step 4. Evaluate

Lastly, the highly skilled personnel must be evaluated. Each image in the testing collection is shown to the network, and it is asked to guess the label. The model’s estimations for each image in the testing set are then tallied. Finally, the model predictions are linked to the ground-truth labels from our study dataset. The number of correct predictions generated by our classifier is then calculated, as well as aggregate

Table 1 Video capture test case

Test case	1
Name of the test	Video capture
Input	Web camera, resolution (width, Height)
Expected output	Display video specified by user resolution from web-cam
Actual output	User specified video is displayed
Result	Successful

Table 2 Load trained model test case

Test case	2
Name of the test	Load trained model
Input	YOLO trained weight, cfg, and coco names
Expected output	Model loading without any error
Actual output	Model loading without any error successful
Result	Successful

Table 3 Classification of person test case

Test case	3
Name of the test	Classification of person
Input	Camera video input forming a frame
Expected output	To classify person or individual in frame
Actual output	Individual classified in frame
Result	Successful

reports such as accuracy, recall, and f-measure, which are used to assess our network's overall effectiveness.

6 Test Cases

System developed is tested for its working and its performance. A few test cases prepared for this are in Tables 1, 2, 3 and 4.

7 Results

According to the findings, the distance tracking system had an accuracy range of 56.5–68% when tested on outdoor and demanding input films, while indoor testing in

Table 4 Distance calculation test case

Test case	4
Name of the test	Distance calculation
Input	Camera video input forming a frame
Expected output	Calculating the distance between two individuals in the frame
Actual output	Calculating the distance between two individuals in the frame and alerting
Result	Successful

a controlled setting yielded 100% accuracy. Both are shown in Fig. 10. It was discovered that the “safety violation alert feature based on segmented ROI had superior accuracy,” ranging from 95.8 to 100% for all input videos analysed.

This implemented system development was performed using Python 3, OpenCV for image processing techniques, and the Caffe object identification model framework. Some research has been conducted to determine the efficacy of the system that has been built, and findings have been acquired. “The MobileNet SSD Caffe model” has been employed as the essential algorithm for detecting people. The main surveillance video is from the full scene set in the lounge room, where the camera is mounted high to acquire overhead view, is taken for programme tuning.

8 Impact of the Project and Future Scope

Due to the recent rapid spread of coronavirus disease (COVID-19), social separation has become one of the most important prophylactic methods to avoid physical contact. The project’s main goal is to develop a surveillance system that employs OpenCV, computer vision, and deep learning algorithms to track people and minimize overcrowding while keeping a safe space between them. As the existing system gives performance only up to a certain extent, i.e., the individuals not maintaining social distancing can be identified overall, it can be further enhanced by incorporating additional features and thereby extending the functionality provided. An application that detects the location coordinates of an individual in real time can be introduced. The application has to be installed by a particular individual on his mobile.

This application can then use the GPS feature through which it extracts the location of that particular individual in a required frame and on violation of the set minimum distance constraints; can be personally notified with the help of the app.

9 Conclusion

The necessity for self-responsibility becomes evident as we consider the world in the aftermath of the COVID-19 pandemic. The main focus of the scenario would

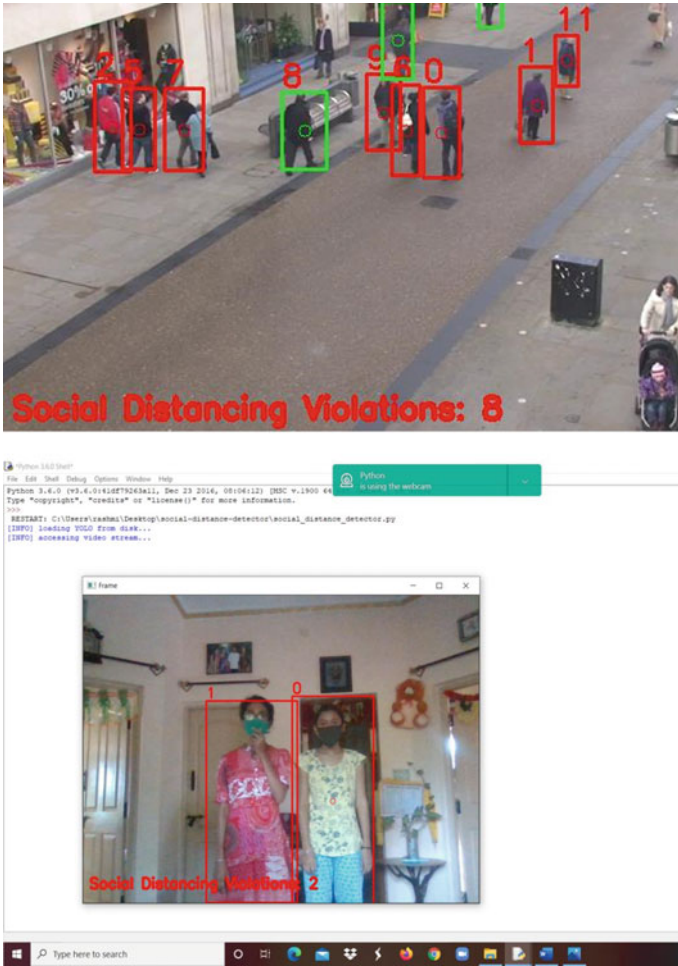


Fig. 10 Result footage still from both indoor and outdoor setting

be on accepting and complying with the WHO’s protections and laws, with the person having ultimate responsibility for themselves rather than the government. Because COVID-19 is disseminated by close contact with sick people, social distance is unquestionably the most important factor. An efficient solution is required to supervise huge crowds, and this is what our system aims to provide. Authorities can maintain track of human activities and control big crowds by using cameras to bring them back together and stop breaking the law. Using cameras to bring people back together and stop breaching the law, authorities can keep track of human actions and control large gatherings. They will be marked with a green boundary box if they are staying a safe distance, and with a red boundary box if they are not.

References

1. Punn NS, Agarwal S (2019) Crowd analysis for congestion control early warning system on foot over bridge. In: 12th International conference on contemporary computing (IC3), IEEE
2. Ren S et al (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv Neural Inf Process Syst* 28:91–99
3. Agarwal A, Gupta S, Singh DK (2016) Review of optical flow technique for moving object detection. In: 2nd International conference on contemporary computing and informatics (IC3I), IEEE
4. Punn NS et al (2020) Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and deepsort techniques. arXiv preprint [arXiv:2005.01385](https://arxiv.org/abs/2005.01385)
5. Charan SS, Saini G (2018) Pedestrian detection system with a clear approach on raspberry Pi 3. In: International conference on inventive research in computing applications (ICIRCA), IEEE
6. Ahamad AH, Zaini N, Latip MF (2020) Person detection for social distancing and safety violation alert based on segmented ROI. In: 10th IEEE international conference on control system, computing and engineering (ICCSCE), IEEE (2020)
7. Tarimo W, Sabra MM, Hendre S (2020) Real-time deep learning-based object detection framework. In: IEEE symposium series on computational intelligence (SSCI), IEEE
8. Li K, Lu C (2020) A review of object detection techniques. In: 5th international conference on electromechanical control technology and transportation (ICECTT), IEEE
9. Xu Y et al (2016) Background modeling methods in video analysis: a review and comparative evaluation. *CAAI Trans Intell Technol* 1(1):43–60
10. Tsutsui H, Miura J, Shirai Y (2001) Optical flow-based person tracking by multiple cameras. In: Conference documentation international conference on multisensor fusion and integration for intelligent systems. MFI 2001, IEEE
11. Dollár P et al (2005) Behavior recognition via sparse spatio-temporal features. In: IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance, IEEE
12. Niyogi SA, Adelson EH (1994) Analyzing gait with spatiotemporal surfaces. In: Proceedings of IEEE workshop on motion of non-rigid and articulated objects, IEEE
13. Piccardi M (2004) Background subtraction techniques: a review. In: IEEE international conference on systems, man and cybernetics, vol 4. IEEE
14. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 25:1097–1105
15. Musaev M, Khujayorov I, Ochilov M (2020) The use of neural networks to improve the recognition accuracy of explosive and unvoiced phonemes in Uzbek language. In: Information communication technologies conference (ICTC), IEEE (2020)
16. Brunetti A et al (2018) Computer vision and deep learning techniques for pedestrian detection and tracking: a survey. *Neurocomputing* 300:17–33
17. Zhao Z-Q et al (2019) Object detection with deep learning: a review. *IEEE Trans Neural Netw Learn Syst* 32:3212–3232
18. Ahmed Z, Iniyavan R (2019) Enhanced vulnerable pedestrian detection using deep learning. In: International conference on communication and signal processing (ICCSP), IEEE (2019)

Autism Spectrum Disorder Prediction Using Machine Learning



A. C. Ramachandra, N. Rajesh, G. Sai Harshitha, and C. R. Prashanth

1 Introduction

Autism spectrum disorder is neurodevelopmental disorder. It could start as early as the age of 12 months and will continue to grow with age [1]. This is known as a spectrum disorder is a condition where one or more symptoms of a disease can be present. It is important to determine the cause of the disorder. To determine the severity of autism disorder, we must do more research. More than 5 symptoms can be attributed to both communication and behavioural issues. Together with this means that symptoms may vary among individuals. This disorder is difficult to predict. We use the traditional method of observation to analyze the statues of patients. This takes a lot of time to decide. The clinicians' decision may or may not be final. It is not true. People are also less aware of the existence of disorders. Very sensitive. This is why we must think about it. Toddlers [2]. Repetitive behaviour symptoms are some of the signs that can be associated with this condition. Movement, laughter unnecessarily, less alert to dangerous situations.

Communication disorder is a condition where the person does not respond to the name or understands. Expressing your feelings will not involve eye contact (Fig. 1).

The brain is the command centre for human body, and it sends every message. Message to the body all through the life. A healthy brain is vital. As autism is a neurodevelopmental disorder that belongs to the brain. Because of the messages, the neurons carry information from the brain to the brain [3]. This disorder causes less. This can affect the transmission of messages by causing connections between

A. C. Ramachandra (✉) · N. Rajesh · G. S. Harshitha
Nitte Meenakshi Institute of Technology, Bengaluru 560064, India
e-mail: ramachandra.ac@nmit.ac.in

C. R. Prashanth
Dr. Ambedkar Institute of Technology, Bangalore 560056, India



Fig. 1 Characteristics autism spectrum disorder

neurons. These connections can be made between neurons. Some brain parts are especially affected: cerebral cortex, basil ganglia, and amygdala (Fig. 2).

People are now taking better care of themselves by visiting the general health check- ups should be done every six months or once a year. Take care of your mental health, as people are so busy these days. Life is hard. Humans can be subject to stress and pressure, and children might grow up alone. Their growth [4]. We propose a model that will allow them to take advantage of this. Assessment of the people is based upon their inputs. Decide if they are suffering with disorder or not. You can take this assessment remotely at any time. Helps to diagnose the disorder and offer

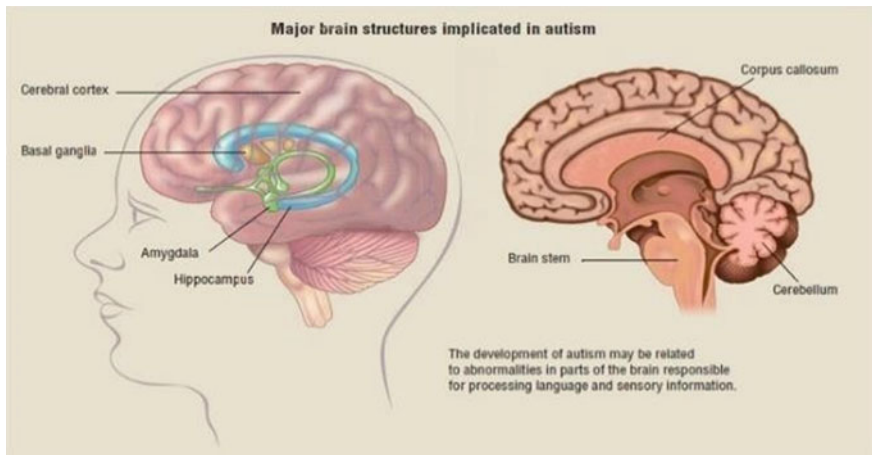


Fig. 2 Brain structures implicated in autism

therapy, if necessary, disorder. We can thus improve our quality of life. This model was designed by us. Machine learning is used where the machine has been trained, and the machine will automatically learn. Predict for the new input. This is because machine learning is developing. It allows us to know more about the world and makes it easier for us to make informed decisions. Technology is also used by people.

It is easy to find a doctor these days, and all diagnoses are available. Remote assessment so that we can offer the prediction to the user the disorder.

2 Related Works

Huang and his colleagues discussed the identification of autism spectrum disorder [5] in their discussion. As autism spectrum disorder (ASD) is on the rise, it is crucial to recognize ASD. Patients for early intervention and effective treatment, particularly in the area of childhood. Yaneva and colleagues worked to detect high-functioning autism among adults [6] whether there are visual processing differences in adults with and without high-powered vision. Eye tracking can be used for diagnosing autism by capturing functioning autism. Yuan et al. analyzed the automatic identification of high-risk autistic spectrum disorder [1, 7, 8] that the symptoms of autism spectrum disorders (ASDs) could be improved by early intervention, which is a great way to help the situation. Identification of ASD: Mostafa and colleagues presented the eigenvalues for brain networks diagnose [9] autism spectrum disorder neurodysfunction, which causes Patients with repetitive behaviours and social instability are two examples. Reem Haweel and colleagues worked on Autism severity using a response-to-speech study. A neurodevelopmental disorder associated with impairments of social and lingual skills.

In the ASD population, failure in language development can be variable and follow a wide range of factors. Spectrum Koirala et al. [10] conducted virtual reality-based touch and visual sensory experiments processing assessment for adolescents with autism spectrum disorders. Sensory individuals with autism spectrum disorder (ASD) experience abnormalities. A developmental disorder affects children around the world. Del Coco et al. [11] the mechanisms of stimulation of social interaction in autism spectrum disorder have been studied. It has been proven that information and communication technologies can have a tremendous impact on our lives. Impact on children's social, communicative, and language development Munoz and co. presented software that supports the [12] improvement of the theory of mind in autism spectrum disorder children. Sang wan lee et al., Deeply explored the strategic and structural bases of autism spectrum disorders. Learning: clinical research is conducted using deep learning models [3] to aid in diagnosis. Disease: it is difficult to diagnose autism spectrum disorders (ASDs) because of its complexity. Complex psychiatric symptoms and a general insufficient amount of medication can lead to complex psychiatric problems. Neurobiological evidence.

Akter et al. worked on machine learning-based models for early-stage detection autism spectrum disorders. Muhammad awais bin altaf and others analyzed on chip.

For chronic neurological disorders, processor [5] CNDs are lifelong illnesses that cannot be eliminated, but they can be treated. Preventive measures taken early can reduce the severity of these problems. Wang et al. worked on identification of autism-based upon SVM-RFE [7] in order to improve the accuracy of classification based on the complete autism brain imaging data exchange dataset, patients with autism. They first applied the resting state functional magnetic resonance imaging data and calculate the functional connectivity. They also adopted the support vector machine recursive features.

They also trained an auto-encoder stacked sparse with two hidden layers for extracting. The high-level latent features and complex features of the 1000 features.

Mingxia Liu et al. worked on the identification of autism spectrum disorder using FMRI. They propose a multidisciplinary approach. Site [13] adaption framework via low-rank representation decomposition (MALRR) for functional MRI (FMRI) is used to identify ASDs. It is important to identify a common low-rank representation of data from multiple sites. This is a goal to reduce. There are differences in data distributions.

3 Proposed System Architecture

We wanted to create a model that could predict new inputs and disorder. This is where the machine learning model must be trained. This requires the dataset. Next, we must label the data in order for machine learning to take place. Next, we must apply algorithm to machine to make right decision and predict output. All details are provided in Fig. 3.

3.1 Input Data

A comma separated values file is a file that contains delimited text and uses a comma for separation values. Each data record is a line in the file. Each record is composed of one or more data records. More fields separated by commas [2]. Use of the comma to separate fields is the source name of this file format. CSV files typically store tabular data. Plain text will display the lines with the same number fields as in plain text. We are here we need to have the dataset of individuals and their related symptoms, so we are gathering by conducting survey. We are looking at 1100 patients and 21



Fig. 3 Proposed block diagram

attributes. Any family members who have jaundice are advised to check the symptoms and where they live. Disorder before it was related to genetic factor.

3.2 Labelling the Data

Data labelling is an essential part of data pre-processing in ML, especially for the case of data labels. Supervised learning is where input and output data are labelled in order to create a learning base for future data processing. Data labelling can also be used. Constructing ML algorithms to create an autonomous model [4]. Labelling data is intended to aid the machine to learn because not all data is understandable by humans machine. So, labelling the item will allow machine to make the correct decision about new set data. To move data between programmes, you do not normally use, and you can use a CSV file. Data exchange possible.

3.3 Classifier

Classification is the process by which data points are predicted to belong to a particular class. There are two types of classes. Sometimes referred to as targets/ labels/ categories. Predictive modelling for classification is the task of approximating the mapping function (f), from input variables (X), to discrete output variables (y). The category of supervised learning includes classification, where the targets are also included in the input data. There are [14] different applications in classification in many domains, such as credit approval, medical diagnosis, and target marketing.

Although there are many classification algorithms now, it is not sufficient. It is possible to determine which one is superior. It all depends on the application. Nature of the dataset: [1] if the classes can be separated linearly, for example, linear classifiers such as logistic regression and fisher's linear discriminant may outperform high-end models and vice versa.

3.4 Predicting for New Input

Prediction is the output of an algorithm that has been trained using historical data. Dataset used to forecast the likelihood of a particular event result. Machine learning algorithms look for rules that will allow them to. Determine the general characteristics of elements in a group to achieve the goal of Apply the learning to other elements.

3.5 Work Flow of the Designed Model

We have labelled the data so that it is easy for machines to understand. We now need to apply classifier, and there are many options available. Requirement: there are many symptoms in our project that are called attributes. We need to verify that the attributes are present. This can be done by Because it works on the probability function, the Naive Bayes classifier is used. Number of instances: this is illustrated by 8 instances and 4 attributes' 8 patients (Table 1; Fig. 4).

Calculating probability function for attributes A1 to A4.

A1	Yes	No
----	-----	----

(continued)

Table 1 Example attributes

Patients	A1	A2	A3	A4	Results
P1	0	1	0	1	2
P2	0	1	1	1	3
P3	1	0	0	0	1
P4	0	1	1	0	2
P5	1	1	1	1	4
P6	1	0	1	1	3
P7	1	1	1	1	4
P8	0	0	0	0	0

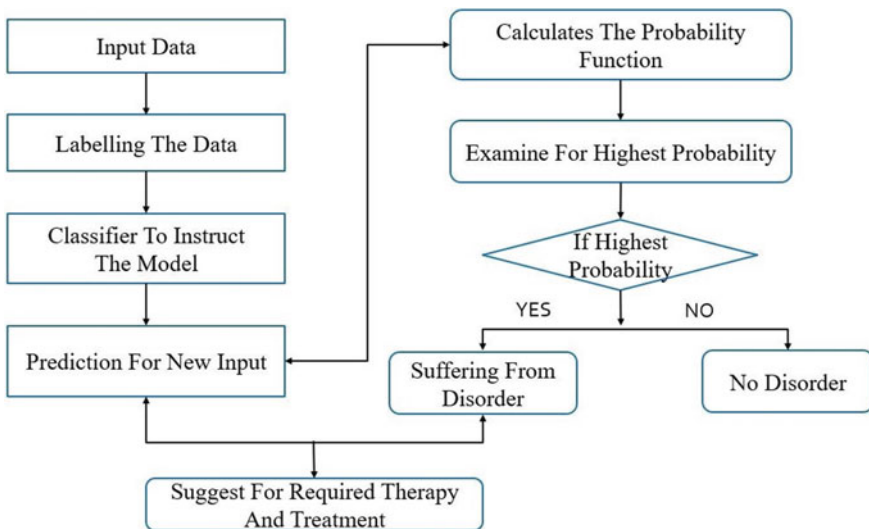


Fig. 4 Work flow of proposed model

(continued)

0	1/3	2/5
1	2/3	2/5
A2	Yes	No
0	1/3	2/5
1	2/3	3/5
A3	Yes	No
0	0	3/5
1	1	2/5
A4	Yes	No
0	0	3/5
1	2/3	2/5

The above all calculation is based on Naïve Bayes classifier.

- $P(\text{NEW}|\text{YES}) = P(\text{YES}) * P(A1 = 0|\text{YES}) * P(A2 = 1|\text{YES}) * P(A3 = 0|\text{YES}) * P(A4 = 1|\text{YES})$.
- $P(\text{NEW}|\text{NO}) = P(\text{NO}) * P(A1 = 0|\text{NO}) * P(A2 = 1|\text{NO}) * P(A3 = 0|\text{NO}) * P(A4 = 1|\text{NO})$.

Before training, the image resolution is uniformly scaled to 224×224 to ensure the rationality of network input.

Naive Bayes classifier can be one of the most simple and effective classifications. Algorithm aids in the construction of fast machine learning models that are able to make quick predictions. It is a probabilistic classification which predicts on the base of the object’s probability. The Naive Bayes algorithm can be used for supervised learning algorithm, which is based upon Bayes theorem. It is used to solve classification problems. It is used mainly in text classification, which includes a high-dimensional training dataset. It is a probabilistic classification machine, meaning it can predict on the basis. It is used to calculate the likelihood of an object. Because it assumes that there is no object, it is called Naive.

The occurrence or absence of any feature is not dependent on other features. As if the fruit was identified based on its colour, shape, taste, and texture, then red. Spherical and sweet fruits are recognized as apples. Each feature is therefore unique. It helps to recognize that an apple is an apple, without relying on the other. It is called Bayes, because it is based on the principal Bayes’ theory. Bayes’ theorem can also be found. Bayes rule, also known as Bayes law, is used to calculate the likelihood of a given outcome. Hypothesis with prior knowledge: it is dependent on the conditional probabilities. It depends on the conditional probability.

The above example demonstrates how working is done.

By applying the Naïve Bayes probability function for the new instance with 4 attributes and applying in the above formula, we can come to know how the prediction is done.

<i>New instance 1</i>			
A1	A2	A3	A4
0	1	0	1
<i>New instance 2</i>			
A1	A2	A3	A4
1	1	1	1

- $P(\text{NEW|YES}) = 0$
 $P(\text{NEW|NO}) = 0.036.$
- $P(\text{NEW|NO}) > P(\text{NEW|YES})$

For new instance 1, we can see occurrence of no is greater than occurrence yes, hence, we can say the user has no disorder.

Similarly for new instance 2, we got occurrence of yes is greater, so the user is found suffering from disorder. This is how the prediction works.

- $P(\text{NEW|YES}) = 0.111$
 $P(\text{NEW|NO}) = 0.025$
- $P(\text{NEW|YES}) > P(\text{NEW|NO}).$

Attributes we are considering from each individual are shown in Table 3.

We are using Tables 2 and 3 to help us formulate the assessment questions for you. We will explain in detail how it works behind the scenes during the implementation. We are also considering how we will interface with the server and user. We are currently considering. These attributes are important as follows,

1. The age at which the disorder may begin is known as age.
2. Gender refers to the analysis of male or female affected more, and why.

Table 2 Attributes with its description

Attributes	Type	Description
Age	Number	Age in years
Gender	String	Male or female
Ethnicity	String	list of ethnicities
Born with Jaundice	Boolean (Yes or No)	Whether born with jaundice
Genetic factor	Boolean (Yes or No)	Whether any family members have disorder
Country of residence	String	List of countries
Why took screening	String	Are you finding any change in your daily routine
Who completing screening	String	Parent, self, clinicians

Table 3 Screening questions

Screening questions	Type of symptoms	Boolean/integer
A1	Have limited speech	Binary (0,1)
A2	Give random answers	Binary (0,1)
A3	Will not respond to their name	Binary (0,1)
A4	Avoid eye contact	Binary (0,1)
A5	Struggle to understand and express feelings	Binary (0,1)
A6	Engage in repetitive behaviours	Binary (0,1)
A7	Sensitive to sensory things	Binary (0,1)
A8	Struggle to socialize with others	Binary (0,1)
A9	Have little awareness of dangerous situations	Binary (0,1)
A10	Incorporate attachment with objects	Binary (0,1)

3. To find out which part of the world is most popular; you need to know your ethnicity and country suffered.
4. To learn more about autism spectrum effects of jaundice in children born with it disorder.
5. To determine if a family member has this disorder, genetic factors are considered. It is also known as hereditary.
6. If there is any significant change in the daily routine, a screening test will be performed.

4 Implementation

In the implementation stage, how we programme in the backend. Here, we are using PyCharm IDE tool. To work on this firstly, we need to create a new environment for new project. Once environment is created, install all the packages and libraries required for our project. If everything is set, we can see the command in the terminal that environment is created.

```
(venv) C:\Users\hsai2\Desktop\autism with flask
```

Once the environment is set, import all the libraries and dataset.

```
from random import random
import pandas as pd
data = pd.read_csv('data.csv')
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
```

Once the dataset is imported, we need to label the data, and this done use `fit_transform()` function for this. Labelling will help target labels, and it is also used to transform non numerical data. `fit_transform()` is used on the training data so that we can scale the training data and also learn the scaling parameters of that data. Here, the model built by us will learn then used to scale our test data.

```
age = le.fit_transform(data.age)
gender = le.fit_transform(data.gender)
ethnicity = le.fit_transform(data.ethnicity)
jaundice = le.fit_transform(data.jaundice)
austim = le.fit_transform(data.austim)
country_of_res = le.fit_transform(data.country_of_res)
used_app_before = le.fit_transform(data.used_app_before)
age_desc = le.fit_transform(data.age_desc)
relation = le.fit_transform(data.relation)
```

The above figure shows the labelled data.

As we are using the Naïve Bayes classifier, we are importing all the required libraries for it, and then, we are selecting BernoulliNB in Naïve Bayes because it is multinominal and works with occurrence counts. This is designed by binary/Boolean features and also it helps to use multiple classes.

```
#-----naives bayes-----
import sklearn
from sklearn.naive_bayes import BernoulliNB
from sklearn import metrics
from sklearn.metrics import accuracy_score

BernNB = BernoulliNB(binarize=.1)
BernNB.fit(X_train,y_train)
print(BernNB)
y_expect = y_test
y_pred = BernNB.predict(X_test)
print(accuracy_score(y_expect,y_pred)*100)
```

Once we have used the machine learning algorithm, we need an interface between the user interface and server so we need a frame work which acts as a interface. For this, we are using Flask framework.

Figure 5 explains the interface between the user end and server. The ML model in the server space receives the input from the flask which is entered by the user in the frontend. The ML model works with the probability function and gives output predicted back to the use space.

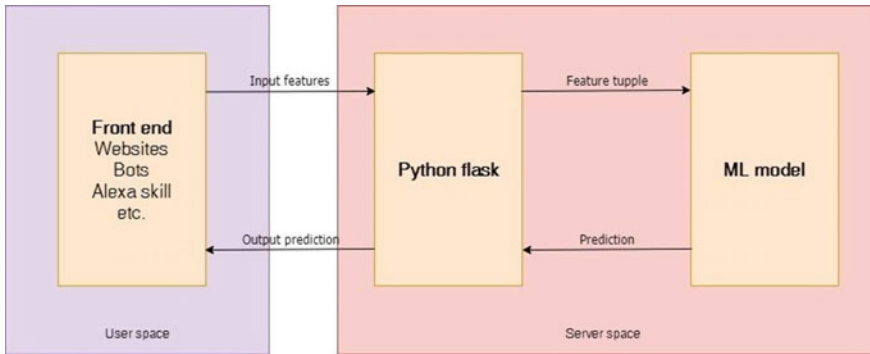


Fig. 5 User interfacing

```
from flask import Flask, render_template, url_for, request, flash, redirect, session
app = Flask(__name__)
```

Flask is an API of Python that allows us to build up Web applications. A Web application framework or Web framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as protocols. Flask is based on Web server gateway interface (WSGI). We have used this because implementation has base code and easier to learn. @app. Route will provide URL which is required for user interface. Along with this url_for, we help us to redirect to the templates where we are using the home page, registration page, login page, and about page. We can direct to any page by using url_for followed by the template name.

```
@app.route('/predict', methods=['POST', 'GET'])
def predict():
    global BernNB
    A1_Score = request.form['A1_Score']
    A2_Score = request.form['A2_Score']
    A3_Score = request.form['A3_Score']
    A4_Score = request.form['A4_Score']
    A5_Score = request.form['A5_Score']
    A6_Score = request.form['A6_Score']
    A7_Score = request.form['A7_Score']
    A8_Score = request.form['A8_Score']
    A9_Score = request.form['A9_Score']
    A10_Score = request.form['A10_Score']
```

These attributes we are using which are named as A1_Score to A10_Score. Next, we are giving the details required from the user in the user registration page. Then, login page allows the user to login for assessment using the gmail id and password. If new user is using, he/she need to register first and then login next. Also, there is about

page which helps the users to know about the disorder and steps to be taken to take the assessment. Based on the prediction output, we are suggesting required therapy and diagnosis required. The post and get will transfer data. url_for will redirect to the templates designed for the Web page. Some of them are home page, user page, login page, about page, and suggestion page.

5 Results

In the results, we are going to show the graph where we can analyze how many patients have disorder with highest number of attributes. Along with that we are showing the how the home pages, about page, and assessment pages look (Figs. 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17).

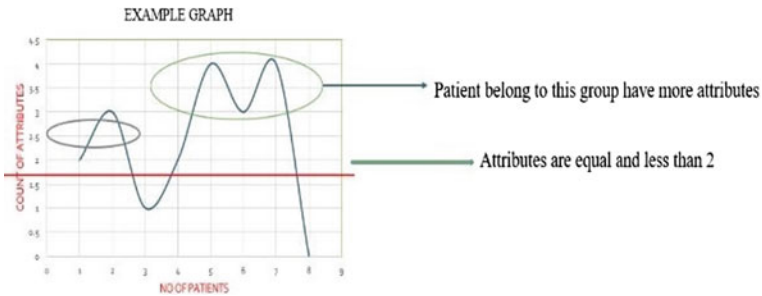


Fig. 6 Graph of example 1

Fig. 7 Bar graph with attributes count

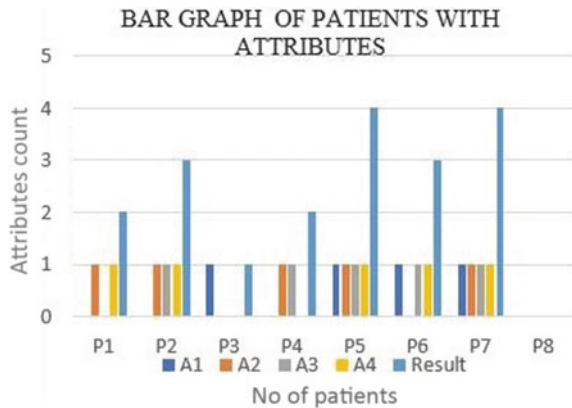


Fig. 12 Registration page

A registration form on a dark background. It contains the following fields and labels: 'Username' with a placeholder 'Enter Name', 'Email Address' with a placeholder 'Enter Email', 'Password' with a placeholder 'Enter Password', 'Gender' with a placeholder 'Enter Gender', and 'Your AGE' with a placeholder 'Enter your age'. A 'Submit' button is located at the bottom right of the form area.

A form titled 'Autism Disease Diagnosis'. It lists 13 screening questions on the left and corresponding input fields on the right. The questions and their respective input fields are: 'Enter your age' (text input), 'Enter your Gender' (radio buttons for Male and Female), '#1_Score: Limited Speech' (text input), 'A2_Score: Give Random Answers' (text input), 'A3_Score: Will not respond to their Name Irregularly' (text input), 'A4_Score: Avoid Eye Contact' (text input), 'A5_Score: Struggle to understand and express feelings' (text input), 'A6_Score: Engage in repetitive behavior' (text input), 'A7_Score: Under Sensitivity to heat, smell and taste' (text input), 'A8_Score: Struggle to cook/eat' (text input), 'A9_Score: Will be nervous in dangerous situations' (text input), and 'A10_Score: Inappropriate attachment with objects' (text input). A 'Test submit app in quiz' button is located at the top right of the form.

Fig. 13 1st set of screening questions

A form titled '2nd set of screening questions'. It contains the following fields: 'ethnicity' (text input with 'Hispanic' entered), 'Born with problem' (radio buttons for Yes and No), 'Any family members have the symptoms previously' (radio buttons for Yes and No), 'Country of resident' (text input with 'India' entered), 'Used App Before' (radio buttons for Yes and No), 'Have you consulted any doctor before' (radio buttons for Yes and No), 'age, disc' (text input with '0-10 years please take test under vision of parents' entered), and 'relation' (text input with 'Self' entered). A 'Submit' button is located at the bottom right of the form.

Fig. 14 2nd set of screening questions

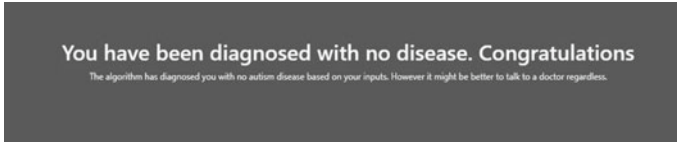


Fig. 15 Output as no disorder

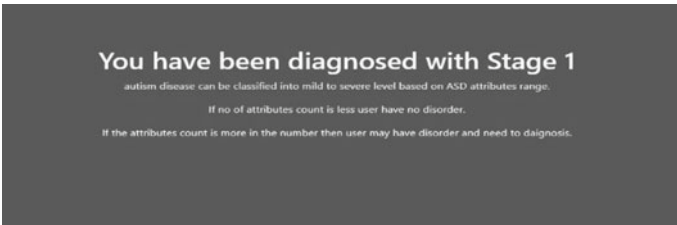


Fig. 16 Output as diagnosed with disorder

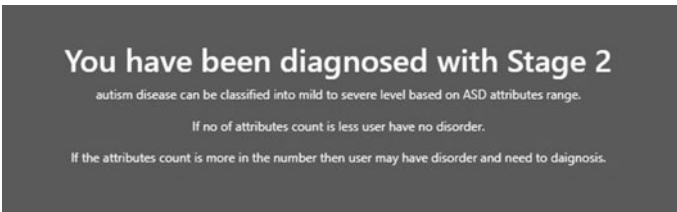


Fig. 17 Output as diagnosed with disorder

6 Conclusion

We have analyzed the data for adults, adolescents, and toddlers. We are now ready to share our findings with you. We conclude that this classifier works best when we apply the Naive Bayes probability function. Predict the presence of the symptoms or attributes by examining the frequency. The proposed model can accurately predict and treat the disorder. Early diagnosis is key for patients. Future studies will be based on ethnicity, gender, and country. We can identify the most affected areas and alert them to help save their lives.

References

1. Tang C et al (2020) Automatic identification of high-risk autism spectrum disorder: a feasibility study using video and audio data under the still-face paradigm. *IEEE Trans Neural Syst Rehabil Eng* 28(11):2401–2410

2. Haweel R et al (2021) A novel grading system for autism severity level using task-based functional MRI: a response to speech study. *IEEE Access*
3. Ke F et al (2020) Exploring the structural and strategic bases of autism spectrum disorders with deep learning. *IEEE Access* 8:153341–153352
4. Akter T et al (2019) Machine learning-based models for early stage detection of autism spectrum disorders. *IEEE Access* 7:166509–166527
5. Aslam AR, Altaf MAB (2020) An on-chip processor for chronic neurological disorders assistance using negative affectivity classification. *IEEE Trans Biomed Circ Syst* 14(4):838–851
6. Yaneva V et al (2020) Detecting high-functioning autism in adults using eye tracking and machine learning. *IEEE Trans Neural Syst Rehabil Eng* 28(6):1254–1261
7. Wang C et al (2019) Identification of autism based on SVM-RFE and stacked sparse auto-encoder. *IEEE Access* 7:118030–118036
8. Chén OY et al (2020) Building a machine-learning framework to remotely assess Parkinson's disease using smartphones. *IEEE Trans Biomed Eng* 67(12):3491–3500
9. Mostafa S, Tang L, F-X Wu (2019) Diagnosis of autism spectrum disorder based on eigenvalues of brain networks. *IEEE Access* 7:128474–128486
10. Koirala A et al (2021) A preliminary exploration of virtual reality-based visual and touch sensory processing assessment for adolescents with autism spectrum disorder. *IEEE Trans Neural Syst Rehabil Eng* 29:619–628
11. Del Coco M et al (2017) Study of mechanisms of social interaction stimulation in autism spectrum disorder by assisted humanoid robot. *IEEE Trans Cogn Dev Syst* 10(4):993–1004
12. Munoz R et al (2018) Developing a software that supports the improvement of the theory of mind in children with autism spectrum disorder. *IEEE Access* 7:7948–7956
13. Wang M et al (2019) Identifying autism spectrum disorder with multi-site fMRI via low-rank domain adaptation. *IEEE Trans Med Imag* 39(3):644–655
14. Tamilarasi FC, Shanmugarn J (2020) Evaluation of autism classification using machine learning techniques. In: 2020 3rd international conference on smart systems and inventive technology (ICSSIT). IEEE, 2020
15. Huang Z-A et al (2020) Identifying autism spectrum disorder from resting-state fMRI using deep belief network. *IEEE Trans Neural Netw Learn Syst* 32(7):2847–2861
16. Zhao Z et al (2019) Applying machine learning to identify autism with restricted kinematic features. *IEEE Access* 7:157614–157622
17. Banire B et al (2020) The effects of visual stimuli on attention in children with autism spectrum disorder: an eye-tracking study. *IEEE Access* 8:225663–225674
18. Liang S et al (2021) Autism spectrum self-stimulatory behaviors classification using explainable temporal coherency deep features and SVM classifier. *IEEE Access* 9:34264–34275

Early Detection of Infection in Tomato Plant and Recommend the Solution



A. C. Ramachandra, N. Rajesh, N. B. Megha, Apoorva Singh,
and C. R. Prashanth

1 Introduction

Agriculture has been the basis of human existence since its inception. India's main occupation is agriculture. India is second in terms of agricultural production. Wherein variety of crops are grown. Modern organic farming has brought more attention to quality and yield. As the number of crops increases year on year, so do the diseases. Plant diseases can ruin agricultural yields. It is a serious problem for food safety. Climatical conditions are not control of humans, and this is a major setback for farmers and hence a big loss. Due to uncontrolled change in climate, the agriculture sector is attacked by millions of pests. This should be detected in early stages, failing which there are the chances of completed failure in crop yield. The symptoms can be seen in different parts of plants, such as the leaves, stems and lesions, and the fruits. The leaf will show the symptoms by changing color or showing spots. This process will help in plant disease classification and detection, which leads to better quality and higher plant productivity.

Traditional methods for diagnosing disease require extensive knowledge and experience in the field. Manual observation and pathogen detection are the best methods for diagnosing disease. However, this can be costly and time-consuming. Farmers used to monitor their crops at regular intervals. If they could not identify the disease symptoms, they would apply a certain amount of pesticide or fertilizer which may lead in reduction in yield.

The absence of disease can lead to incorrect fertilizer applications, which ultimately harm both the plant as well as the soil. Farmers often resort to pesticides and

A. C. Ramachandra (✉) · N. Rajesh · N. B. Megha · A. Singh
Nitte Meenakshi Institute of Technology, Bengaluru 560064, India
e-mail: ramachandra.ac@nmit.ac.in

C. R. Prashanth
Dr. Ambedkar Institute of Technology, Bangalore 560056, India

expensive methods to avoid these diseases. This approach also increases production costs and causes major monetary losses to farmers. Effective disease management begins with early detection.

To improve accuracy and minimize detection of traditional leaf diseases, as well as to take into account leaf position, we use image processing with the neural network. The proposed approach improves the detection of tomato diseases and can even suggest treatments.

The aim is to develop a sheet recognition algorithm based on specific features from photography. This therefore introduces an approach where the plant is identified based on the properties of its leaves such as area, histogram equalization, and edge detection and classification. The main purpose of this algorithm is to use OpenCV resources.

The tomato plant is considered for experimental study. Compare to all other plants, tomato plant is quite sensitive, and it requires particular weather conditions to grow. As the prize of tomato is fluctuating in our day-to-day life, it is very important to detect the diseases reduce the loss.

Table 1 shows the life span of tomato plant stage by stage. It is very important to observe or monitor the plant during the period of 30 to 40 days. Because during thig stages, there is a high chance of plants getting infected. If we monitor plants correctly during this period, then we can reduce the loos of production due to infection.

To help farmers, a new method to identify tomato diseases is suggested. Our approach is able to detect tomato diseases more accurately. For experimentation, we are using totally 2511 image of 5 disease classes to train the model. Finally, by comparing both the outputs from ResNet-50 and CNN model, the system gives the better accuracy compared to existing methods. The different images with respect to the diseases are as shown in Table 2.

Table 1 List of leaf images for training

Growth stage	Stage duration (days)	Crop age (days)
Planting	1	1
Vegetative	14	15
First Flowering	15	30
First fruit Set	10	40
Fruit growth	20	60

Table 2 Number of leaf images after classification

Diseases	No. of images
Healthy	123
Yellow leaf curl virus	488
Bacterial spot	491
Leaf mold fungus	825
Septoria	584

2 Related Work

The rapid advancement of computers in the last few years has made vision and deep learning possible. This has significantly increased image recognition's flexibility as well as accuracy. Deep learning is able to extract classifications in a better way compared to other technology. Using deep learning, features can be extracted directly without the need to use classifiers. Deep learning is an effective method of classifying in many situations. It works very effectively in at generalization, especially when it comes to the extraction of complex and special features.

Aravinth et al. [1] introduced a method to identify brinjal leaf diseases like Bacterial Wilt and Cercospora Leaf Spot. Collar Rot and the method to detect diseases with care. Artificial neural network was used for classification. K -means clustering algorithm was used for segmentation, and texture features identification is used for feature identification. Kamlapurkar proposed a system that can give more precise results in the classification and identification of disease from an image of a leaf. They used different methods [2] such as pre-processing, training, and identification. They used feature extraction to classify images and diagnose. Zhou et al. had restructured the residual dense network to identify tomato leaf diseases. The hybrid deep learning model combines the best of dense and deep residual networks. This can improve the accuracy of [3] calculations as well as increase the flow of information. The model achieves a top-one average identification accuracy, according to experimental results. Ding and colleagues had used tomato leaves for their experiments. They used [4] deep learning to extract disease features from the leaf surface. ResNet-50 is used as the base network model in this experiment. Subhajit Maity and colleagues proposed a simple method to detect leaf diseases [5] by using images of leaves. This was done with image processing and segmentation. For identification, they used Otsu's method and k -means clustering. Ding et al. used a pixel wise [4] instance segmentation technique, mask region-based convolutional neural network, of an improved version, in order to detect cucumber fruits. This [6, 7] research identifies the disease in four stages: image acquisition, image segmentation, and feature extraction. The extracted features include contrast, energy, homogeneity, and mean, standard deviation, variance, and energy. Saxen and colleagues proposed an easy and quick face alignment method for pre-processing. They also address the [8, 9] problem of estimating facial attributes using RGB images for mobile devices. MobileNetV2 and NASNet mobile are two lightweight CNN architectures.

3 Proposed System Flow

The flow of the system is shown in Fig. 1. Firstly, we are taking raw images of five different diseases of different sizes. To make size of the images, same pre-processing is done. The block diagram is the proposed model which is as shown in Fig. 1.

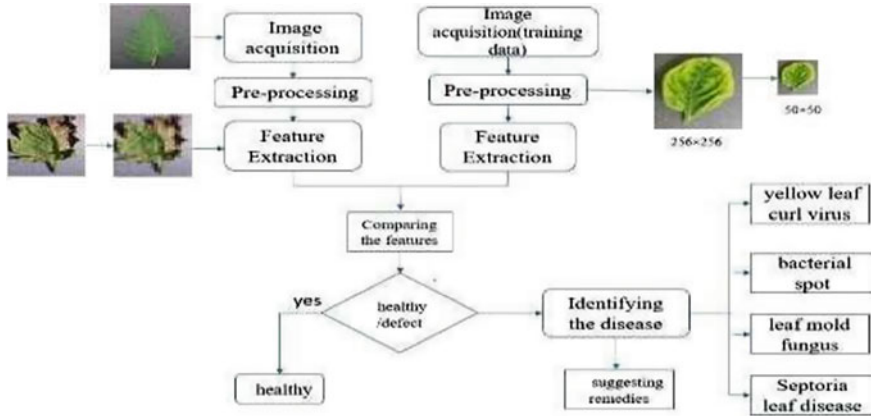


Fig. 1 System flow diagram

3.1 Data Acquisition

Tomatoes are one of the most widely grown agricultural crops. It is grown extensively in both north and south India. The experiment produced 2511 images showing the five most prevalent tomato leaf diseases: Septoria, bacterial spot, and leaf mold fungus. The data was obtained from Kaggle. Some examples can be seen in Fig. 2.

3.2 Pre-processing

The database is pre-processed, such as image reshaping and resizing. The test image also undergoes similar processing. Pre-processing refers to the improvement of image data in order to suppress unwanted distortions or enhance some important image features for further processing. The resizing of image is shown in Fig. 3.

3.3 Feature Extraction

Here, we use convolutional neural network which acts as a combination of two components: feature extraction part and the classification part. Where feature extraction part uses the convolutional layers and extracts the image feature, then classification is done using softmax classifier. Initially, the image is converted into pixel format, and the values are based on RGB. The average of these three values is taken and used as features. The kernel now learns about the features and identifies the disease.

convolution2D and maxpooling2D. These layers can be used to extract the feature or classify the disease. The algorithms is able to detect the disease in a plant species once it has been trained. This is done by comparing features from the test image and tarin. The trained model and the test image can detect the disease in the leaf.

The experiment involved a classification and sorting of the training photos. These were then placed in the folder that corresponded to the disease category name. Comparing the ResNet-50 network with its original ResNet-50 network, we found different activation functions and convolution kernel size. For the classification of diseases in a image, we use a convolution neural network. Here, we implement CNN using pre-trained ResNet-50 [10] neural network architecture using TensorFlow and OpenCV in Python platform.

4 Result Analysis

The experiments is done using the available dataset which is having images of all five different diseases considered. The pre-processed images are shown in Table 3. The values are for all five diseases with five models. The values are noted which are numbers of iteration, as the number of iteration increases, the diseases are identified in specific. The values obtained are plotted using bar chart, and it is evident that the infection is identified only with more numbers and iteration as shown in Fig. 4.

The weights obtained by both techniques are compared to calculate in efficiency of a proposed model. The efficiency of each infection, for each model, is shown in Table 4. These values are plotted as shown in Fig. 5. It is observed that between fifth of sixth iteration, the infection is identified in specific, and hence, the healthy leaf value comes down.

Table 3 Weights of each disease obtained

IT	Leaf disease	Model 0	Model 1	Model 2	Model 3	Model 4
1	Healthy	99,892,813 93,772,137	16,712,600 10,430,181	32,347,725 31,566,061	58,746,134 44,915,400	28,483,203 37,346,215
2	Bacterial spot	87,590,320 92,079,681	99,978,274 99,926,645	28,309,009 46,268,087	20,947,462 69,885,084	41,056,064 20,812,868
3	YLCV	13,841,494 41,679,632	93,675,131 45,999,852	99,173,248 99,936,825	81,539,955 57,848,782	60,871,885 31,273,246
4	Septoria	59,750,591 18,049,400	80,880,192 19,150,148	15,128,533 23,207,408	99,991,322 23,207,408	43,844,339 46,173,584
5	Leaf mold	34,911,890 12,615,236	99,525,062 12,122,111	11,140,285 51,717,105	34,760,145 31,528,813	99,961,406 99,995,613

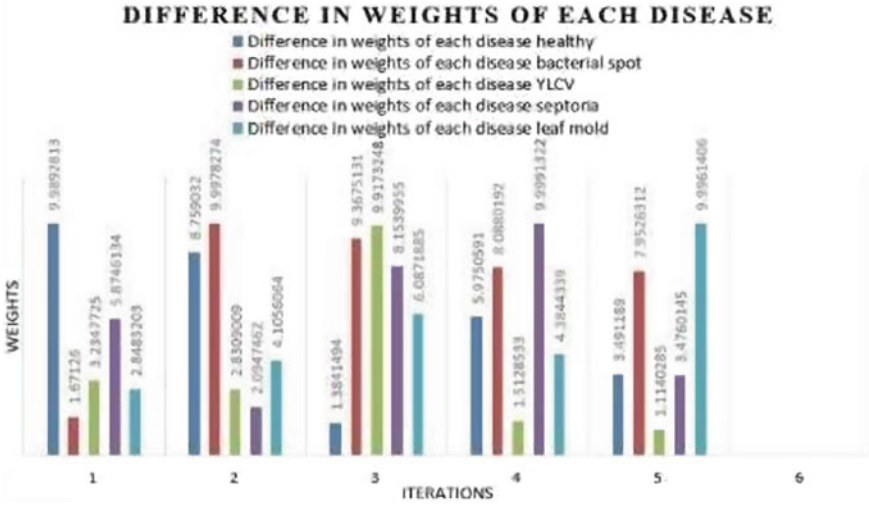
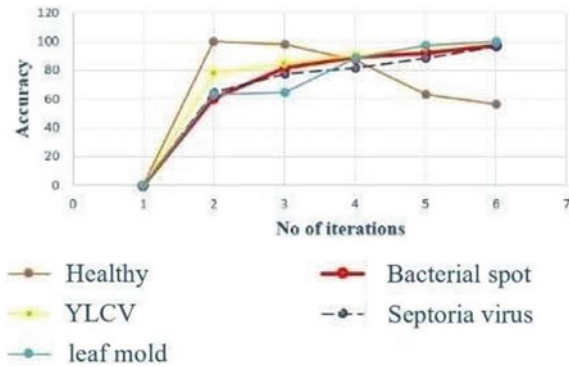


Fig. 4 Bar graph represents weights of each disease obtained

Table 4 Accuracy values of detected diseases

Disease	Accuracy
Healthy	99.89, 89.04, 97.77, 82.45, 67.3%
Bacterial spot	99.98, 73.43, 94.86, 81.88, 99.93%
Yellow leaf curi virus	99.18, 89.94, 78.13, 90.32, 90.32%
Septoria leaf fungus	99.99, 77.53, 99.99, 83.65, 99.98%
Leaf mold fungus	99.96, 88.19, 64.53, 97.56, 99.99%

Fig. 5 Accuracy of detected diseases



5 Conclusion

The proposed model is evaluated using two methods to extract features, identify, and classify. The experimentation has been done using the existing dataset and a sample dataset created by our own images. The proposed model is able to perform better compared to existing models, because of dual model application, the results obtained are shown in Table 4, where the accuracy is calculated for different diseases.

References

1. Anand R, Veni S, Aravinth J (2016) An Application of image processing techniques for detection of diseases on Brinjal leaves using k-means clustering method. In: 2016 5th international conference on recent trends in information technology, 2016
2. Kamlapurkar SR (2016) Detection of plant leaf disease using image processing approach. *Int J Sci Res Publ* 6(2)
3. Zhou C et al (2021) Tomato leaf disease identification by restructured deep residual dense network. *IEEE Access* 9
4. Ding J et al (2020) A tomato leaf diseases classification method based on deep learning. In: 2020 Chinese control and decision conference (CCDC). IEEE, 2020
5. Maity S, Sarkar S, Vinaba Tapadar A, Dutta A, Biswas S, Nayek S, Saha P (2018) Fault area detection in leaf diseases using k-means clustering. In: 2018 2nd international conference on trends in electronics and informatics (ICOEI). IEEE, 2018
6. Kumar V, Arora H, Sisodia J (2020) ResNet-based approach for detection and classification of plant leaf diseases. In: 2020 international conference on electronics and sustainable communication systems (ICESC). IEEE, 2020
7. Saxen F, Werner P, Handrich S, Othman E, Dinges L, Al-Hamadi A (2019) Faceattribute detection with MobileNetV2 and NasNet- mobile. In: 11th international symposium on image and signal processing and analysis (ISPA), 2019
8. Ren S, He K, Girshick R, Sun J (2016) Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell* 39(6) (2016)
9. Yin X et al (2020) Enhanced faster-RCNN algorithm for object detection in aerial images. In: 2020 IEEE 9th joint international information technology and artificial intelligence conference (ITAIC). vol 9. IEEE, 2020
10. Jiang P et al (2019) Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access* 7:59069–59080

Design and System Level Simulation of a MEMS Differential Capacitive Accelerometer



S. Veena , Newton Rai, H. L. Suresh , and Veda Sandeep Nagaraj

1 Introduction

Accelerometers are electromechanical devices which are used to measure the acceleration of a moving system. Wide range of applications require acceleration measurement. These inertial sensors are also useful in applications where motion sensing such as vibration detection, shock and tilt is required [1]. As the MEMS accelerometers are small in size, consume low power and offer high precision, they are widely used in automobiles for airbag deployment, flight control and navigation, smartphones, etc. [2]. These advantages make them suitable for IoT-based applications such as industrial automation, structural health monitoring, condition monitoring of machines, medical applications and many more.

The basic mechanical structure of MEMS accelerometer contains proof mass supported by spring and appended to a dashpot [3]. The spring and the dashpot are in turn connected to frame as shown in Fig. 1. When an external force is applied, the proof mass gets displaced from its rest position and this displacement can be measured by some electronic circuitry to know about the acceleration. Based on the principle of sensing, the accelerometers are classified as piezoelectric, piezoresistive and capacitive accelerometers.

S. Veena (✉) · N. Rai

Department of Electrical and Electronics Engineering, Nitte Meenakshi Institute of Technology, Bangalore, Karnataka, India
e-mail: veena.s@nmit.ac.in

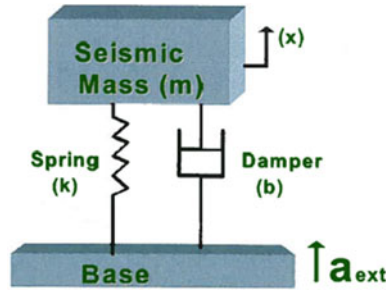
H. L. Suresh

Department of Electrical and Electronics Engineering, Sir M Visvesvaraya Institute of Technology, Bangalore, India

V. S. Nagaraj

Tyndall National Institute, University College Cork, Cork, Ireland

Fig. 1 Basic structure of mass-spring-dashpot



$$F_{\text{total}} = F_{\text{inertial}} + F_{\text{damping}} + F_{\text{spring}} \tag{1}$$

where $F_{\text{inertial}} = -ma$, $F_{\text{damping}} = bx$, $F_{\text{spring}} = Kx$.

A displacement x is caused when a force F , generated by an external acceleration acting on the mass, m , causes a displacement x . The differential equation describing the system response is given by equation

$$m \frac{dx^2(t)}{dt^2} + b \frac{dx(t)}{dt} + kx(t) = F = ma(t) \tag{2}$$

where b is the damping coefficient and K is the spring coefficient, the stiffness.

Using Laplace transformation, the mechanical transfer function can be obtained as

$$\frac{x}{F} = \frac{1}{ms^2 + bs + k} = \frac{1}{m(j\omega)^2 + bj\omega + k} \tag{3}$$

where resonant frequency $(\omega_{\text{res}}) = \sqrt{\frac{k}{m}}$

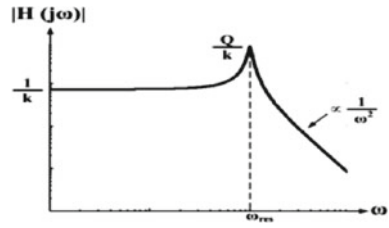
$$\text{Quality factor } (Q) = \frac{\sqrt{km}}{b} \tag{4}$$

The magnitude of Eq. 4 can be plotted as in Fig. 2. It can be seen that for frequencies much lower than the resonance frequency, the system responds with a linear proportional trend of $1/K$. Now considering that the force that will cause the displacement is $F = ma$, it is easy to find out the relationship between x and the acceleration for frequencies much lower than ω_{res} :

$$x = \frac{ma}{k} = \frac{a}{\omega_{\text{res}}^2} \tag{5}$$

As seen from Eq. 5 to have larger displacements of x , so being able to measure it better, it is necessary to have smaller ω_{res} , but lowering it also reduces the system bandwidth, so is required to find a compromise.

Fig. 2 Magnitude response of the second order mass-spring-damper system



Most of the automotive appliances adopt capacitive accelerometers as they provide a very simple capacitive displacement sensing structure. The capacitive accelerometers are small in size, fabrication cost is less, high sensitivity can be obtained, and it is easy to integrate with CMOS for readout circuitry [4].

In this paper, an attempt is made to analyze the behavior of two accelerometers whose dimensions are same but with different structure. One of the devices is accelerated in lateral direction whereas the other in traverse direction.

2 Principle of Operation

The capacitive accelerometers basically comprise a pair of parallel plates, one fixed and the other movable. The parallel plates are very commonly used as interdigital comb fingers as shown in Fig. 3 [5] which can be moved in lateral (in-plane) direction or in vertical (out-of-plane) direction. As the lateral dimensions of the plate can be increased to a few tens of millimeters, it is used for sensing the acceleration in lateral direction, whereas the vertical dimensions of the plates are restricted to only few microns and hence suitable for sensing the acceleration in traverse direction.

From Fig. 3a, it is seen that 3 comb fingers (one of which is movable) form two capacitors, C_1 and C_2 , and any force applied to the movable finger results in a change in capacitance. This differential capacitance may be measured, and thus, the voltage associated with it can be determined using Eq. 6. Out-of-plane forces can be caused due to additional parasitic capacitances such as those between the fingers and the body, as well as the asymmetry of the fringing fields, which can be reduced with more complex designs.

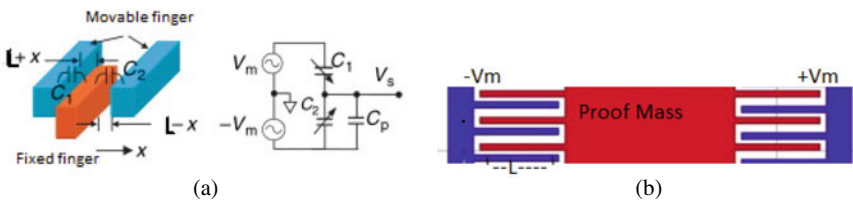


Fig. 3 a Working principle of capacitor. b Lateral accelerometer structure

$$V_s = V_{out} \frac{(C_1 - C_2)}{(C_1 + C_2 + C_p)} V_m \tag{6}$$

where C_p is the parasitic capacitance, C_1 and C_2 are the capacitances, V_m is the voltage applied to electrodes, and V_s is the output voltage.

We know that $C = \epsilon_0 \epsilon_r \frac{A}{d} = \epsilon_0 \epsilon_r \frac{HL}{d}$.

Where H is the thickness of the comb, L is the overlap area of the combs, d is distance of separation between the two combs, and ϵ_0 and ϵ_r are the absolute and relative permeabilities.

As the spacings between the electrodes are symmetrical when there are no accelerations, the capacitance between the movable and fixed fingers is the same. However when in motion, the capacitance does not change equally in both directions, but behaves differently depending on the acceleration direction. If assumed that the fingers are accelerated along the x -axis, the overlap surface on one side will increase by say x which will be added to the overlap length of the comb finger ($L + x$) thereby varying the capacitance and get reduced by same value on the other side of the movable comb ($L - x$) thereby resulting in corresponding change in capacitance on the other side.

In this regard, the magnitude of the two capacitances is given by [6]

$$C_1 = \epsilon_0 \epsilon_r \frac{H}{d} (L + x) \text{ and } C_2 = \epsilon_0 \epsilon_r \frac{H}{d} (L - x) \tag{7}$$

Also, $C_1 = (L + x)$ and $C_2 = \gamma(L - x)$ where $\gamma = \epsilon_0 \epsilon_r \frac{H}{d}$.

We get

$$V_{out} = \frac{(C_1 - C_2)}{(C_1 + C_2 + C_p)} = \frac{x}{L} V \tag{8}$$

So, from the above equation it is seen that the when the fingers are accelerated, the output voltage is directly proportional to the displacement.

Now, from Hooke’s law as applied to Fig. 3b, the displacement of proof mass (x) is given by [7]

$$x = \frac{F}{k} = \frac{M}{k} * a \tag{9}$$

where F is the force applied to the mass of the system (M) by the external acceleration (a) through the springs whose stiffness is given by [8]

$$k = \frac{Etw^3}{4L^3} \tag{10}$$

Therefore, the output voltage which is directly proportional to the displacement is now given by [9]

$$V_{out} = \frac{Ma}{kL} V \tag{11}$$

3 Accelerometer Structure

The two accelerometer structures are as shown in Fig. 4a, b. Model 1 is accelerated in lateral direction, and Model 2 is accelerated in traverse direction.

Model 1: The accelerometer structure shown in Fig. 4a has unique structure with movable and fixed parts. The movable part consists of two proof masses which are symmetrically suspended to a central anchor by a single folded beam on one side and interdigitated sensing fingers on the other side as shown in figure. The two electrodes on either side of the proof mass have the fixed fingers attached to it as shown in figure. In response to the force applied, the two proof masses vibrate, and movable fingers form the interdigitated finger capacitor pair C_1 and C_2 with the fixed fingers across which the differential capacitance can be measured. This differential capacitance varies linearly with the applied acceleration [10].

Model 2: The accelerometer structure shown in Fig. 4b has a micromechanical proof mass which is placed at the central part of the accelerometer and acts as the sensing element. This proof mass is suspended by four serpentine springs and will move with respect to the moving frame of reference when an external acceleration is applied.

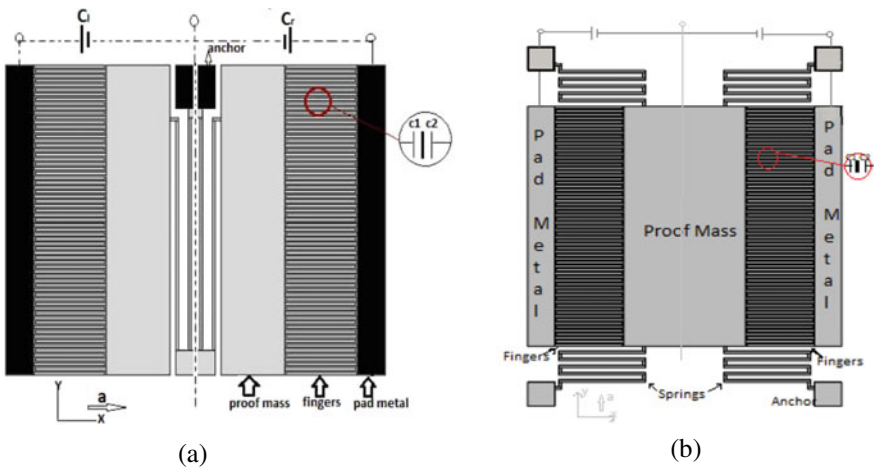


Fig. 4 a Structure for Model 1. b Structure for Model 2

Table 1 Dimensions of the accelerometer structure

	Model 1	Model 2
Proof mass	225 × 1000 × 25 μm (2 nos.)	450 × 1000 × 25 μm
Gap between two fingers (<i>d</i>)	5 μm	5 μm
Length of finger (<i>lf</i>)	250 μm	250 μm
Breadth of finger (<i>bf</i>)	10 μm	10 μm
Thickness of the finger (<i>H</i>)	25 μm	25 μm
Number of fingers on either side	33 pairs	33 pairs
Overlap of fingers (<i>L</i>)	245 μm	245 μm
Pad metal	100 × 1000 × 25 μm	100 × 1000 × 25 μm
Beam/spring length	750 μm	310 μm
Beam/spring width	10 μm	10 μm

The displacement made by the proof mass gives the measure of the acceleration applied to it. In the capacitive approach, the change in the capacitance between the proof mass and the fixed electrodes is measured to determine the displacement. The dimensions of both the models are kept same and are as shown in Table 1.

4 Analytical Modeling

The analytical calculations to find the natural frequency of Model 1 and Model 2 are discussed in this section.

Model 1

The spring constant for the folded beam for Model 1 is found to be 10 Nm from Eq. 10.

The total sensing mass of Model 1 accelerometer is given by [10]

$$\text{Mass } (m) = \text{Proof mass} + \text{moving finger mass} \tag{12}$$

So, from the above equation, the total sensing mass is found to be $m = 3.53625 \times 10^{-8}$ kg.

The natural frequency of the lateral structure is given by Eq. 13 and is found to be 2.3 kHz

$$f_n = \frac{1}{2\pi} \sqrt{\frac{k}{m}} \tag{13}$$

Model 2

The spring constant for the Model 2 is calculated by Eq. 14 and is found to be 115.8 Nm [3].

$$k = 2 \left(\frac{\pi^4}{6} \left(\frac{E t w_s^6}{(2L_1^3) + (2L_2^3) + (2L_3^3) + (2L_4^3) + (2L_5^3)} \right) \right) \quad (14)$$

Total mass of Model 2 is calculated from Eq. 15 and is found to be $4.5435 * 10^{-8}$ kg.

$$m = \rho V \quad (15)$$

where ρ is density of polycrystalline silicon = 2320 kg/m³, volume = $V = t (l \cdot w + 2 \cdot N \cdot l_f \cdot w_f)$.

Substituting for mass and spring constant in Eq. 13, we get the frequency for Model 2 as $f_n = 8.03$ kHz.

5 Simulation Study

The performance of the two accelerometer structures is studied using COMSOL Multiphysics. The frequency analysis and the displacement simulation of the accelerometer models are discussed in this section. Eigen frequency response of the structure is done to analyze the different deformation shapes of the accelerometer structure at natural frequencies. The stationary study of the two models is as shown in Fig. 5a, b. It is noted that Model 1 resonates at 2096.7 Hz and Model 2 resonates at 7834 Hz.

The resonant frequency of the 2 models obtained analytically and from simulations is tabulated in Table 2. It is observed that the analytical and simulated values are almost matching.

Simulations are carried out to understand the effect of frequency, acceleration and stress on the displacement of the proof mass.

Frequency Analysis: Figure 6a shows the frequency analysis of the accelerometer Model 1, and it is observed that the bandwidth is around 1 kHz. Figure 6b shows the frequency analysis of the accelerometer Model 2, and the bandwidth is 2 kHz.

Displacement: The effect of acceleration on the displacement of the proof mass of both the models is tabulated in Table 3. It is observed that the proof mass displacement of both the models is linear with respect to the acceleration.

The analysis of Model 1 is done based on the application of a direct acceleration along the x -axis, i.e., horizontally, to see how the structure moves, how it is stressed and how the capacitance varies. The deviation of the structure is analyzed using a 3D cut line passing from the center of the structure in y direction to detect the movement

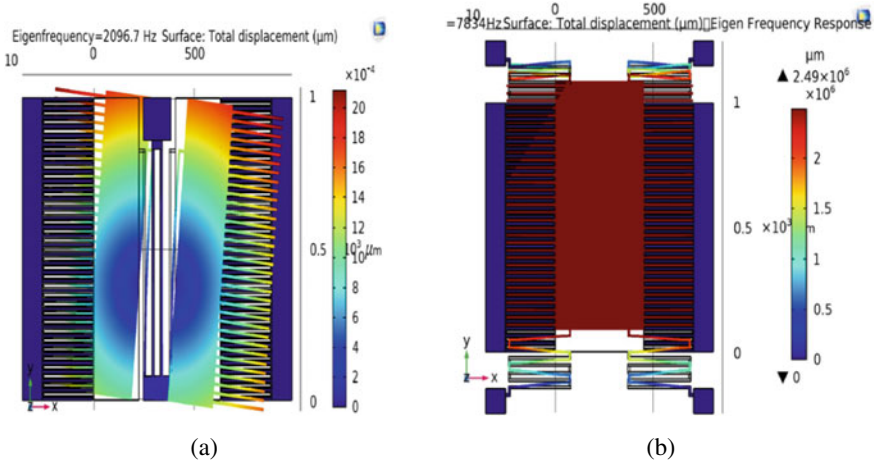


Fig. 5 a Eigen frequency for Model 1. b Eigen frequency for Model 2

Table 2 Comparison of frequencies of the 2 models

Resonant frequency	Analytical value in kHz	Simulated value in kHz
Model 1	2.3	2.1
Model 2	8.03	7.83

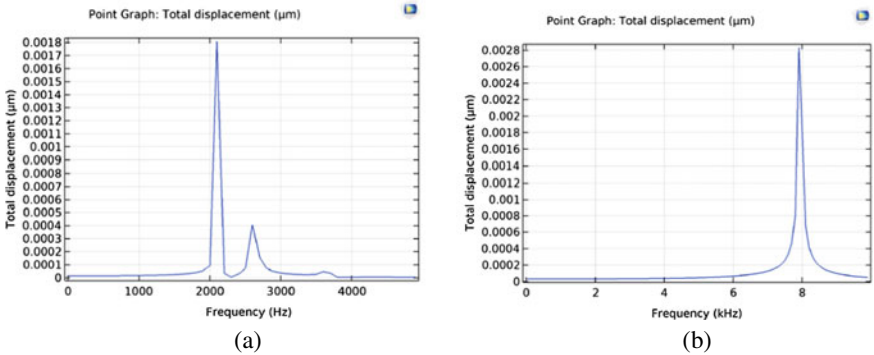


Fig. 6 a Frequency analysis for Model 1. b Frequency analysis for Model 2

of the mass along this line as shown in Fig. 7a. For Model 2, analysis is made by applying acceleration in traverse direction. The displacement in the y direction is analyzed using a 3D cut line along the x-axis while keeping the y constant in the center of the structure as shown in Fig. 7b.

It can be seen that the deviation is in the order of a few nano-meters for Models 1 and 2. Also, it is observed that the maximum deviation for Model 1 is 1 nm and is

Table 3 Effect of acceleration on the displacement of the models

Acceleration (g) in m/s^2	Displacement in μm	
	Model 1 in μm	Model 2 in μm
1	$18 * 10^{-6}$	$25 * 10^{-7}$
2	$35 * 10^{-6}$	$5 * 10^{-6}$
3	$50 * 10^{-6}$	$8 * 10^{-6}$
4	$7 * 10^{-5}$	$10 * 10^{-6}$
5	$9 * 10^{-5}$	$12 * 10^{-6}$
6	$10 * 10^{-5}$	$16 * 10^{-6}$
7	$12 * 10^{-5}$	$18 * 10^{-6}$
8	$14 * 10^{-5}$	$20 * 10^{-6}$
9	$16 * 10^{-5}$	$20 * 10^{-6}$
10	$18 * 10^{-5}$	$25 * 10^{-6}$

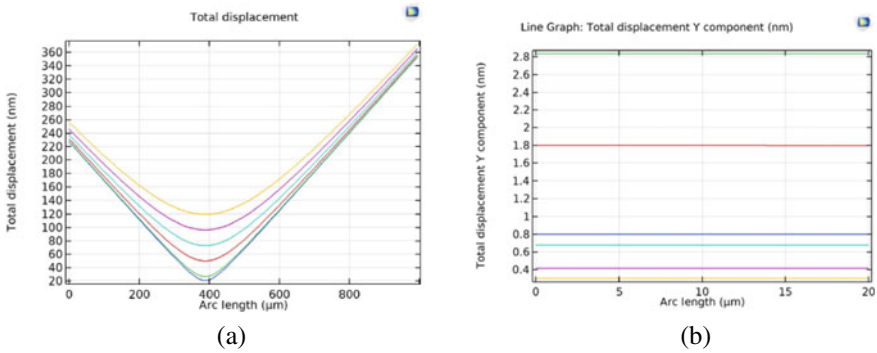


Fig. 7 a Displacement in x direction when passed through 3D cut line with $x = \text{constant}$. b Displacement in y direction when passed through 3D cut line with $y = \text{constant}$

10 nm for Model 2. The stress analysis of the two structures is as shown in Fig. 8a, b. It can be seen that maximum stress is accumulated on the springs at a point where it is attached to the proof mass. Model 1 has 2 springs connected to the either proof mass. So Fig. 8a shows 2 maximum stress points. Figure 8b shows 4 points corresponding to the maximum stress values which relate to the four springs in Model 2.

Capacitance: Electrostatic study is done in COMSOL to find the capacitance of the two models which is shown in Fig. 9a, b. It is observed that the capacitances 1.31 pF and 16.4 pF are obtained from Models 1 and 2, respectively.

To understand the resonant behavior of the two accelerometers, MATLAB simulations are done. The maximum response and phase response of Model 1 and Model 2 as a function of applied external force are as shown in Fig. 10a, b, respectively. It is seen that the natural frequency of the Model 1 is 2.6 kHz and 7.8 kHz for Model 2.

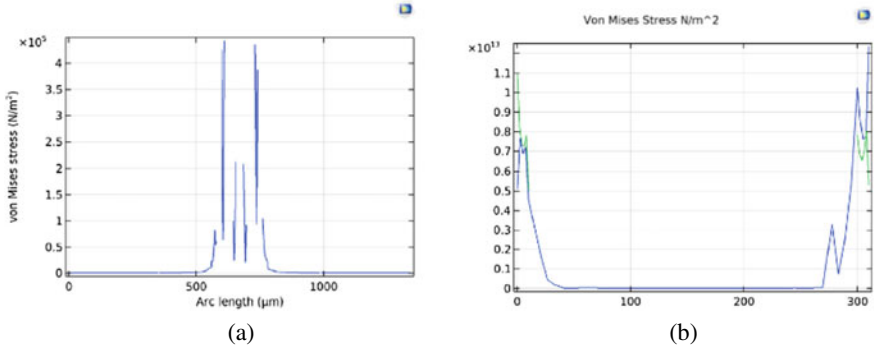


Fig. 8 a Stress on Model 1. b Stress on Model 2

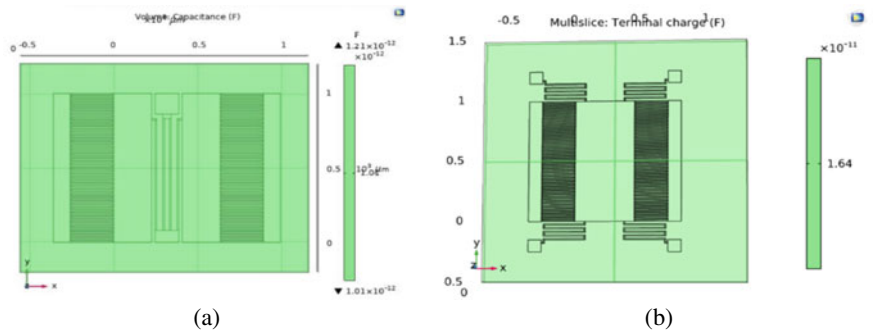


Fig. 9 a Capacitance for Model 1. b Capacitance for Model 2

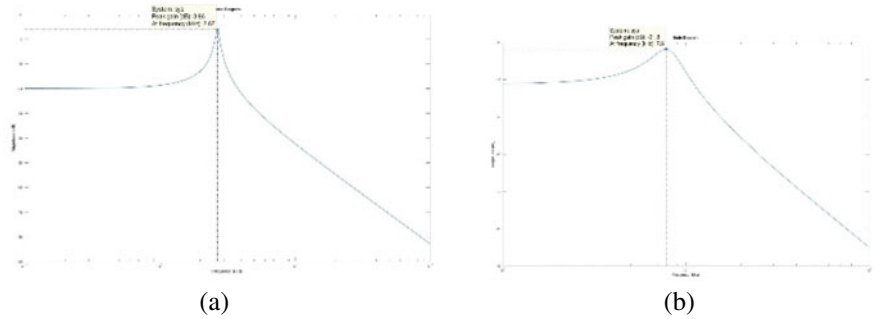


Fig. 10 a Phase plot for Model 1. b Phase plot for Model 2

Table 4 Effect of acceleration on capacitance and displacement

Acceleration (g) in m/s^2	Model 1		Model 2	
	Capacitance in pF	Displacement in μm	Capacitance in pF	Displacement in nm
100	1.31	1600	16.4	5.38
200	1.31	1500	16.4	8.16
300	1.31	1400	16.4	12.8
400	1.31	1370	16.4	20.64
500	1.31	1340	16.4	26.16
600	1.31	1300	16.4	29.93
700	1.31	1200	16.4	35.12
800	1.31	1100	16.4	37.58
900	1.31	1050	16.4	46.49
1000	1.31	1000	16.4	54.71

To understand the effect of large acceleration on the capacitance and displacement, simulations are carried out using COMSOL Multiphysics and are tabulated in Table 4. It is observed that as the acceleration increases, the capacitances remain constant and are in the operating range of the accelerometer. This is because the accelerometer structure moves by a very small value compared to the other quantities, and therefore, the capacitances vary very little.

6 Conclusion

The results from the COMSOL simulations are almost in line with the analytical values calculated. The fundamental issue with these simulations is that the capacitance remains nearly constant with variable acceleration; therefore, a variation of the output voltage cannot be calculated in differential mode from the two capacitances C_1 and C_2 . To get appreciable variations in capacitance, much larger displacements would be required, perhaps several orders of magnitude, both for lateral and traverse motion, and to obtain them, very strong accelerations may be required, which could lead to an excessive accumulation of stress and possibly cause the structure to break; as seen, springs are the points where the most stress accumulates, and these could be the critical points.

It is worth noting that excessive accelerations are not only unnecessary for the purposes for which this gadget is proposed, but they are also nearly impossible to achieve. To use capacitances for position detection, the system's displacement must be greatly increased, which can be accomplished by expanding the entire volume. But, increasing the volume without increasing the K of the springs, or perhaps without increasing their size, could result in stiffening of the system springs. Furthermore,

as the springs are too small in comparison with the rest of the system, they may be subjected to excessive stress, resulting in structural damage.

One more method can be thought of is to reduce the distance between the fingers but however this is not possible as further reduction in distance between the fingers may not be acceptable for fabrication. When the results of both the models are compared, it is observed that the natural frequency of the Model 1 is 2.1 kHz and that of Model 2 is 7.8 kHz. Model 2 exhibits better displacement for various accelerations as compared to Model 1.

Acknowledgements This work is a part of ISSS community chip activity. The authors are grateful to Prof. Ananth Suresh, Chairman, ISSS, IISc Bangalore, for giving the opportunity to participate in community chip activity supported by ISSS, IISc, Bangalore. The authors are thankful to Dr. Habibuddin Shaik, Associate Professor, Physics Department, NMIT, Bangalore, for the support extended in carrying out this work. The authors also thank Mrs. Nithya and Mrs. Stuthi, Centre for Nano-Materials and MEMS, NMIT, Bangalore, for their timely support and the Department of Electrical and Electronics Engineering and authorities of Nitte Meenakshi Institute of Technology, Bangalore, for the continuous support and encouragement. The authors extend their sincere gratitude to Visveswaraya Institute of Technology, Belagavi, for the opportunity and support.

References

1. Mukhiya R, Agarwal P et al (2019) Design, modelling and system level simulations of DRIE-based MEMS differential capacitive accelerometer. *Microsyst Technol*. <https://doi.org/10.1007/s00542-018-04292-0>
2. Vijayakumar S, Vijila G, Alagappan M, Gupta A (2011) Design and analysis of 3D capacitive accelerometer for automotive applications. In: COMSOL conference, Bangalore
3. Puccioni G (2020) Design and analysis of a MEMS capacitive accelerometer. *Sens Microsyst J*
4. Xie H, Sulouff RE (2008) Capacitive accelerometer. *Compr Microsyst*
5. Senturia SD (2001) *Microsystem design*. Kluwer Academic Publishers. ISBN 0-7923-7246-8
6. Kannan A (2008) Design and modeling of a MEMS-based accelerometer with pull in analysis
7. Padmanabhan Y (2017) MEMS based capacitive accelerometer for navigation, 20 Apr 2017. <https://doi.org/10.13140/RG.2.2.35625.49769>
8. Singh P, Srivastava P, Chaudhary RK, Gupta P (2013) Effect of different proof mass supports on accelerometer sensitivity, pp 896–900. <https://doi.org/10.1109/ICEETS.2013.6533506>
9. Sinha S, Shakya S, Mukhiya R, Gopal R, Pant BD (2014) Design and simulation of MEMS differential capacitive accelerometer. In: *Proceeding of ISSS international conference on smart materials, structures and systems*, Bangalore, India, 8–11 July 2014
10. Veena S, Rai N, Suresh HL, Nagaraja VS (2021) Design, modelling, and simulation analysis of a single axis MEMS-based capacitive accelerometer. *IJETT J*

Auto-Load Shedding and Restoration Using Microcontroller



G. L. Harsha, A. S. Prathibha, S. A. Rakshit Kumar, P. G. Suraj, M. J. Nagaraj, and V. Shantha

1 Introduction

Modern electrical power system is itself a very complex technical system which had been developed and built by the mankind. The main aim of such service is to provide a continuous and uninterrupted service to the consumer end. With civilization, the power systems need to be expanded to meet the needs. However, constructing a new transmission lines or building a new generation plants to have a reliable service is difficult task. Since installation of a plant would take roughly around three to four years and moreover it is a high budget installation requiring huge capitals. In concern with providing services, a best way to mitigate such issues would be load shedding. A load shedding is a common practice that takes place when the demand for power is generally more than the generated power. To make sure that the system is stable and available during all the conditions, under a main power station, there are several substations which handles power-cut for a certain duration to cope up with this shortage in electrical energy.

When the power system blackouts occur due to abnormalities [1], it causes huge losses to the customers as well as utility. This disturbance spreads into larger areas and leads to complete system failure and also unexpected risks [2]. Even the time for restoration of the networks cannot be estimated. Hence, to overcome all these issues, load shedding concept plays prominent role.

G. L. Harsha · A. S. Prathibha · S. A. Rakshit Kumar · P. G. Suraj · M. J. Nagaraj (✉)
Department of E&EE, NMIT, Bengaluru 560064, India
e-mail: nagaraj.mj@nmit.ac.in

V. Shantha
Department of ME, Sir MVIT, Bengaluru 562157, India
e-mail: shantha_mech@sirmvit.edu

2 Background

Conventionally, practiced load shedding techniques are very slow, and also, they do not calculate the correct amount of load that need to be shed. This will cause unnecessary load shedding [3]. This would decrease the efficiency of the system as they shed excessive loads and providing inconvenience to the customers.

Load shedding is classified in three types: (1) traditional, (2) semi-adaptive and (3) adaptive depending on the strategy involved in load shedding and restoration [4]. Traditional load shedding normally called as conventional load shedding is most used among these three categories because it is simple and doesn't require complicated relays. Under this scheme, certain amount of load sheds when system frequency falls below a certain threshold level. In semi-adaptive scheme, rate of change of frequency, ROCOF in system is compared with the system frequency reaching the threshold [5]. Based on the value of ROCOF, certain amount of load is shed. Adaptive scheme uses real-time data which are collected from phasor measurement unit, and using that, disintegration of the system into several islands where generation tripping with initiation of load shedding is adapted in a controlled manner [6].

Whenever there is a heavy machine is switched on, a drop in frequency at lines is general, and this should not be considered as a pickup value for load shedding. Instead, the rate of fall of frequency over the given time lag, i.e., df/dt [7]. Many blackout data in the power distribution network revealed that voltage stability is also important for a power system else they would cause undesirable disturbances. Frequency as well as voltage are affected on loading, and hence, frequency and voltage are used as parameters for load shedding [8].

3 Basic Design

The prototype developed comprises the power supply module, the sensing unit, the dynamic load controller, the programmable control switch, the distribution feeders, and the output display for other type of fault related to power line. The sensing unit serves as input to the dynamic load controller. The available power is input to the system through the analog pin of microcontroller. The dynamic load controller contains the program that implements the load shedding based on available power. The generation represents the available power in the transmission line. The load represents the load point. The programmable unit helps the relay switch to transfer power automatically to the load. The system will help improve the load shedding schedules, accuracy, quality, and fairness. Figure 1 shows block representation of implemented technique.

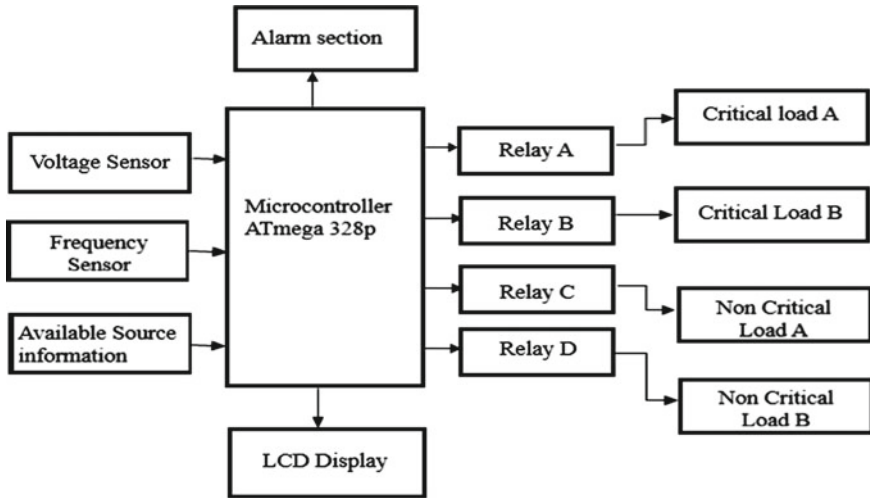


Fig. 1 Block diagram representation of auto-load shedding and restoration

4 Components for Auto-Load Shedding and Restoration

The implementation of this system requires the following major components as shown in Fig. 2.

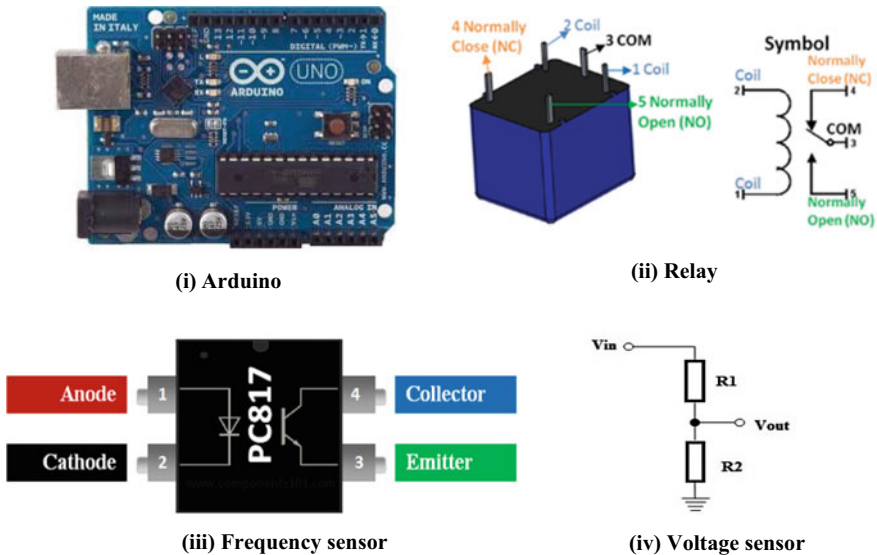


Fig. 2 Major components used for auto-load shedding and restoration

Sensors are basically a device which can sense or identify and react to certain types of electrical or some optical signals.

$$V_{out} = \frac{V_s * R_2}{R_1 + R_2} \tag{1}$$

From Eq. 1, the output voltage can be calculated based on ohms law where

- V_s = source voltage, in volts (V)
- R_1 = resistance of the 1st resistor, in ohms (Ω)
- R_2 = resistance of the 2nd resistor, in ohms (Ω)
- V_{out} = output voltage, in volts (V).

5 Simulations with Result and Discussions

Figure 3 shows the simulation circuit for automatic load shedding and restoration using microcontrollers with underfrequency relay circuit, by using Proteus simulation software.

The model is subdivided into:

- Frequency sensing unit (input to microcontroller)
- Load shedding block (controller actions are given from microcontroller).

Frequency sensing unit

Figure 4 shows the model of opto-coupler coupled with step-down transformer [9], by using Proteus simulation software.

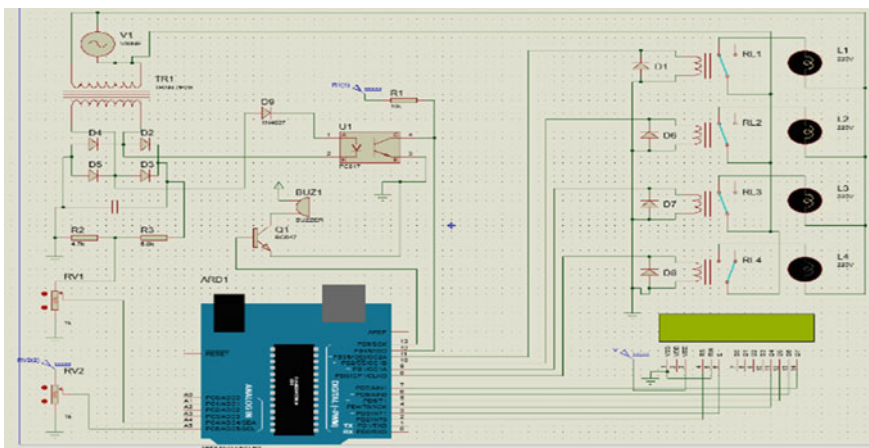


Fig. 3 Simulation circuit for auto-load shedding and restoration

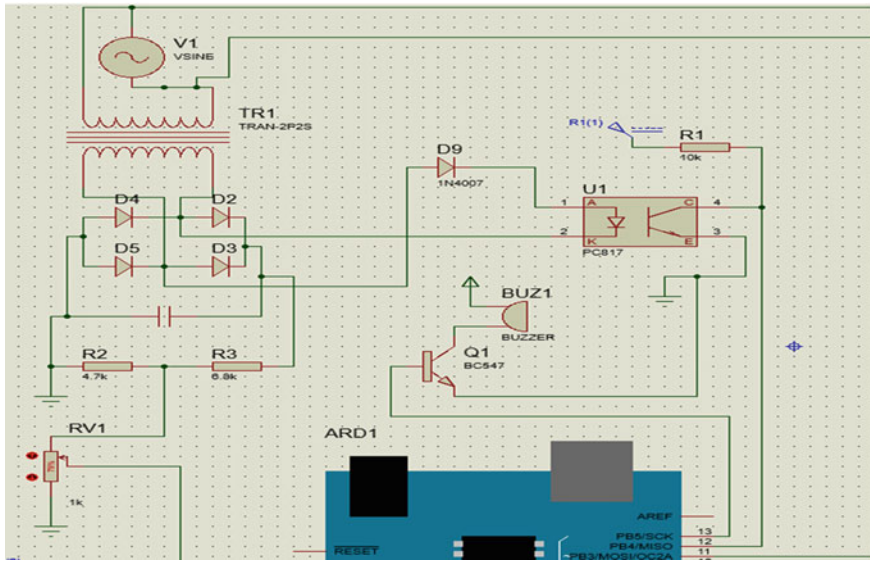


Fig. 4 Simulation circuit for frequency sensing unit

It is a four-step load shedding with 25% of total load shed at each step as shown in Table 1. A conventional method of auto-load shedding using microcontroller is very useful as they provide accurate and error-less operation.

The amount of load that need to be shed is just necessary to bring back the system frequency (50 Hz). That means the load that need to be shed is nearly equal to amount of overload that is being calculated [10].

It is not necessary to restore the frequency exactly for 50 Hz instead it could be above 49 Hz, remaining would be recovered by the system generation action by speed governor. Frequency levels at which load shedding are initiated depend on various factors. In an interconnected system, a frequency deviation of 0.2–0.3 Hz would indicate severe disturbance in the system. Hence, load shedding must be initiated with 49.3 Hz of decrease in system frequency. Load shedding must be coordinated with the equipment operating since most of the undesirable phenomenon would take place at reduced frequency. The increase in load demand compared to generation would cause performance of the power plant outputs to reduce. And this would drive system toward unbalanced condition further decreasing the efficiency of the plants.

Table 1 Four step load shedding scheme

Steps	% Load shed
Step-1	25%
Step-2	25%
Step-3	25%
Step-4	Complete interruptions

Typically, 46 Hz is considered as the margin for maximum frequency decay in the system. Most instances limit the frequency drop till 47 Hz only. Load restoration is the reverse process of load shedding which takes place when frequency returns to normal value. The complete algorithm is shown in Fig. 5.

Load shedding is initiated at 49.3 Hz, and maximum permissible frequency drop allowed in the system is 47 Hz. The minimum time delay of 1 s is taken and around 0.1 s for initiation of breakers.

Step-1 Load Shedding (25%)

Pickup frequency setting: 49.3 Hz.

Relay operational time delay: 1 s.

Breaker time: 0.1 s.

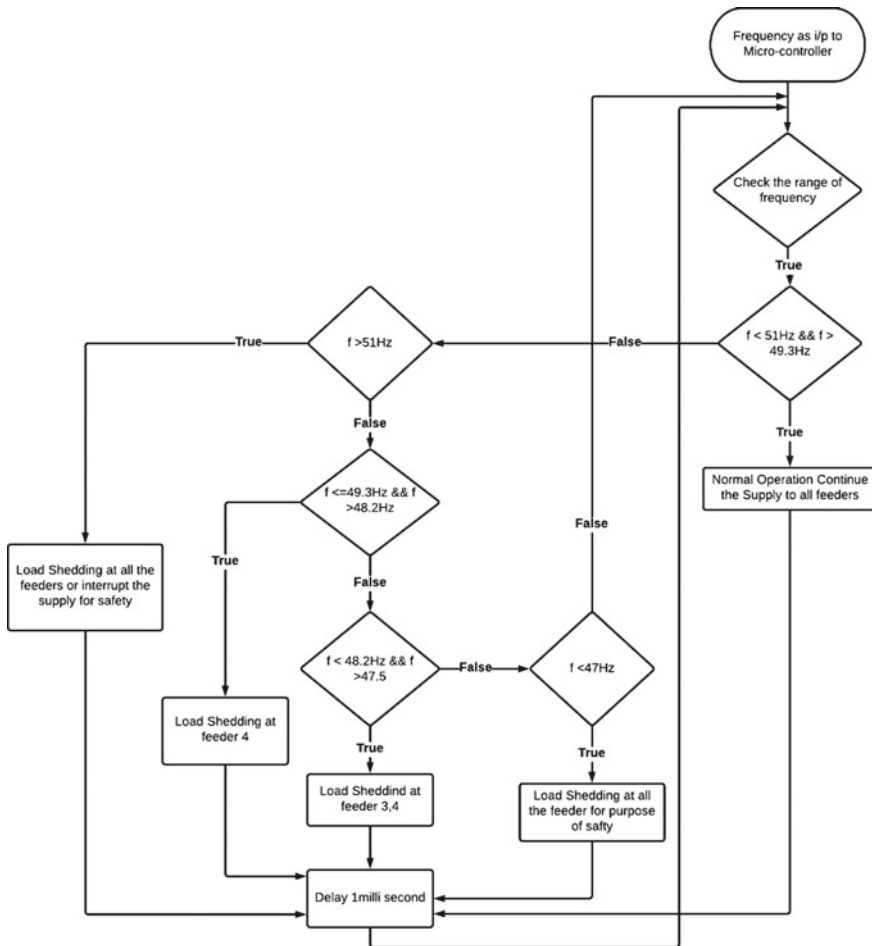


Fig. 5 Flowchart of load shedding and restoration based on value of frequency

Step-2 Load Shedding (25%)

Pickup setting at second step should be such that before frequency reaches 2nd threshold limit, 1st step load shedding must be performed, i.e., for 25% overloading, 1st step pickup would have taken 1 s and load sheds for 49.3 Hz at 1.3 s. Hence, at step 2, relay could be set at 48.2 Hz.

Step-3 Load Shedding (25%)

Similarly, pickup should be done once steps 1 and 2 load shedding are performed.

Pickup frequency setting: 47.5 Hz.

Relay operational time delay: 1 s.

Breaker time: 0.1 s.

Hence, load shedding is initiated at 47.3 Hz.

Step-4 Load Shedding (25%)

The above 3 steps are for permissible frequency operations and take place above 47 Hz. A still drop in frequency is not to be entertained. All load and generation damping must happen around this frequency ranges (49.3–47 Hz). Hence, load shedding takes at higher frequency, and system frequency would recover nearer to 50 Hz. Below the maximum permissible values (47 Hz in general but could be counted to 46–45 Hz), system supplies to the load demands must be shut down.

Consider the situation, for example, whenever there is a need to have any maintenance in the power generation units, then we go for a temporarily shutting down the units. This would be the main reason for decrease in the power generation. Though alternative means like backup arrangements were taken, these would not be sufficient if the units are not back to service for a prolonged hours. Another example would be failure in the turbine governing actions at hydel power generation could also produce lesser output. Hence, to have a bridge between supply and demands, we can generally initiate load shedding at the secondary distribution ends with controllers at the primary distributors (Substation).

The percentage of load demand is usually calculated as by Eq. 2.

$$\% \text{ overload} = \frac{\text{Load Demand} - \text{Remaining Generation}}{\text{Remaining Generation}} \quad (2)$$

Based on Eq. 2, the amount of load demand over generated power is calculated to bring load shedding as well as restoration. With increase in loading on feeder system or if generated power itself has been reduced, then a supply and load demand vary; hence, there would be decrease in voltage as well as a frequency. Using these parameters, auto-load shedding as well as restoration shown in Fig. 6, at duly time, is implemented as discussed in Sect. 1.

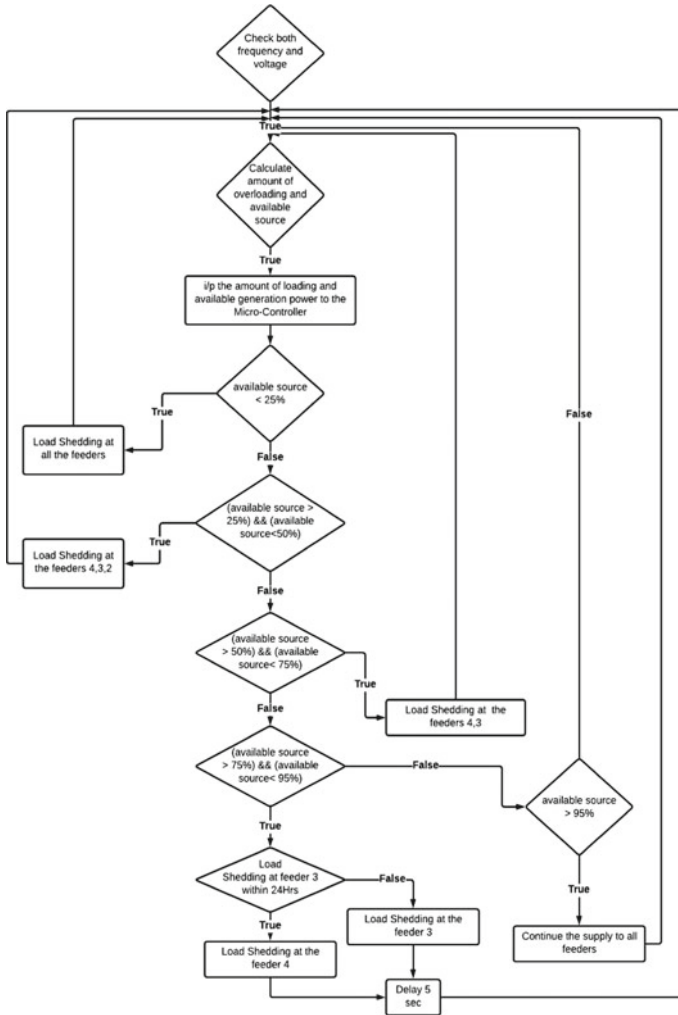


Fig. 6 Flowchart of load shedding and restoration with decreased generated power

6 Conclusion

In this paper, the design and construction of automatic load shedding and restoration in system. This work presents an enhanced solution to the challenges of poor scheduling of load shedding. The system integrated sensor, microcontroller, LCD display, relay switch to automate the load shedding system. The proposed system provides information about voltage and frequency. The automation of the load shedding system will help in ensuring fairness, accuracy, and efficiency in scheduling as human errors are eliminated. The operational values of substation are generally

66/11 kV or 66/33 kV; hence, kVA rated circuit breakers are required to integrate this module. This packed module can be installed at the substations with the help of contactors and interposing relays.

7 Future Scope

Wide area measurement system (WAMS) is a hot topic in research of power system domain as there is increase in use of advanced real-time measurement system for collection of data in power system (like frequency, voltage, current, and generation capacity). This paper is a complete setup for required working, and in future, many other function and features like IoT module can be added for more reliable operation. Another significant field would be WAMS-based load shedding. Wide area measurement system uses adaptive relaying technology to achieve greater adaptability [6]. Underfrequency load shedding and restoration are brought using supervisory controls where at significant generation loss or loads, load shedding scheme is implemented by formation of islands.

References

1. Wu Y-K, Chang SM, Hu Y-L (2017) Literature review of power system blackout. In: 4th international conference on power and energy systems engineering, CPSE 2017, Berlin, Germany, 25–29 Sept 2017
2. Alhelou HH, Hamedani-Golshan ME, Njenda TC, Siano P (2019) A survey on power system blackout and cascading events: research motivations and challenges, 20 Feb 2019
3. Hirodantis S, Li H, Crossley PA (2019) Load shedding in a distribution network. In: International conference on sustainable power generation and supply, Nanjing, China
4. Shafiullah M, Alam MS, Hossain MI, Ahsan MQ (2014) Impact study of a generation rich island and development of auto load shedding scheme to improve service reliability. In: International conference on electrical engineering and information & communication technology (ICEEICT)
5. Ahsan MQ, Chowdhury AH, Ahmed SS, Bhuyan IH, Haque MA, Rahman H (2012) Technique to develop auto load shedding and island scheme to prevent power system blackout. *IEEE Trans Power Syst* 27(1)
6. Phadke AG, Thorp JS. *Computer relaying for power systems*, 2nd edn
7. Perumal N, Amran AC (2003) Automatic load shedding in power system. In: National power and energy conference (PECCon) 2003 proceedings, Bangi, Malaysia. 0-7803-8208-0/03/\$17.00©2003 IEEE
8. Joshi P. Load shedding algorithm using voltage and frequency data. https://tigerprints.clemson.edu/all_theses/240
9. <https://simple-circuit.com/arduino-frequency-meter-220-380v-ac/>
10. Berdy J (1968) Load shedding—an application guide. General Electric Company, Electric Utility Engineering Operation, Schenectady, NY

A Review on Design and Performance Evaluation of Privacy Preservation Techniques in Data Mining



Jagadevi N. Kalshetty  and N. Nalini 

1 Introduction

As the world is moving towards digitization in all sectors like education, banking, voting, education, transportation, etc. [1]. There are possible chances of attacks in all fields like denial of service, man in the middle attack, malware, phishing, impersonation attack, etc. All of these attacks can be either active or passive. These attacks alter the results hence the security is compromised, there are many protocols used to protect the privacy of users known as data hiding. User's trust is very important in sectors like banking, payroll, voting, etc. [1]. Users trust is questioned in many ways during hacking attacks, and there are many privacy preserving mechanisms for data encryption.

Data mining is another technique used to rearrange large sets of data or patterns to extract useful information [2]. The data is collected and assembled in warehouses on particular areas of interest for efficient analysis in effective decision-making cost reduction, etc. Data mining to obtain necessary information without privacy leakage is an import task. Many algorithms have been designed for preserving privacy through data mining.

Data mining is a key player in healthcare; the huge amount of data generated by the healthcare transforms this data into useful form for decision-making [3]. It also helps healthcare industries to detect fraud and abuse, customer relationship management decisions, evaluation of cost of treatment, identifying the risk factors associated with diabetes [4]. Data mining in healthcare uses EHR systems, which provides a view of person's health stored at different locations, connected to a central mining server. This central server preserves the privacy and stores the information from different her systems by using classification and clustering.

J. N. Kalshetty (✉) · N. Nalini
Nitte Meenakshi Institute of Technology, NREA UOM, Bengaluru, India
e-mail: jagadevi.n.kalshetty@nmit.ac.in

The main goal of the DPPP is to maintain data confidentiality; there are three privacy preservation techniques, randomization, anonymization and encryption [5]. When randomizing, the noise is added to the original data, the noise is large enough for the individual values of the records to no longer be recorded [6]. In anonymization, the particular individual record may be made indivisible in a group of records using the generalization and aggregation technique k [7]. Encryption is the process for encrypting information. This process involves the conversion of original information, referred to as plaintext, into an alternative form known as cipher text.

Cyber threats are a major concern these days there are many techniques developed to overcome cyber threats, using privacy preserving techniques we can protect the data by privacy preserving. Privacy preserving method include blockchain, authentication and cryptography [7]. Cryptography is a technique of providing secure communications that allow only the sender and intended recipient of a message to view its contents. Here, data encryption is done using a secret key. After the encryption, both the encoded message as well as secret key are sent to the recipient for the process of decryption. Block chain collects information as blocks that hold the information together. As huge amounts of heterogeneous data is collected continuously through IoT, privacy preserving mechanisms are used to protect user's privacy.

IoT provides interconnection among various heterogeneous devices [8]. Data is collected from the sensors about machines, human beings as well. Despite there are many advantages of collecting data there are many third party intruders who can mine the data extract necessary information. In IoT, large amounts of data is continuously collected by different sensors, IoT sensors includes: pressure sensors, humidity sensors, proximity sensors, accelerometers, level sensors, gyroscope, infrared sensors, gas sensors, temperature sensors, and optical sensors [9]. Data is classified into three types depending on its structure structured, unstructured, semi-structured data. Privacy preserving data mining is gaining more popularity because it preserves the quality of data without altering it. Collection of information across various sources of data while preserving the privacy of data.

2 Privacy Concerns in Data Mining

As the saying goes these days, "Data is the new gold", the importance of data is not unknown to anyone. To make data readable and interpretable, we deploy various data mining techniques and in this process, make our data vulnerable to various privacy concerns. These privacy concerns may arise from lack of awareness, personal embracement, or surveillance. To maintain the confidentiality, integrity and availability of data, we need PPDm methods. PPDm methods make sure that the results acquired after data mining is intact in terms of confidentiality and integrity.

3 Privacy Preservation Techniques

In [9] a recommendation system is nothing but a subset of *information filtering system* that are used to predict the “rating” or “preference” a user would provide to a particular item. The recommendation systems become difficult when there is huge amount of data for the recommendation system to process along with this many artifacts can increase the information overload. Recommendation technique collect the device data and use publicly available information to recommend preferred items depending on their interest to the user.

Mobile recommendation system (MRS) model: Is a mobile recommendation system that generates recommendation for mobile users in an interconnected network via Internet. Privacy concerns regarding the data theft may reduce the intentions of the users to use the recommendation system.

This system may suffer from privacy issues and threats as a result of sensitive user information. To address these issues in this document, a data collection protocol based on Reverse Data Transformation (RTD) that uses the distributed and collaborative functionality environment in MRS. This mechanism does not limit itself to protecting against internal and external security breaches. The Reverse Data Transformation (RTD) algorithm may disturb and restore the data. In this algorithm, we use a dynamic weighting mechanism, which adjusts the degree of data disruption to improve privacy flexibility. In addition, to determine whether the disrupted data has been altered. Compared to existing algorithms, RTD has more reserves of knowledge and is better when it comes to efficiently reducing data loss and confidentiality risks the privacy protocol for RTD-based data collection does not need a private channel hypothesis.

In [10], for the functioning of the matrix, the authors implemented the QR decomposition in the framework of the SPDZ for the reflection of household members. Analysis and comparison of QR and LU decomposition performance across different data sizes in actual scenarios is performed. A very comprehensive extraction task is performed on the vehicle driving registers to assess their accuracy and efficiency. The data flow is optimized by extracting the encryption in plain language from the SPDZ protocol. The document suggests the use of the SPDZ protocol linked to the QR and LU decomposition for MPC [11]. The demand for data mining on healthcare is analyzed and results for critical diseases is improved source anonymous privacy preserving distributed healthcare data collection and mining technique is used to improve the accuracy of EHR systems. The proposed system is efficient and preserves privacy in terms of healthcare. The accuracy of the healthcare predictions is improved in coordination with individual EHR systems.

In this paper a scheme for improving the healthcare services by accumulating all EHR systems data at one nodal data mining server, at the same time also keeping the privacy intact using k -source anonymous. This scheme is also collision resilient, which preserves the purpose of privacy in case of collision among malicious EHR systems and central data mining server. Limitations of this paper are dynamic joining, in the join type two or more fields from two data sources are joined using a join condition that changes dynamically and leaving of the EHR systems.

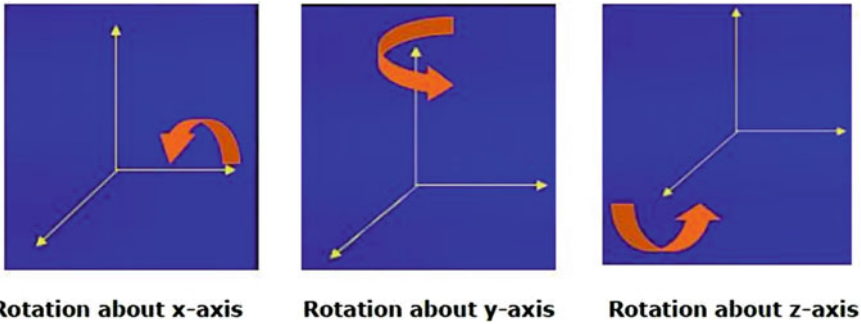


Fig. 1 Transformation using rotation data disturbance (RDP) [12]

The main objective achieved in the proposed system in [1] is verified voters, Duplicate vote detection, protected voter details, Online Ballot Integrity, Vote storing and Verification, End-to-End Security. The basic idea here is that the user can login with the user details. ECS server provides some values to poll the votes the voting server communicates and verify the details of the user on the ECS server and provide the online ballot based on the ward candidate list.

Cube-structured data storage is used to store encrypted data in order to facilitate data recovery from the cloud used to check the integrity of the user. To secure voting and results, verification relies on a user-differentiated system model. It has five entities: Users, Trusted ECS Server, Trusted Vote Verification Server, Online Voting Server and Cloud Voting Storage server. The proposed system consists of three phases, three phases: the first consists in saving user data, candidate data offline. Phase two is the voting phase, and phase three is the voting results announcement phase.

Transformation using rotation data disturbance (RDP) involves the rotation of a point present in the axis of the coordinates into different axes without affecting the metric [4]. Geometrically disturbed data will be output from actual data sets (Fig. 1).

The algorithm proposed in this paper consists of a original matrix m, n number of attributes are considered for perturbation, Singular value decomposition takes a rectangular matrix of gene expression data (defined as A , where A is a $n \times p$ matrix) in which the n rows represents the genes, and the p columns represents the experimental conditions. The SVD theorem states:

$$A_{n \times p} = U_{n \times n} S_{n \times p} V_{p \times p}^T$$

where the columns of U are the left singular vectors (gene coefficient vectors); S (the same dimensions as A) has singular values and is diagonal (mode amplitudes); and V^T has rows that are the right singular vectors (expression level vectors). The SVD represents an expansion of the original data in a coordinate system where the covariance matrix is diagonal. The features, which are having smaller values than the threshold, will be set to zero.

Differential protection is a framework for openly sharing data about a dataset by depicting the examples of gatherings inside the dataset while keeping data about people in the dataset [13]. The thought behind differential protection is that assuming the impact of making a subjective single replacement in the data set is sufficiently little, the inquiry result cannot be utilized to derive much with regards to any single individual, and hence gives security. Laplace instrument is the workhorse of differential protection, applied to many cases where mathematical information is handled. Nonetheless, the Laplace component can return semantically incomprehensible qualities, like negative counts, because of its endless help are two famous answers for this, bouncing/covering the result esteems and jumping the system support. Gaussian component is a fundamental structure block utilized in huge number of differentially private information investigation calculations. The Gaussian instrument is an option in contrast to the Laplace component, which adds Gaussian commotion rather than Laplacian clamor.

4 Literature Survey

S. No.	[Reference] year	Technique/methodology/algorithm	Performance	Dataset	Research gaps
1	[4] 2021	Singular value decomposition (SVD) and 3D rotation data perturbation (RDP) for preserving privacy of data	Balances the data privacy and utility effectively	Adult heart disease dataset	Perturbation technique to improve accuracy
2	[11] 2020	K-source anonymity scheme, collusion resilient scheme	Efficient in terms of computation cost	EHR systems	In this algorithm complexity time consumption should be measured as low
3	[8] 2020	Optimization algorithm for privacy preserving attack and defense (PPAD)	Maximum privacy leakage for privacy defender	Game data (PPAD)	Game model evolving into a Steinberg game
4	[10] 2020	One-hot encoding and LU decomposition based on SPDZ protocol to support MPC	Better performance compared with the Cholesky factorization version AND conjugate gradient descent version	Cotton trading records and vehicle driving logs	Mining results should be used by single machine implementation system

(continued)

(continued)

S. No.	[Reference] year	Technique/methodology/algorithm	Performance	Dataset	Research gaps
5	[14] 2019	DeepChain	DeepChain guarantees data privacy for each participant and provides audit ability for the whole training process but the encryption part and adding data on the chain consumes a considerable amount of time	Data collected from blockchain	Security should be redefined
6	[3] 2018	Medical agglomeration behaviors mining (MAMB)	MABM algorithm has better scalability, more efficient in running parameter than Eclat algorithm and apriori algorithm	Medical insurance industry	Firstly threshold need to be determined and then the fraud of the person's card can be identified
7	[5] 2018	Ant colony optimization, random rotation perturbation, K-means clustering	The proposed algorithm is better in utilization and a combination of K-means clustering algorithm. This method protects the privacy of individual more accurately	Real-time healthcare	Different types of data hiding techniques can be used by choosing suitable QI attributes

(continued)

(continued)

S. No.	[Reference] year	Technique/methodology/algorithm	Performance	Dataset	Research gaps
8	[7] 2018	PPDM hybrid algorithm (cryptography and perturbation)	In this paper, the data is converted in to their respective ASCII values then apply perturbation techniques, after then cryptography technique was applied on the data	archive.ics.uci.edu (machine learning, UCI) (Indian liver patient dataset, balance scale dataset, abalone dataset, and bank marketing dataset)	We can also try to work with the video and audio data
9	[2] 2018	Privacy preserving item-centric mining algorithm PP-UV-Eclat	It is used in the Apache Spark environment to find frequent patterns	Dataset is not disclosed	PPDM mining can be improved and publishing step can be shifted from the post processing step to intermediate processing step
10	[15] 2017	Data mining tools which are efficient in detecting upcoding frauds	Efficient review	Healthcare	Semi-supervised method of learning would be highly appreciable in fraud detection
11	[6] 2017	Privacy preserving hybrid technique (suppression and perturbation)	Information loss is zero, execution time are minimum and privacy preserved is maximum	Database stored on MS Access	Does not throw light in case of very huge datasets with high dimensionalities
12	[16] 2017	Multi-party privacy preserving data mining for vertically partitioned data	Performance in the terms of accuracy is high, error rate is low, time consumption high and memory consumption is low	Student data	In this algorithm complexity time consumption should be measured as low

5 Comparison of PPDM Techniques

Privacy and its preservation are very big concerns in data mining. Now there is an abundance of ways, algorithms, and techniques for privacy preservation. The paper throws light on some of them. Table 1 shows a comparison of various PPDM techniques along its methods, scenarios, advantages and limitations.

Table 1 Comparison of PPDM techniques

S. No.	Techniques	Methods employed	Scenarios	Advantages	Limitations
1	Anonymization	k-anonymity-model, algorithm apriori based	Centralized	It reduces granularity of data	Anonymized data appears less meaningful
2	Cryptography	ECB algorithm	Distributed	Better privacy and minimal data loss	Time consuming due to heavy calculations
3	Perturbation	Adding noise, removing noise (reducing data)	Central and distributed both	Quality of data is maintained	Possibility of data loss
4	Clustering	K-means	Centralized	Storage required is very less	Sensitive to outliers
5	Classification	Decision tree, support vector machine, Naïve Bayes	Central and distributed both	Reduces cost by eliminating irrelevant data	Probability of privacy attacks
6	Neural network based PPDM	CNN, deep learning	Centralized	Efficiency up to 93–97% is received	Space required is more
7	Blockchain	Proof of work (POW), proof of stake (POS), BFT protocol	Decentralized/distributed	Immutability of data is ensured, highly time ordered and secured	Costlier and hard to implement
8	Randomization	Adding noise, resampling	Centralized and distributed but highly effective in centralized	Increases privacy preservation by hiding correlation	Not efficient in distributed scenarios

6 Conclusion

Privacy preservation becomes a crucial issue in the developmental phase of data mining. PPDM has become tremendously popular because it enables sharing of confidential data for analysis purposes. In various papers, we have the importance of privacy preserving in various fields like voting, banking, education, and various data mining techniques like cryptography utilizing encryption and decryption method. Various privacy preserving data mining techniques like authentication and cryptography, methods to handle cyber threats, and various attacks like denial of service, impersonation attack, man in the middle attack. We have also discussed based privacy preserving protocols based on RDT for data collection that does not rely on a private channel assumption.

Based on the various techniques analyzed in the paper, data perturbation technique used in [5] achieves most efficiency when used for privacy preservation in data mining.

References

1. Shankar A, Pandiaraja P, Sumathi K et al (2021) Privacy preserving E-voting cloud system based on ID based encryption. *Peer-to-Peer Netw Appl* 14:2399–2409. <https://doi.org/10.1007/s12083-020-00977-4>; Cunha M, Mendes R, Vilela JP (2021) A survey of privacy-preserving mechanisms for heterogeneous data types. *Comput Sci Rev* 41:100403. ISSN 1574-0137. <https://doi.org/10.1016/j.cosrev.2021.100403>
2. Leung CK, Hoi CSH, Pazdor AGM, Wodi BH, Cuzzocrea A (2018) Privacy-preserving frequent pattern mining from big uncertain data. In: 2018 IEEE international conference on big data (big data), pp 5101–5110. <https://doi.org/10.1109/BigData.2018.8622260>
3. Zhou S, Zhang R, Feng J, Chen D, Chen L (2018) A novel method for mining abnormal behaviors in social medical insurance. In: 2018 IEEE 9th annual information technology, electronics and mobile communication conference (IEMCON), pp 744–748. <https://doi.org/10.1109/IEMCON.2018.8614806>
4. Kousika, Premalatha K (2021) An improved privacy-preserving data mining technique using singular value decomposition with three-dimensional rotation data perturbation. *J Supercomput* 77:1–9. <https://doi.org/10.1007/s11227-021-03643-5>
5. Kaliappan S (2018) A hybrid clustering approach and random rotation perturbation (RRP) for privacy preserving data mining. *Int J Intell Eng Syst* 11:167–176. <https://doi.org/10.22266/ijies2018.1231.17>
6. Kaur A (2017) A hybrid approach of privacy preserving data mining using suppression and perturbation techniques. In: 2017 international conference on innovative mechanisms for industry applications (ICIMIA), pp 306–311. <https://doi.org/10.1109/ICIMIA.2017.7975625>
7. Siddhpura A, Vekariya V (2018) An approach of privacy preserving data mining using perturbation & cryptography technique. *Int J Future Revol Comput Sci Commun Eng* 4:255–259. ISSN: 2454-4248
8. Wu N, Peng C, Niu K (2020) A privacy-preserving game model for local differential privacy by using information-theoretic approach. *IEEE Access* 8:216741–216751. <https://doi.org/10.1109/ACCESS.2020.3041854>

9. Beg S, Anjum A, Ahmad M, Hussain S, Ahmad G, Khan S, Choo K-KR (2021) A privacy-preserving protocol for continuous and dynamic data collection in IoT enabled mobile app recommendation system (MARS). *J Netw Comput Appl* 174:102874. ISSN 1084-8045. <https://doi.org/10.1016/j.jnca.2020.102874>
10. Zhou Y, Tian Y, Liu F, Liu J, Zhu Y (2019) Privacy preserving distributed data mining based on secure multi-party computation. In: 2019 IEEE 11th international conference on advanced infocomm technology (ICAIT), pp 173–178. <https://doi.org/10.1109/ICAIT.2019.8935900>
11. Domadiya N, Rao UP (2021) Improving healthcare services using source anonymous scheme with privacy preserving distributed healthcare data collection and mining. *Computing* 103:1–23. <https://doi.org/10.1007/s00607-020-00847-0>
12. Upadhyay S, Sharma C, Sharma P, Bharadwaj P, Seeja KR (2018) Privacy preserving data mining with 3-D rotation transformation. *J King Saud Univ Comput Inf Sci* 30(4):524–530. <https://doi.org/10.1016/j.jksuci.2016.11.009>
13. Keshk M, Moustafa N, Sitnikova E, Turnbull B, Vatsalan D (2020) Privacy-preserving techniques for protecting large-scale data of cyber-physical systems. In: 2020 16th international conference on mobility, sensing and networking (MSN), pp 711–717. <https://doi.org/10.1109/MSN50589.2020.00121>
14. Weng J, Weng J, Zhang J, Li M, Zhang Y, Luo W (2021) DeepChain: auditable and privacy-preserving deep learning with blockchain-based incentive. *IEEE Trans Dependable Secure Comput* 18(5):2438–2455. <https://doi.org/10.1109/TDSC.2019.2952332>
15. Sheshasayee A, Thomas SS (2017) Implementation of data mining techniques in upcoding fraud detection in the monetary domains. In: 2017 international conference on innovative mechanisms for industry applications (ICIMIA), pp 730–734. <https://doi.org/10.1109/ICIMIA.2017.7975561>
16. Sharma S, Shukla D (2016) Efficient multi-party privacy preserving data mining for vertically partitioned data. In: 2016 international conference on inventive computation technologies (ICICT), pp 1–7. <https://doi.org/10.1109/INVENTIVE.2016.7824852>

Controller Area Network (CAN)-Based Automatic Fog Light and Wiper Controller Prototype for Automobiles



Sowmya Madhavan, Supriya Kalmath, R. Ramya Rao, Shreya P. Patil, and M. D. Tejaswini

1 Introduction

Recent automobile technology is always tending toward automation. Present day automobiles perform wide variety of functions. They are very sophisticated, with a lot of safety features, passenger comfort, reduction in fuel consumption, pollution management, in built entertainment systems, etc. High level of automation is incorporated into the vehicle so that human intervention is minimized or nil [1].

Most automobiles have a system called as the Electronic Control Unit (ECU). The ECU can be called as the main controller of the engine management system. An automobile can have a single ECU or multiple ECUs. ECU can also be called as the embedded system part of the automobile. ECUs can be Anti-braking System, Battery Management System, in vehicle infotainment system, fuel injection system, etc. It also provides hardware and software support for future functionality expansions [2].

Many ECUs in an automobile system can actually make the network complex and the communication between different ECUs should be seamless for efficient automobile control and management. The need for a highly coherent inter-ECU communication led to the development of the CAN protocol. Earlier, protocols like Local Interconnect Network (LIN) and FlexRay were used in automobiles [3]. But the main advantages of CAN over these include higher data rate, less expensive and error detection. Also, complicated wiring among ECUs was eliminated because of CAN protocol. Any CAN node should have the following components for transmitting and receiving messages.

Host Microcontroller Unit (MCU) is the core microcontroller which executes various functions according to its program. In our project, the MCU is the Arduino which has a cortex M3 processor [4].

S. Madhavan (✉) · S. Kalmath · R. Ramya Rao · S. P. Patil · M. D. Tejaswini
Nitte Meenakshi Institute of Technology, Bangalore, India
e-mail: sowmya.madhavan@nmit.ac.in

CAN controller will be a part of the microcontroller unit, or it may be connected as a separate unit. The function of the CAN controller is to convert the data according to the CAN protocol. CAN transceiver converts the data according to the physical voltage levels required by the system and then transmits or receives the data [5].

In our project, the ECU designed is automatic fog light and wiper control system. In this ECU, 2 Arduino controllers are used. One Arduino controller is the slave controller to which sensors are interfaced. Second Arduino is the master controller which can DC motor for wiper movement and switch ON or switch OFF fog lights. Exact sensor details and the block diagram of the prototype are discussed in further sections.

2 Literature Survey

The literature survey provides a bird's eye view about the research done in this area so far.

Networks built using CAN and traditional vehicle circuit connections are compared. It is proved that CAN has drastically changed the electrical wiring system of the vehicle which has a direct impact in reducing component costs and increasing the reliability [6].

Any general embedded system can employ a CAN bus. The CAN protocol defines both hardware and software standards. The main advantage of designing with CAN is that it is very easily implementable, more like a plug and play fashion. It also means that the designers need not understand every bit of the CAN protocol, but they can use the protocol to build an application by knowing the hardware interfaces and output values [7].

Single Board Computers (SBC) are usually employed to implement the CAN bus protocol. Raspberry Pi can also be used for CAN implementation. The Raspberry Pi is also equipped with a high performance processor which also helps in faster data transmission and processing [8].

Semi-autonomous vehicles are built with an analog driver vehicle interface. The analog interface can be converted into a digital interface. The design has a data acquisition system in which the Analog to Digital Converter (ADC) converts all data into digital for. The digital data will be sent to the display. Low speed CAN and high speed CAN were used according to the requirements [9].

Traditionally, CAN is a wired protocol. But as the world is embracing artificial intelligence-based systems, there is an ever increasing need to control automobile systems remotely. Wireless CAN and IoT can merge for realizing a remotely controlled automobiles [10].

There is an ever increasing need to develop more safer and reliable systems in automobiles, along with other entertainment technologies. Automatic control of the speed of the windshield also comes under increasing the safety of the automobile. There are various methods to start the wiper motion and adjust its speed automatically [11].

Existing systems might have some limitations in their functionality. One such problem is the blurriness for the driver because of high intensity head lights coming from the opposite direction. Also, the windshield haze is a threat because everything in front of the driver appears smudged and cause accidents. Fog light automation is one of the main solutions for this, which doesn't exist in economy cars [12].

Increased safety means there would be a lot of parameters to monitor like temperature, gas leakage, glaring effect, current sensor, fuel leakage, etc. All sensors can be interfaced to Single Board Computers and to both master and slave ECUs, if there are two. CAN efficiently supports interface between different ECUs which monitor all these parameters [13].

In many cases, high intensity headlights are problematic which can cause lot of glare to drivers coming in the opposite direction. Automobile drivers might not decrease the intensity of their head lights according to the lighting on the road or when other vehicles pass beside them. Automatic variation in the intensity of head lights can be benefitting. This can be accomplished using an accelerometer sensor. It has already been tested in the lab with 97% success rate [14].

In a modern-day automobile, there exists a complex interconnection of sensors, actuators, controllers, and physical cables along with other electronic or electrical equipments. The widely used protocol for this connection is the CAN protocol which is wired, having differential voltages on the physical cable. However, a new hybrid communication protocol termed as vehicular wireless CAN (ViCAN) is introduced. This is called "hybrid" since it combines both wired and wireless versions of the CAN protocol. The major advantage of this hybrid architecture is that it reduces the overall complexity of the system and supports reliable node to node communication [15].

3 Hardware and Software Requirements

The description of the sensors used, CAN Frame format, CAN Transceiver, CAN controller and the software used to develop the application are briefly described here.

3.1 Hardware Requirements

Light Dependent Resistor (LDR) is a sensor which is nothing but a photoresistor which can detect the presence of the light and the intensity of the light. The photoresistor's resistance decreases with increasing light intensity. LDRs find utility in light weight detector circuits. A track is made out of an exposed semi-conductor like Cadmium Sulfide and is mounted on a base leads are taken out of it. A LDR is as shown in Fig. 1. Here LDRIB7206 is used.

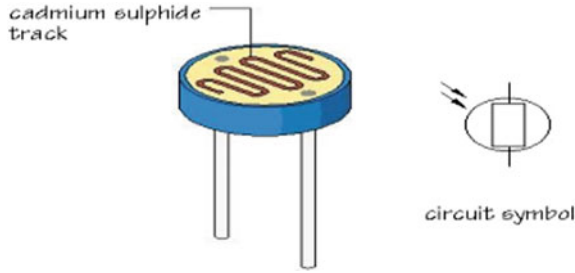


Fig. 1 Light dependent resistor

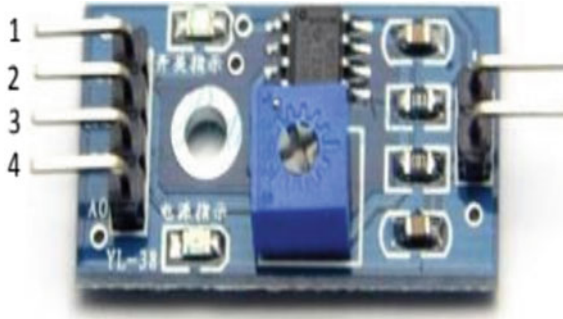


Fig. 2 Rain detection sensor

Rain Detection Sensor SKU890228 actually works like a switch. If rain drops are detected, the switch will be closed. Nickel-coated lines are mounted on the board in the form of fringes. The resistance of the sensor decreases when rain drops fall on it and increases whenever the sensor ID dry. The Rain Detection Sensor is as shown in Fig. 2.

CAN version 2.0B is implemented by the CAN controller MCP 2515. The data is converted according to the CAN data frame format. The CAN controller will be interfaced with the microcontroller via the Serial Peripheral Interface (SPI) protocol. MCP 2551 is the CAN Transceiver, which converts data into physical voltage levels. Differential physical voltages are employed in the range of 2.5–3.5 V. This is placed between the CAN controller and the physical bus. Figures 3 and 4 show CAN controller and CAN Transceiver, respectively.

The tool used to develop this application is the Arduino IDE.

The CAN protocol defines a standard format of data which will be transmitted on the CAN physical bus. CAN data format is as shown in Fig. 5.

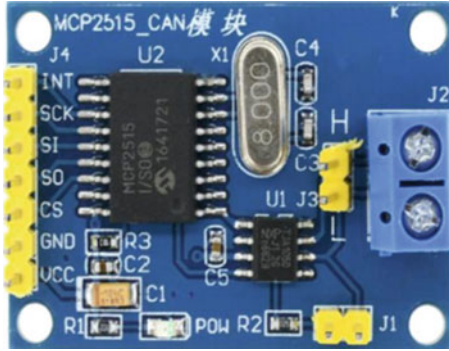


Fig. 3 CAN controller 2515

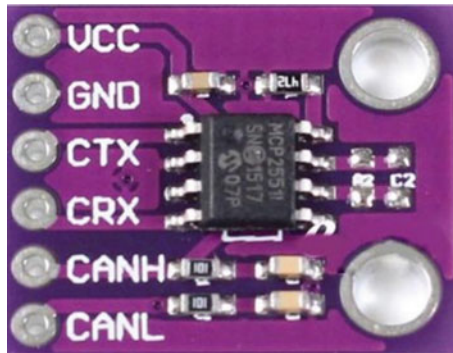
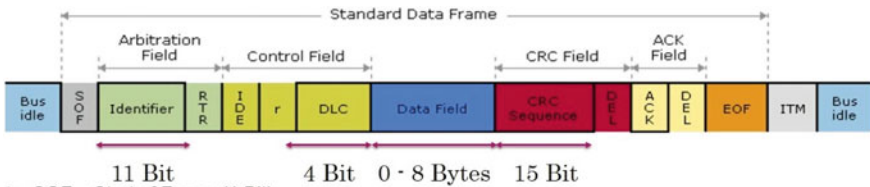


Fig. 4 CAN transceiver 2551



- SOF – Start of Frame (1 Bit)
- Identifier – A message identifier sets the priority of the data frame.
- RTR – Remote Transmission Request, defines the frame type (data frame or remote frame) (1 Bit).
- Control Field – User defined Functions.
- DLC – Data Length Code (4 Bits).
- Data Field – User defined Data (0 to 8 Bytes)
- CRC Field – cyclic redundancy check for Error (Data corruption) detection.
- ACK Field – Receivers Acknowledgement.
- EOF – End of Frame (7 Bits)

Fig. 5 CAN transceiver 2551

4 Design and Implementation

The block diagram of the prototype is as shown in Fig. 6.

The system consists of two Arduino processors or two Electronic Control Units (ECUs). Initial processor which is the slave processor takes the input from the LDR and the Rain Detection Sensor. LDR senses for presence of sunlight, and Rain Detection Sensor checks for drops of rainfall. The slave processor sends this information to the second processor which is the master processor over the CAN bus. The master processor takes call whether to turn fog lights ON or OFF and whether to start the wiper or not. This information is sent toward the initial processor. The vehicle servo motors can tilt so that the light beam can bend in the upward or downward direction. Digital Temperature and Humidity sensor can also be optionally interfaced to the slave ECU.

The design and implementation of wiper control is as shown in the flowchart in Fig. 7.

The design and implementation of fog light control is as shown in the flowchart in Fig. 8.

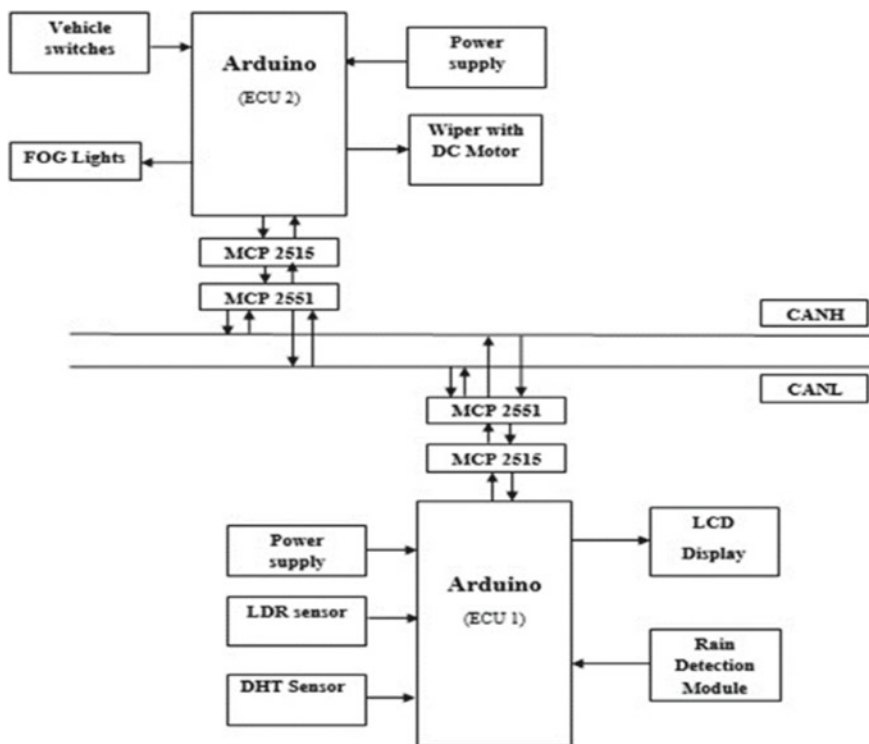


Fig. 6 Prototype block diagram

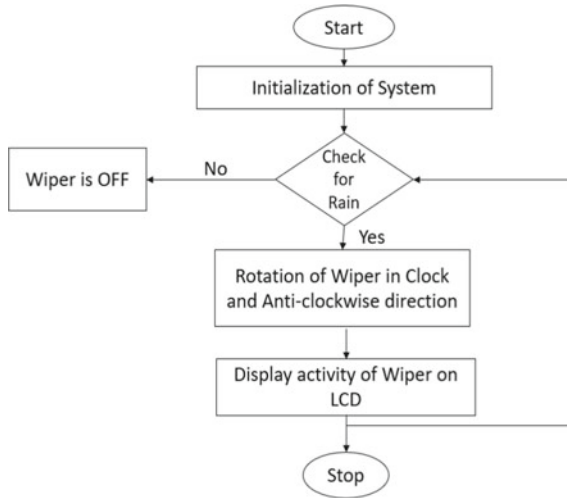


Fig. 7 Flowchart for wiper control

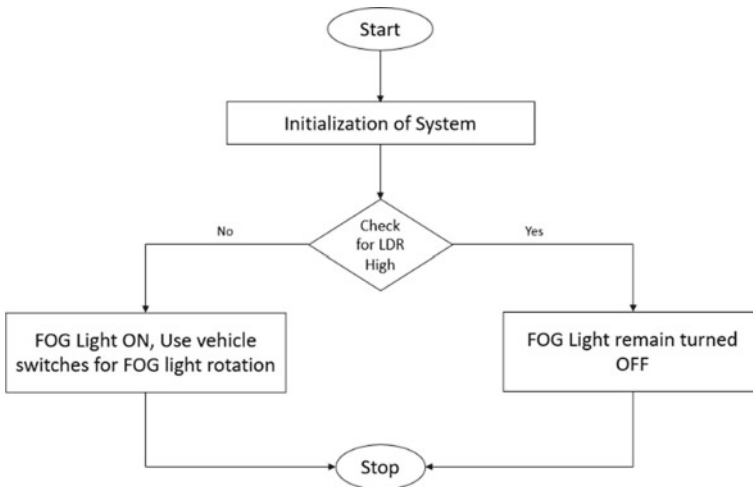


Fig. 8 Flowchart for wiper control

5 Results and Discussions

A prototype of the automatic fog light and wiper control system is as shown in Fig. 9.

Master and slave ECUs communicate with each other to perform specific functions, namely displaying temperature and humidity values, turning ON or OFF a DC motor which controls the wiper and turning ON or OFF Light Emitting Diodes (LED) lights, corresponding to fog lights of the automobile. It is battery powered.

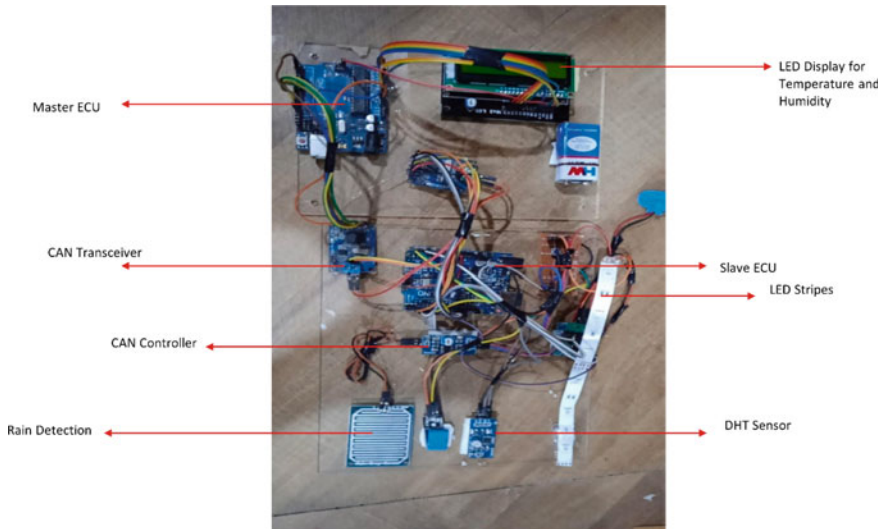


Fig. 9 Flowchart for wiper control

6 Conclusion and Future Scope

Automation in automobiles is today's need as well as luxury requirement also. With this motive, authors have developed a prototype of automatic fog light and wiper controller using the CAN protocol. CAN protocol is the most widely used protocol in automobiles for communication in between ECUs. The prototype is successful in implementing automation for wiper and fog light control. The prototype can be used further for development of the product.

References

1. https://www.streetdirectory.com/travel_guide/56847/cars/the_modern_day_car_a_sophisticated_high_tech_gadget.html
2. <https://www.bosch-mobility-solutions.com/en/solutions/control-units/engine-control-unit/>
3. <https://www.cselectricalandelectronics.com/difference-between-lin-can-most-flexray/>
4. <https://www.embitel.com/blog/embedded-blog/what-is-can-protocol-stack-why-its-critical-software-solution-for-ecu-communication>
5. https://www.infineon.com/cms/en/product/transceivers/automotive-transceiver/automotive-can-transceivers/?gclid=CjwKCAiAtouOBhA6EiwA2nLKH-206vOZ05EGfiPZmFBDQ_J59OQW0ovExNxIgxwbxgVm7QAZzrj1jBoC9oEQAvD_BwE&gclsrc=aw.ds
6. Guo S (2011) The application of CAN-bus technology in the vehicle. In: 2011 international conference on mechatronic science, electric engineering and computer (MEC), pp 755–758. <https://doi.org/10.1109/MEC.2011.6025574>
7. Li X, Li M (2010) An embedded CAN-BUS communication module for measurement and control system. <https://doi.org/10.1109/ICEEE.2010.5661248>

8. Salunkhe AA, Kamble PP, Jadhav R (2016) Design and implementation of CAN bus protocol for monitoring vehicle parameters. In: 2016 IEEE international conference on recent trends in electronics, information & communication technology (RTEICT), pp 301–304. <https://doi.org/10.1109/RTEICT.2016.7807831>
9. Vijayalakshmi S (2013) Vehicle control system implementation using CAN protocol. *Int J Adv Res Electr Electron Instrum Eng* 2(6). ISSN (Print): 2320-3765, ISSN (Online): 2278-8875
10. Chikhale SN (2018) Automobile design and implementation of CAN bus protocol—a review. *IJRDO J Electr Electron Eng* 4(1):01–05. ISSN: 2456-6055. Retrieved from <http://www.ijrdo.org/index.php/eee/article/view/1528>
11. Naresh P, Haribabu AV (2015) Automatic rain-sensing wiper system for 4-wheeler vehicles. *J Adv Eng Technol* 3:1–5
12. Balaji RD (2020) A case study on automatic smart headlight system for accident avoidance. *IJCCI* 2(1):70–77
13. Wagh PA, Pawar RR, Nalbalwar SL (2017) A review on automotive safety system using can protocol. *Int J Curr Eng Sci Res (IJCESR)* 4(3). ISSN (PRINT): 2393-8374, (ONLINE): 2394-0697
14. Muhammad F, Dwi Yanto D, Martiningsih W, Noverli V, Wiryadinata R (2020) Design of automatic headlight system based on road contour and beam from other headlights. In: 2020 2nd international conference on industrial electrical and electronics (ICIEE), pp 112–115. <https://doi.org/10.1109/ICIEE49813.2020.9276906>
15. Laifenfeld M, Philosof T (2014) Wireless controller area network for in-vehicle communication. In: 2014 IEEE 28th convention of electrical and electronics engineers in Israel, IEEEI 2014. <https://doi.org/10.1109/EEEI.2014.7005751>

Multivariate Long-Term Forecasting of T1DM: A Hybrid Econometric Model-Based Approach



Rekha Phadke and H. C. Nagaraj

1 Introduction

Type 1 diabetes mellitus (T1DM) patient's glucose variations are highly influenced by multitude of parameters, namely insulin dosage, diet, lifestyle, sleep quality, stress, etc. This demands that an accurate blood glucose prediction model should be based on multivariate analysis approach. A few univariate-based blood glucose prediction algorithms have been studied in literature [1–5]. These univariate time series algorithms predict the blood glucose values by only considering the blood glucose history. The univariate models were reliable, simple and fast; they are more suitable for short-term prediction [6]. As univariate model is less comprehensive, it is more judicious to use a multivariate model for long-term prediction of blood glucose values [7]. In this paper, the author implements econometric model-based approach, where multivariate time series algorithms are used for long-term blood glucose forecasting.

A survey of various machine learning algorithms involving multiple variables is as discussed.

Jensen et al. [8] have used machine learning approach, namely autoregressive integrated moving average (ARIMA) and support vector regression (SVR) model for hypoglycemia prediction. They claim that SVR model has outperformed clinical diagnosis in predicting 23% of hypoglycemic events 30 min in advance.

Marling et al. [9] divided the available dataset of patients into training and testing data. Initial 7 days data was used to train the model and last 3 days data was used to test. Moving average (MA), simple exponential smoothing (SES) and support vector machine (SVM) techniques were used. SVM outperformed by obtaining RMSE of

R. Phadke (✉) · H. C. Nagaraj
Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of
Technology, Bangalore, India
e-mail: rekhapadke@gmail.com

18.0 mg/dL for 30 min prediction horizon (PH) and RMSE of 30.9 mg/dL for 60 min PH. Neural network-based approach for predicting glucose level is employed in [10–14].

Eren-Orukulu et al. [15] have collected databases of 2 patients under hospitalized and normal life conditions measured by CGM sensor. The time series model employed both recursive identification and change detection methods to adapt to variability and glycemic disturbances dynamically among inter/intra patients. Prediction performances were evaluated based on glucose prediction error and Clark's error grid analysis (C-EGA). Analysis using C-EGA resulted in accurate readings of 90% or more.

Petridis et al. [16] built a probabilistic time series predictor using Bayesian combined predictor (BCP). The BCP outperformed the conventional predictors.

In the next section, dataset being used in this study is discussed.

2 Librepro and Ohio T1DM Dataset

Long-term prediction time series algorithms are implemented in this study, on two different datasets, namely:

- Dataset A: Librepro CGM sensor dataset (Courtesy: Jnana Sanjeevini Diabetes Hospital and Medical Center, Bangalore).
- Dataset B: Ohio T1DM CGM sensor dataset (Courtesy: Ohio University. An DUA was signed by NMIT, Bangalore, and Ohio University for using the dataset strictly for academic research).

2.1 Dataset A: Librepro CGM Sensor Dataset

Abbott's Freestyle LibrePro is a continuous glucose monitoring (CGM) system. The blood glucose data is collected automatically, once in every 15 min. Figure 1 shows the picture of the sensor.

Dataset A, i.e., LIBREPRO CGM sensor dataset consists of 10 Type I diabetic individual's data recorded for 10 days. The blood glucose (BG) level is recorded at each 15 min interval; hence, there are a total of 960 reading per patient leading to 9600 readings for the entire dataset. These readings are verified and taken from "Jnana Sanjeevini Diabetes Hospital and Medical Center," Bangalore.

In [7], the author has implemented various univariate time series algorithms on T1DM Librepro CGM sensor dataset. The results are recorded in Table 1.

The univariate algorithms have obtained good results for prediction horizon up to 45 min. But for long-term prediction, these algorithms were found to be less reliable. The MAPE obtained for 1-day prediction in ARIMA model was 63.45. Hence, it was unsuitable for multivariate analysis, as it contained only one independent variable.

Fig. 1 Snapshot of Abbott’s FreeStyle LibrePro CGM sensor with reader [17]



Table 1 Performance statistics of univariate algorithms on T1DM Librepro CGM sensor dataset A; for a prediction horizon of 15 min [7]

Algorithm	RMSE	MAE	MAPE
Moving average	18.09	13.86	10.76
Linear regression	35.52	27.84	25.68
ARIMA	7.07	5.12	3.98
Novel ensemble method	7.38	5.47	3.22
Holts AAN	7.98	5.91	4.57
Holts MMN	8.504	6.11	4.65

Hence, the author decided to implement multivariate long-term prediction algorithms on dataset B.

2.2 Dataset B: Ohio T1DM CGM Sensor Dataset

The Ohio T1DM dataset consists of type 1, six patients eight weeks’ data. The patients age group was in the range 40–60 years. Out of which two were male, and four were female. All the patients used insulin pump therapy (Medtronic 530G) with continuous glucose monitoring (CGM) (Medtronic Enlite). A custom smartphone is used to record the hyper/hypo glycemic data. Basis peak fitness band was used to report physiological data.

The dataset consisted of CGM blood glucose level recorded at 5 min intervals, self-monitored finger stick-based method of blood glucose levels, insulin dosage, meal intake with an estimate of carbohydrate rating (self-reported), exercise time (self-reported), quality and duration of sleep, work and stress, illness and every 5 min

Table 2 Training and test dataset per contributor

Contributor	Training dataset	Test dataset
559	10,769	2514
563	12,124	2570
570	10,982	2745
575	11,866	2590
588	12,640	2791
591	10,847	2760

recording of heart rate, galvanic skin response (GSR), skin and air temperatures and step count.

2.2.1 Ohio T1DM Data Format

Each contributor had 2 XML file, one for training data and the other for test data which are tabulated with respect to their ID numbers (Table 2).

Each XML file contained the following data field (Table 3).

2.2.2 The Ohio T1DM Viewer

The Ohio T1DM viewer [18] is a visualizer tool that graphically displays XML file of the Ohio T1DM dataset. The bottom panel shows the CGM data, insulin and self-reported life events. The panel at the top displays basis peak fitness band data. Snapshot of Ohio T1DM viewer is as shown in Fig. 2.

The Ohio T1DM dataset was originally available to only the participants of the 3rd international workshop IJCAI-ECAI 2018. Later this dataset was made available to all the researchers in the field of health care with the obtainability of a non-disclosure data user agreement (DUA).

3 Dataset B Preparation

The dataset preparation is carried out in three different steps, namely:

- Data sampling and interpolation

- Data conversion

- Data segregation (time-flag methodology) and classification.

The data preparation block diagram is as shown in Fig. 3.

Table 3 XML data fields with 19 different attributes per contributor [18]

Data field	Details
Patient	Patient ID number and the insulin type
Glucose level	CGM data for every 5 min interval
Finger stick	Finger stick-based BG values
Basal	Basal insulin infusion rate
temp_basal	Temporary basal insulin rate
Bolus	Insulin rate delivered before a meal or during hyperglycemia
Meal	Patient’s meal timing and carbohydrate rating
Sleep	Patient’s sleep quality scaled from 1 to 3 with 3 as good
Work	Patient’s physical exercise and exertion rating scaled from 1 to 10 with 10 as most active
Stressors	Time of stressful events
Hypo-event	Time of hypoglycemic event
Illness	Time of sudden illness
Exercise	Exercise intensity scaled from 1 to 10 along with duration in minutes
basis_heart_rate	Heart rate collected for a period of 5 min interval
basis_gsr	Galvanic skin response collected for a period of 5 min interval
basis_skin_temperature	Skin temperature, in degrees Fahrenheit, collected for a period of 5 min interval
basis_air_temperature	Air temperature, in degrees Fahrenheit, collected for a period of 5 min interval
basis_steps	Step count is collected for a period of 5 min interval
basis_sleep	Basis band reports when the subject is asleep, along with the sleep quality estimation

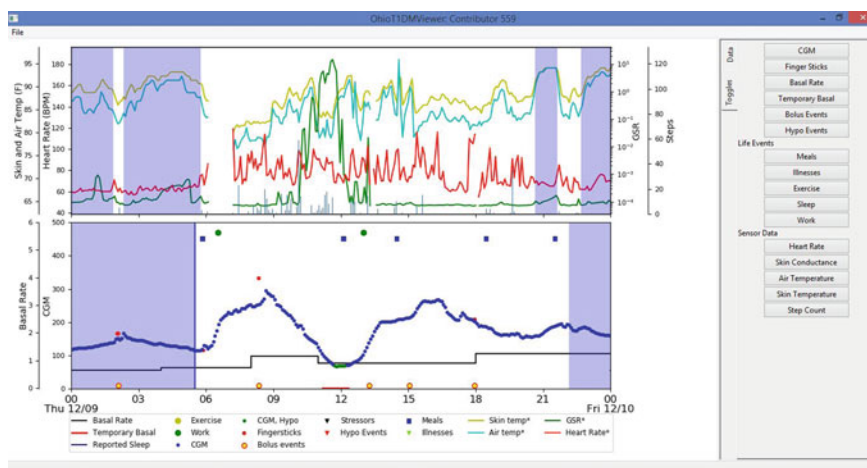


Fig. 2 Snapshot of Ohio T1DM viewer for day-wise display of integrated data [18]

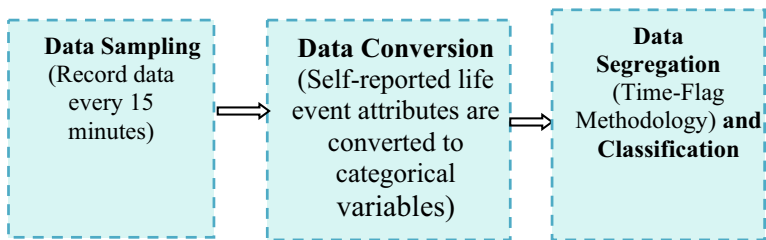


Fig. 3 Block diagram of dataset B preparation

3.1 Data Sampling and Interpolation

Ohio T1DM dataset consisted of every 5 min reading of glucose level, basis_heart_rate, basis_gsr, basis_steps, basis_skin_temperature and basis_air_temperature. But these had many instances of missing data value. Hence, the dataset was sampled to record every 15 min interval data to reduce the percentage of missing data. The quantity of data before and after sampling is as shown in Fig. 4.

The dataset from Ohio University had occurrences of data loss due to missing data. Hence, data had to be interpolated using standard techniques available such as:

- Occurrence of data loss on Day 1: The recent data was duplicated.
- Occurrence of data loss on Day 2 and onwards: Previous day corresponding time stamped glucose value was replicated.
- Consecutive Missing Data: Employed moving average algorithm to account for data loss.

Fig. 4 Data summary after sampling data for every 15 min interval

Data Sampling

Contributor	Train Examples	Test Examples
559	10796	2514
563	12124	2570
570	10982	2745
575	11866	2590
588	12640	2791
591	10847	2760

Sample every 15 minutes

Contributor	Train Examples	Test Examples
559	3980	960
563	3953	897
570	3799	961
575	4320	905
588	4320	949
591	4123	948

3.2 Data Conversion

The various attributes, namely sleep quality, basal_insulin, temp_basal, bolus_insulin, carb, meal, work intensity, stress, illness, exercise, hypo-events from Ohio T1DM dataset, which are self-reported life event attributes, were considered as categorical variables. The category of the parameters as per Ohio T1DM dataset is as given in Table 4.

All these self-reported categorical variables except meal are considered as dummy variables. These dummy variables are converted to a matrix format as shown in Fig. 5.

Table 4 Data category of self-reported variables in Ohio T1DM dataset [18]

S. No.	Variable		Category
1.	Sleep_quality	No	0
		Poor	1
		Fair	2
		Good	3
2.	Tmp_Basal	Insulin given	1
		Insulin not given	0
3.	Work intensity	No work	0
		Less activity	1
		Moderate activity	4
		Good activity	8
		Highly active	9
4.	Exercise	No work	0
		Less activity	1
		Moderate activity	2
		Good activity	8
		Highly active	9
5.	Stress	No stress	0
		Stressed	1
6.	Illness	No illnesses	0
		Illness present	1
7.	Hypo_events	No hypo_event	0
		hypo_event present	1
8.	Meal	Snack	Carbohydrate count as reported by individuals; can vary from 0 to 100
		Breakfast	
		Lunch	
		Dinner	
		Hypo-correction	
		No meal	

Matrix Format of Dummy Variables

Example: Sleep Quality

Patient ID: 559	No 0	Poor 1	Fair 2	Good 3
Sleep quality 0	1	0	0	0
Sleep quality 2	0	0	1	0
Sleep quality 1	0	1	0	0
Sleep quality 3	0	0	0	1
Sleep quality 2	0	0	1	0

Depending on the self-reported sleep quality by the patient, all the 96 readings of a particular day will be filled with the row readings.
 For example, if sleep quality reported by the patient on a particular day is Fair the conversion process is as shown.

Sample	Time Stamp	No	Poor	Fair	Good
1	12/7/2021 21:29	0	0	1	0
2	12/7/2021 21:44	0	0	1	0
3	12/7/2021 21:59	0	0	1	0
4	12/7/2021 22:14	0	0	1	0
5	12/7/2021 22:44	0	0	1	0

Fig. 5 Data conversion of dummy variables to matrix format

3.3 Data Segregation (Time-Flag Methodology)

This involves

- Splitting the 24 h time zone into 7 time-flag buckets. Followed by realigning the data as per the bucket.
- Classification of independent data variables into physiological and psychological data.

Table 5 Time-flag landmarks for time slots

Time-flag bucket		
t6	0–6 h	Flag-1,0 (if data falls in t6 flag will be high, else it will be zero)
t10	6–10	Flag-1,0 (if data falls in t10 flag will be high, else it will be zero)
t13	10–13	Flag-1,0 (if data falls in t13 flag will be high, else it will be zero)
t16	13–16	Flag-1,0 (if data falls in t16 flag will be high, else it will be zero)
t18	16–18	Flag-1,0 (if data falls in t18 flag will be high, else it will be zero)
t21	18–21	Flag-1,0 (if data falls in t21 flag will be high, else it will be zero)
t24	21–24	Flag-1,0 (if data falls in t24 flag will be high, else it will be zero)

3.3.1 Time-Flag Bucket Creation

In this step, the 24 h time slots are divided into 7 different time-flag buckets as tabulated in Table 5.

This method is advantageous as it helps in validation process. During the validation process, begin and end time slot of the test window is checked, and it is mapped with an appropriate time-flag bucket. Forecasting is done considering the previous history of blood glucose values from that corresponding time-flag bucket of the training dataset. This helps in more reliable and accurate prediction of blood glucose value. For example, to predict the blood sugar level at 8:00 a.m., the previous day same time slots blood sugar level will also be considered by the forecasting algorithms.

3.3.2 Data Classification into Psychological and Physiological Variables

Artificial intelligence (AI) models until now have considered only the quantifiable physiological parameters for diabetes monitoring and prediction. The human facet, i.e., the patient’s psychological data is the least being considered by the AI, while they can actually play a very important role in improving the prevention and care of diabetes. Ignorance of this aspect will lead to erroneous treatment being dispensed to a patient [19]. Hence, in this paper, author considers both physiological and psychological data of a patient.

Here, the 19 different attributes in the dataset are classified into physiological and psychological variable. The classification is listed in Fig. 6.

4 Analytical Model

The model is now built using the segregated variables as prepared in the above section and is trained using multivariate machine learning algorithms. The analytic approach used by the model for blood glucose prediction is as shown in Fig. 7.

Fig. 6 Classification of the independent data variables

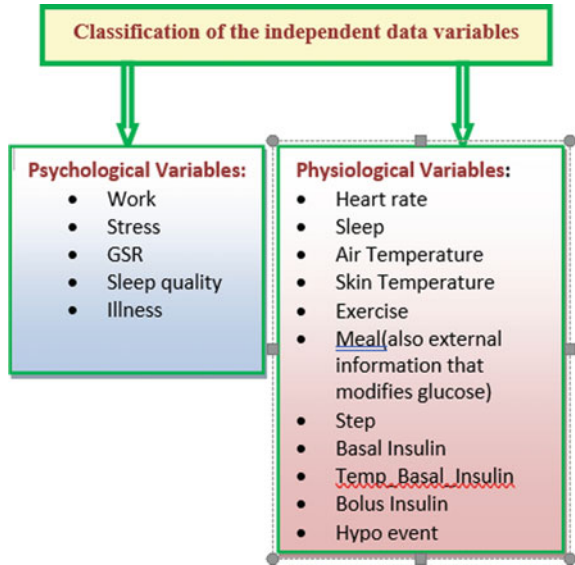
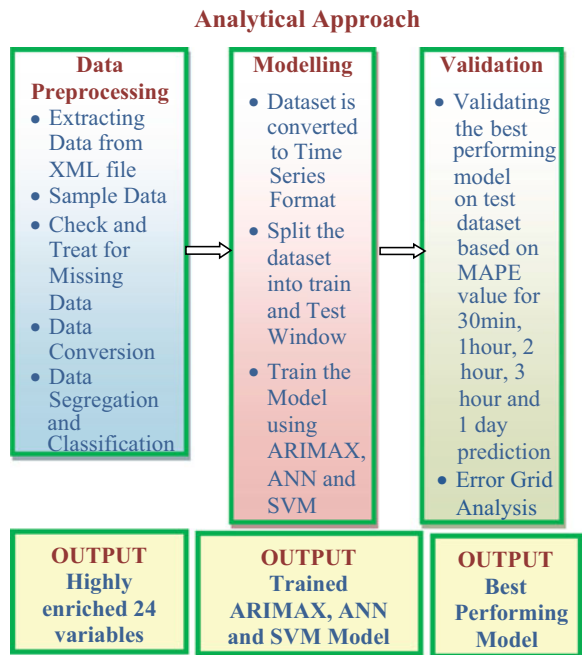


Fig. 7 Analytical model for glucose prediction



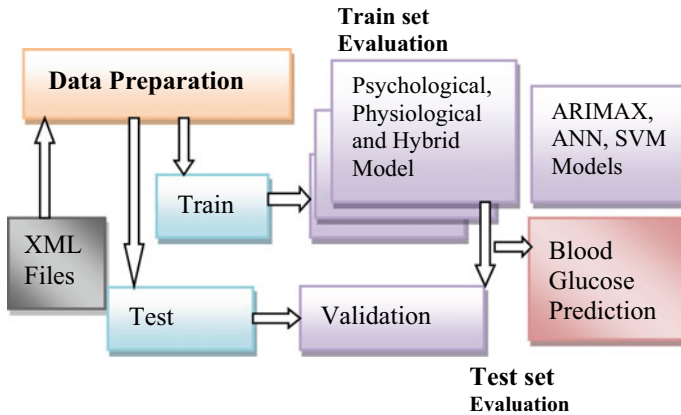


Fig. 8 Analytical model process flow

Three types of analytical model are built for glucose prediction, namely:

- **Psychological Model:** This model predicts the blood glucose value based on time-based BG values and psychological variables.
- **Physiological Model:** This model predicts the blood glucose value based on time-based BG values and physiological variables.
- **Hybrid Model:** This model predicts the blood glucose value based on time-based BG values and a combination of physiological and psychological parameters.
- The process flow of the model implementation using machine learning algorithms is as depicted in the block diagram shown in Fig. 8.

The multivariate time series machine learning models are trained on the three types of analytical models discussed above and is then validated for performance using performance metrics. 80:20 ratio is used for dataset division into train and test sets, respectively. The performance metrics used for evaluation of the multivariate models are:

- Mean absolute percentage error (MAPE)
- Clarks error grid analysis (EGA).

4.1 Multivariate Time Series Models

Blood glucose data is often affected by insulin reaction, physical activity, carbohydrate intake, stress and similar events, which we can refer to as involvement of multiple events. Hence, there is a requisite to analyze multiple parameters for BG prediction.

The multivariate time series models employed in this paper are:

- Autoregressive integrated moving average with explanatory variable (ARIMAX)

- Artificial neural networks (ANN)
- Support vector machines (SVM).

Prior to application of ARIMAX, ANN or SVM Model, we need to convert the dataset to stationary. This involves the following steps:

- **Step 1:** Obtain the time series plot of the dataset (Fig. 9).
- **Step 2:** Determine whether or not the dependent variable (blood glucose level) is stationary using auto correlation function (ACF) plot.
The ACF plot in Fig. 10 is exponentially decaying. However, it is above the significance range (dotted blue line). This indicates a non-stationary data series.
- **Step 3:** If found to be non-stationary, differentiate the dependent variable, until ACF plot indicates it to be stationary.

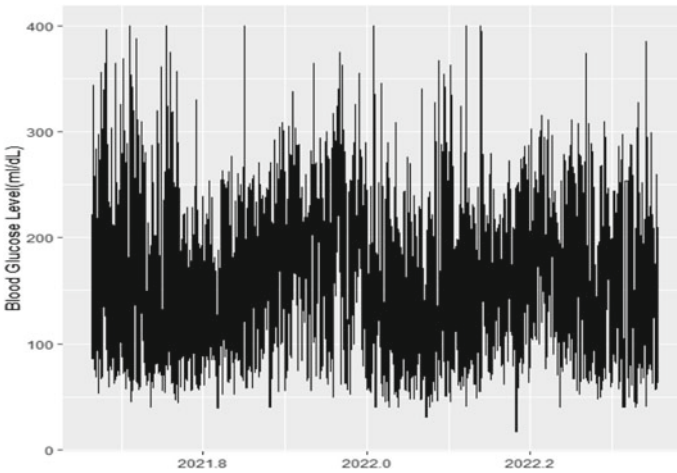


Fig. 9 Time series plot for blood glucose level

Fig. 10 ACF plot of blood glucose level

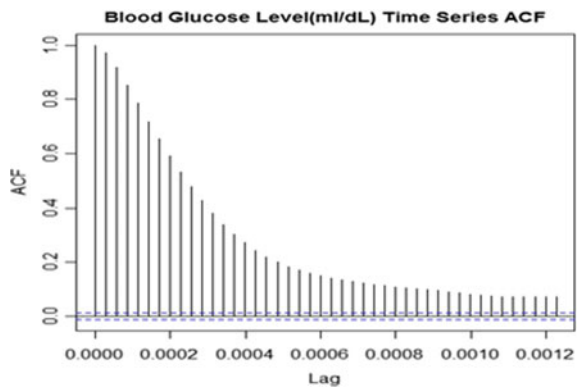
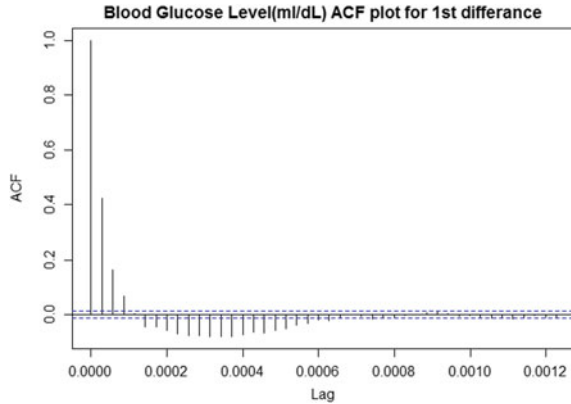


Fig. 11 ACF plot of blood glucose level post differentiation



The ACF plot in Fig. 11 is quickly decaying exponentially. And it goes below the significance range (dotted blue line). This indicates a stationary data series. Hence post differentiation, the dependent variable is made stationary. In the next section, deployment of the multivariate time series models on the stationary dataset is discussed.

4.1.1 ARIMAX

Autoregressive integrated moving average with explanatory variable (ARIMAX) model is a multiple regression model. ARIMA model including multiple input variables is commonly mentioned as ARIMAX model. Pankratz [20] states ARIMAX model as dynamic regression. This method is suitable for forecasting multivariate data irrespective of its data pattern.

The ARIMA model is expressed as:

$$\hat{y}_t = \mu + \varphi_1 y_{t-1} + \dots + \varphi_p y_{t-p} - \theta_1 \epsilon_{t-1} \dots - \theta_q \epsilon_{t-q} \tag{1}$$

where μ is constant term, φ_k is the AR coefficient at lag k , θ_k is the MA coefficient at lag k and $\epsilon_{t-k} = y_{t-k} - \hat{y}_{t-k}$ is the forecast error that was made at period $t - k$ [21]. Also, these polynomials can be represented by $\varphi(k)$ and $\theta(k)$, respectively. The series Y_t is said to be ARIMA (p, d, q) if

$$\varphi(k)(1 - k)d, Y_t = \theta(k)\epsilon_t \tag{2}$$

where d is the d th difference operator.

ARIMAX (p, d, q) can be represented by [22]:

$$\varphi(k)(1 - k)d Y_t = \beta(k)X_t + \theta(k)\epsilon_t \tag{3}$$

where, x_t is a covariate at time t and β is its coefficient.

ARIMAX model is now implemented considering psychological, physiological and combined variables.

Table 6 lists the MAPE for ARIMAX (1, 1, 0) implemented on psychological, physiological and hybrid model for different prediction horizon. Hybrid model combines both physiological and psychological variables in appropriate ratio. The weightage of physiological variable to psychological variable is varied from 80:20 to 20:80. The MAPE is recorded and tabulated for 1-day prediction. The PH for hybrid model was good for 80:20 weightage.

The Clarke grid error analysis zones (EGA) are shown in Fig. 12.

The grid divides the scatterplot obtained considering reference BG values against predicted BG values into five zones as shown in Fig. 12:

- Zone A: This zone indicates the points that lie within 20% of the reference.
- Zone B: This zone indicates the points that lie outside 20% but would not lead to inappropriate treatment.
- Zone C: This zone indicates the points that lead to unnecessary treatment.
- Zone D: This zone indicates the points that are potentially dangerous failures in detecting hypoglycemia or hyperglycemia.

Table 6 MAPE for ARIMAX implemented on psychological, physiological and hybrid model

MAPE for PH	30 min	60 min	2 h	3 h	1 day
ARIMAX physio	30.25	41.75	43.40	32.04	32.16
ARIMAX psycho	33.88	34.63	35.19	29.93	24.39
ARIMAX_hybrid (80:20 weightage)	11.12	15.29	23.41	29.14	22.73

Fig. 12 Clarke grid error analysis zones

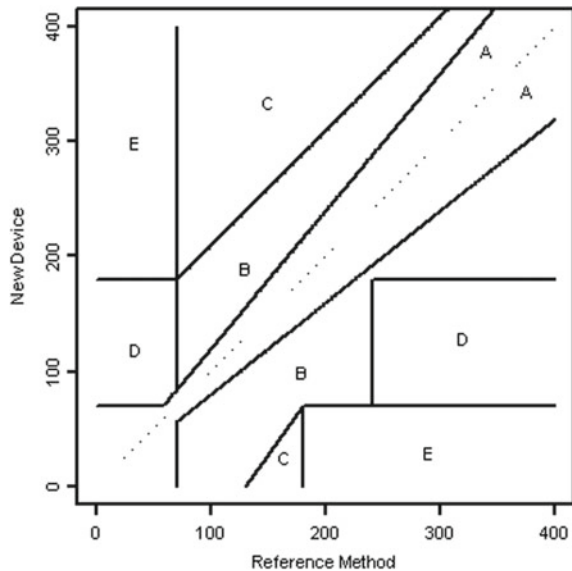


Fig. 13 EGA plot for ARIMAX hybrid model for PH 30 min

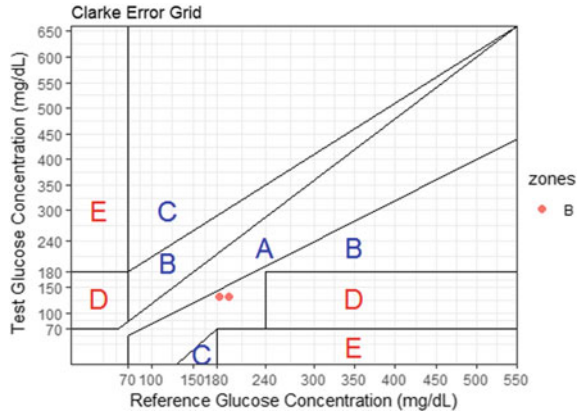
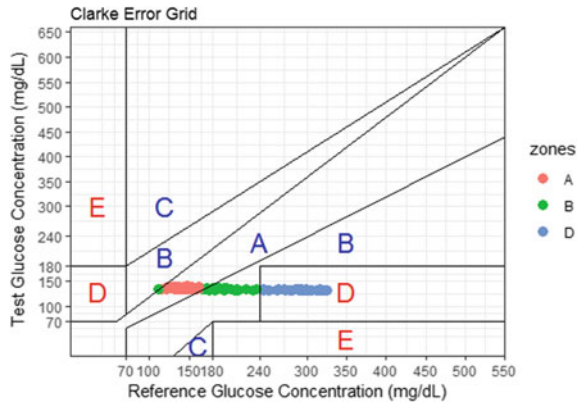


Fig. 14 EGA plot for ARIMAX hybrid model for PH 1 day



- Zone E: This zone indicates the points that would confuse the treatment between hypoglycemia and hyperglycemia.

EGA which was developed in 1987 [23] is used here to quantify the ARIMAX hybrid model. The Clarks error grid analysis plot for hybrid model for PH of 30 min and 24 h is as shown in Figs. 13 and 14.

The error grid analysis percentage table is given in Table 7.

Table 7 gives that in ARIMAX 30.20 percentage of points lie in Zone D, which is potentially dangerous failures in detecting hypoglycemia or hyperglycemia events.

4.1.2 ANN

Artificial neural networks (ANN) describe complex nonlinear relationships between the dependent variable and its predictors [24]. ANNs are proved to be highly effective in solving nonlinear real-world problems [25]. The ANN model used for the nonlinear data is represented as follows [24]:

Table 7 Error grid percentage table for ARIMAX

Error grid percentage table					
PH	A	B	C	D	E
30 min	–	100	–	–	–
60 min	–	100	–	–	–
2 h	–	62.5	37.5	–	–
3 h	–	41.66	58.33	–	–
24 h	43.75	26.04	–	30.20	–

Table 8 MAPE for ANN implemented on psychological (psycho), physiological (physio) and hybrid model

MAPE for PH	30 min	60 min	2 h	3 h	1 day
ANN physio	22.06	24.03	27.26	23.61	21.15
ANN psycho	34.14	34.56	37.37	33.96	18.78
ANN hybrid (80:20 weightage)	3.70	6.89	14.86	21.49	13.37

$$y_t = w_0 + \sum w_j \cdot g\left(w_0j + \sum w_{ij} \cdot y_{t-1}\right) + \varepsilon_t \tag{4}$$

where w_{ij} ($i = 0, 1, 2, \dots, p, j = 1, 2, \dots, q$) and w_j ($j = 0, 1, 2, \dots, q$) are the connection weights, p is the number of input nodes and q is the number hidden nodes. For dataset B, an average of 20 networks is taken each of which is a 19–10–1 network with 211 weights.

Table 8 lists the MAPE for ANN implemented on psychological, physiological and hybrid model for different prediction horizon. Hybrid model combines both physiological and psychological variables in appropriate ratio. The weightage of physiological variable to psychological variable is varied from 80:20 to 20:80. The MAPE is recorded and tabulated for 1-day prediction. The PH for hybrid model was good for 80:20 weightage.

The Clarks error grid analysis plot for hybrid model for PH of 30 min and 24 h is as shown in Figs. 15 and 16.

The error grid analysis percentage table is given in Table 9.

Table 9 gives that in ANN, 14.10 percentage of points lie in Zone D, which is potentially dangerous failures in detecting hypoglycemia or hyperglycemia events.

4.1.3 Support Vector Machine

The capability of SVM to solve nonlinear problems makes it interesting and suitable for time series forecasting [26, 27]. Considering a data training set, T , represented by [28]:

Fig. 15 EGA plot for ANN hybrid model for PH 30 min

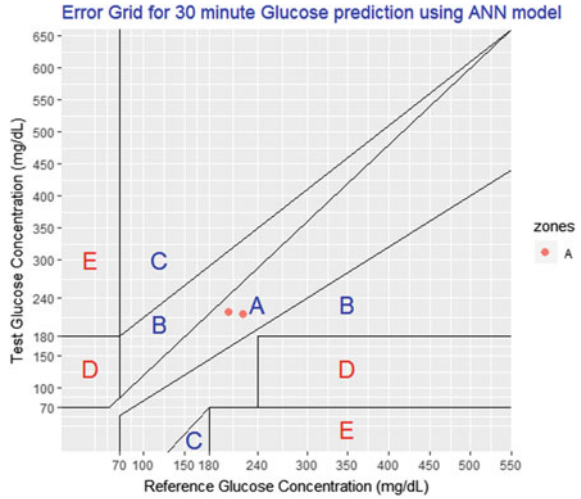


Fig. 16 EGA plot for ANN hybrid model for PH 24 h

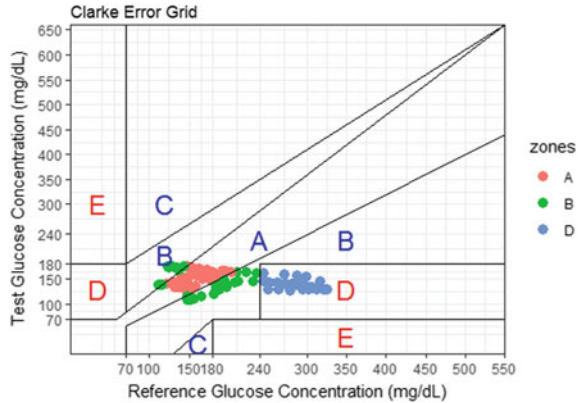


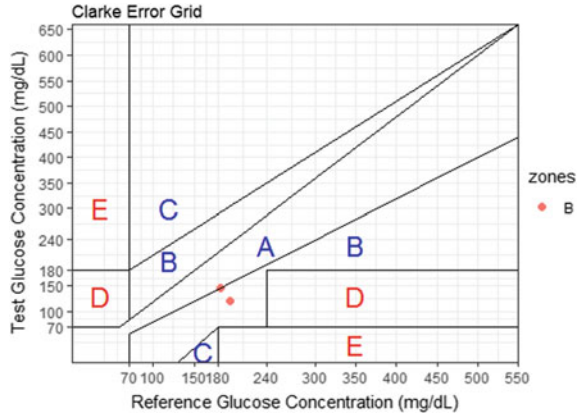
Table 9 Error grid percentage table for ANN

Error grid percentage table					
PH	A	B	C	D	E
30 min	100	–	–	–	–
60 min	100	–	–	–	–
2 h	50	50	–	–	–
3 h	33.33	66.67	–	–	–
24 h	36.65	48.42	–	14.901	0.0147

Table 10 MAPE for SVM implemented on hybrid model

PH	30 min	60 min	2 h	3 h	24 h
MAPE	31.22	30.04	29.04	27.41	26.39

Fig. 17 EGA plot for SVM hybrid model for PH 30 min



$$T = \{(x_1, y_1), (x_2, y_2), \dots (x_m, y_m)\} \tag{5}$$

Assume a nonlinear function, $f(x)$ given by:

$$f(x) = wT\Phi(xi) + b \tag{6}$$

where, w is the weight vector, b is the bias and $\Phi(xi)$ is the high dimensional feature space, which is linearly mapped from the input space x .

The number of support vectors used was 22,185.

Table 10 lists the MAPE for SVM implemented on only hybrid model with weightage set to 80:20 because it had poor performance on physiological and psychological model.

The Clarks error grid analysis plot for hybrid model for PH of 30 min and 24 h is as shown in Figs. 17 and 18.

The error grid analysis percentage table is given in Table 11.

Table 11 gives that in SVM, 25 percentage of points lie in Zone D, which is potentially dangerous failures in detecting hypoglycemia or hyperglycemia events.

5 Results and Discussion

Table 12 gives the comparison of ARIMAX, ANN and SVM-based hybrid model implemented.

Fig. 18 EGA plot for SVM hybrid model for PH 24 h

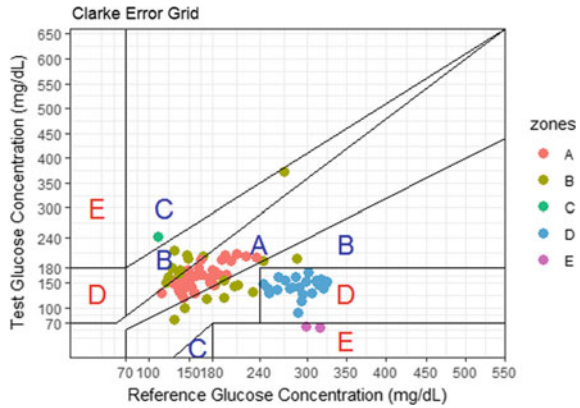


Table 11 Error grid percentage table for SVM

Error grid percentage table

PH	A	B	C	D	E
30 min	–	100	–	–	–
60 min	–	100	–	–	–
2 h	–	62.5	–	37.5	–
3 h	–	41.66	–	58.33	–
24 h	48.95	22.51	1.04	25.00	2.08

Table 12 Comparison table of hybrid model

MODEL	30 min	60 min	2 h	3 h	24 h
ARIMA	14.112	8.7511	10.409	14.797	23.415
ARIMAX	11.12	15.29	23.41	29.14	22.73
ANN	3.70	6.89	14.86	21.49	13.37
SVM	31.22	30.04	29.04	27.41	26.39

The comparison Table 12 reveals that ANN hybrid model results in least MAPE of 13.37 for a prediction horizon of 24 h and Fig. 19 also reveals that ANN follows the trend of the actual data indicating that as a best long-term prediction model for blood glucose prediction for the Ohio T1DM dataset B. Hence, the author recommends ANN for multivariate long-term forecasting of blood glucose values. Also the proposed hybrid model outperforms the other two single models, thus paving way to further work around hybrid permutation model hypothesis.

As seen in Fig. 20, ANN hybrid model yielded better result for 1-day ahead prediction with scatter plot in A, B and D zone since it had more data points to train itself.

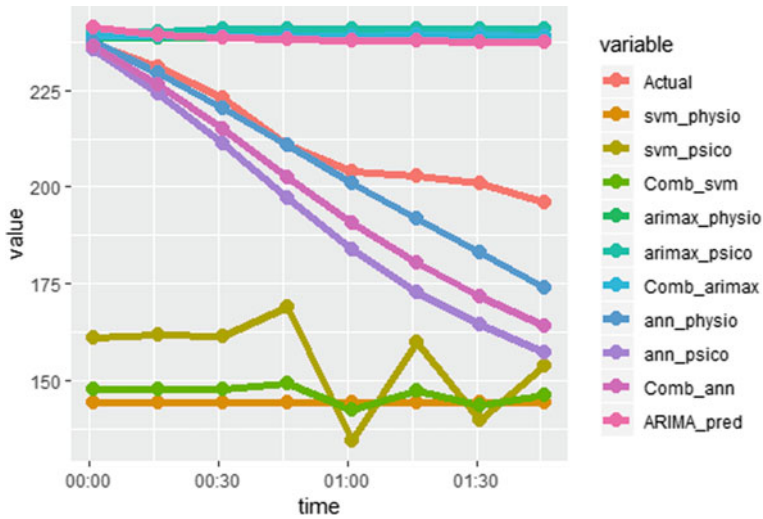


Fig. 19 2 h ahead prediction of all the model

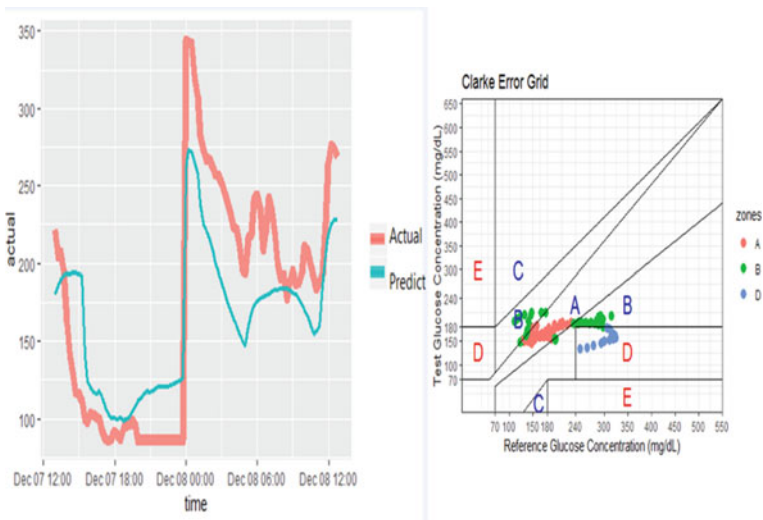


Fig. 20 1 day ahead prediction and EGA plot of hybrid ANN model

6 Conclusion

Time series forecasting requires data at regular intervals for accurate forecasting. They cannot be left unfilled. Hence, interpolation played a major role in our data preparation. Also, some of the self-reported data was converted to categorical. The

advantage of converting them to categorical is that they can be further converted to dummy variables. These dummy variables lead to parameter estimation which was one more supporting factor for forecast accuracy in our work. Lastly, during data preparation, the entire day was divided into 7 flags. This led to seasonality capture of data which enhanced the training model and hence the forecast accuracy. These parameters were well captured by ANN algorithm using hybrid model resulting in better forecast accuracy when compared to other algorithms implemented in this work.

The major role in capturing hyperglycemia or hypoglycemia is based on type of food consumed and physical activity. Since bolus insulin and food are closely related to these 2 parameters. The dataset had only carb count and exercise quality.

The prediction accuracy can hence be improved with large dataset, also sufficient and proper amount of information in the dataset leading to better learning by the algorithm.

Also, patients with T1DM should be educated with their insulin administration, since with no insulin in their body, its very challenging to manage their diabetes when compared to Type 2 diabetes mellitus (T2DM) patients.

Acknowledgements I thank Jnana Sanjeevini Hospital, Bangalore, and Ohio University for providing with their dataset for the research work.

References

1. Frandes M, Timar B, Timar R et al (2017) Chaotic time series prediction for glucose dynamics in type 1 diabetes mellitus using regime-switching models. *Sci Rep* 7. Article Number 6232. <https://doi.org/10.1038/s41598-017-06478-4>
2. Bremer T, Gough DA (1999) Is blood glucose predictable from previous values? A solicitation for data. *Diabetes* 48:445–451
3. Sparacino G, Zanderigo F, Maran A, Facchinetti A, Cobelli C (2007) Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series. *IEEE Trans Biomed Eng* 54:931–937
4. El Youssef J, Castle J, Ward WK (2009) A review of closed-loop algorithms for glycemic control in the treatment of type 1 diabetes. *Algorithms* 2:518–532
5. Gani A, Gribok A, Rajaraman S, Ward W, Reifman J (2009) Predicting subcutaneous glucose concentration in humans: data-driven glucose modeling. *IEEE Trans Biomed Eng* 56:246–254
6. Pros and cons of using the univariate model of financial analysis. <http://www.floridabankruptcyblog.com/pros-and-cons-of-using-the-univariate-model-of-financial-analysis/>. Accessed 20 Dec 2018
7. Phadke R, Prasad V, Nagaraj HC (2019) Time series based short term T1DM prediction of Librepro CGM sensor data: a novel ensemble method. *Int J Eng Adv Technol (IJEAT)* 8(6)
8. Jensen M, Cristensen TF, Tarnow L, Seto E, Johansen M, Hejlesen O (2013) Real time hypoglycemia detection from continuous glucose monitoring data of subjects with type 1 diabetes. *Diabetes Technol Ther* 15(7)
9. Marling C, Wiley M, Buneseu R, Shudrook J, Schwartz F (2012) Emerging applications for intelligent diabetes management. *AI Mag* 67

10. Polat K, Gunes S (2007) An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. *Digit Signal Process* 702–710
11. Baghdadi G, Nasrabadi A (2007) Controlling blood levels in diabetics by neural network predictor. *Eng Med Biol Soc* 3216–3219
12. Zecchin C, Facchinetti A, Sparacino G, Nicolao GD, Cobelli C (2012) Neural network incorporating meal information improves accuracy of short-time predictions of glucose concentration. *IEEE Trans Biomed Eng*
13. Hamdi T et al (2017) Artificial neural network for blood glucose level prediction. In: *International conference on smart, monitored and controlled cities*
14. Pappada S, Cameron BD, Rosman PM, Bourey RE, Papadimos TJ, Olorunto W, Borst MJ (2011) Neural network based real time prediction of glucose in patients with insulin dependent diabetes. *Diabetes Technol Ther* 135–141
15. Eren-Orukulu M, Cinar A, Quinn L, Smith D (2009) Estimation of future glucose concentrations with subject specific recursive linear models. *Diabetes Technol Ther* 243–253
16. Petridis V, Kehagias A, Petrou L, Bakirtzis A, Kiartzis S, Panagiotou H, Masalaris N (2001) A Bayesian multiple models combination method for time series prediction. *J Intell Robot Syst* 31(1–3):69–89
17. <https://diatribe.org/abbott-freestyle-libre-pro-cgm-system-fda-approval>. [Online]. Accessed 2 Mar 2016
18. Marling C, Bunesco C (2018) The OhioT1DM dataset for blood glucose level prediction. In: *IIIrd international workshop on knowledge discovery in healthcare data*, Stockholm, Sweden, 13 July 2018
19. Phadke R, Prasad V, Nagaraj HC (2019) Precise humane diabetes management: synergy of physiological and psychological data in AI based diabetes. *Int J Sci Technol Res* 8(11)
20. Pankratz A (1991) *Forecasting with dynamic regression models*. Wiley-Interscience
21. Khashei M, Bijari M, Ardali GAR (2009) Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs). *Neurocomputing* 72(4–6):956–967
22. Adebisi AA, Adewumi AO, Ayo CK (2014) Comparison of ARIMA and artificial neural networks models for stock price prediction. *J Appl Math* 2014:7 pages. Article ID 614342
23. Clarke WL (2005) The original Clarke error grid analysis (EGA). *Diabetes Technol Ther* 7(5):776–779. <https://doi.org/10.1089/dia.2005.7.776>
24. Zhang G, Patuwo B, Hu MY (1998) Forecasting with artificial neural networks: the state of the art. *Int J Forecast* 14(1):35–62
25. Vapnik VN (1995) *The nature of statistical learning theory*, 1st edn. Springer-Verlag, New York
26. Zbikowski K (2014) Using volume weighted support vector machines with walk forward testing and feature selection for the purpose of creating stock trading strategy. *Expert Syst Appl* 42
27. Cao LJ, Tay EH (2001) Support vector with adaptive parameters in financial time series forecasting. *IEEE Trans Neural Netw* 14:1506–1518
28. Ojemakinde BT (2006) *Support vector regression for non-stationary time series*. Master thesis, University of Tennessee, Knoxville
29. Smith TF, Waterman MS (1981) Identification of common molecular subsequences. *J Mol Biol* 147:195–197. [https://doi.org/10.1016/0022-2836\(81\)90087-5](https://doi.org/10.1016/0022-2836(81)90087-5)
30. May P, Ehrlich H-C, Steinke T (2006) ZIB structure prediction pipeline: composing a complex biological workflow through web services. In: Nagel WE, Walter WV, Lehner W (eds) *Euro-Par 2006*. LNCS, vol 4128. Springer, Heidelberg, pp 1148–1158. https://doi.org/10.1007/11823285_121
31. Foster I, Kesselman C (1999) *The grid: blueprint for a new computing infrastructure*. Morgan Kaufmann, San Francisco
32. Czajkowski K, Fitzgerald S, Foster I, Kesselman C (2001) Grid information services for distributed resource sharing. In: *10th IEEE international symposium on high performance distributed computing*. IEEE Press, New York, pp 181–184. <https://doi.org/10.1109/HPDC.2001.945188>

33. Foster I, Kesselman C, Nick J, Tuecke S (2002) The physiology of the grid: an open grid services architecture for distributed systems integration. Technical report, Global Grid Forum
34. National Center for Biotechnology Information. <http://www.ncbi.nlm.nih.gov>

Master and Slave-Based Test-Bed for Waste Collection and Disposal: A Dissertation



Shreeram V. Kulkarni, N. Samanvita, Shruti Gatade, and Sowmya Raman

1 Introduction

The process of automating things is exploited in almost every major area of life. Making things automatic reduces the burden on humans. The cost and effort used in manually controlled products is much higher than automated systems. Considering the fact that the problem of effective waste management is one of the biggest problems of modern times; it is absolutely necessary to address this problem [1]. The correct waste management system is a must for the sanitary society in general and for the world as entire. Waste management includes the planning, financing, construction, and operation of facilities for the collection, transport, recycling, and final disposal of waste [2]. Currently, large cities around the world are in need of demanding solutions for solid waste management (SWM), due to the growth of residential areas and the economy. SWM is an expensive urban service that consumes around 20–50% of the annual municipal budget in developing countries [3].

In the proposed system, there are two main participant systems, the master bin and slave bin. The master bin is a trash collector and disposer that receives a signal from the slave source garbage bin and begins its procedure after receiving it. The garbage collector moves around a corridor or along predetermined pathways within the building or neighborhood, stopping at designated slave bins to collect trash [4]. The lane that the master bin follows is designated in black so that (infrared) IR sensors can readily pick it up. When the master bin fills up, a signal is sent out, and the container proceeds to its dumpsite without stopping at any other collection station. The collection and disposal of garbage information can be tracked through Web site, and the message is made to send to the concerned authorities. This prototype of proposed system automates the collection and disposal mechanism of an area or

S. V. Kulkarni (✉) · N. Samanvita · S. Gatade · S. Raman
Nitte Meenakshi Institute of Technology, Yelahanka, Bengaluru 560064, India
e-mail: Shreeram.kulkarni@nmit.ac.in

institutional premises. The system comprises of mainly multiple slaves and one master as a participant. It capitalizes on the demarcations made at the edge of roads or at the edge or corridors of premises to make its system functional in its objective. The multiple slave bins of an area are connected to master bin wirelessly using X-bee in this prototype. The slave bin continuously monitors the garbage pile-up level in it. As the garbage is piled up in any of the slave bin, the slave bin communicates this situation wirelessly to master. The master bin in this system is a maneuverable asset that upon receiving the information from any of source bin of an area activates its automatic collection and disposal mechanism [5, 6]. The master generally stays hold until and unless an interrupt is received from any slave, and the moment it receives the interrupt, it starts maneuvering toward slave bin. The appropriate slave bins in the system are located by comparing radio frequency identification (RFID) tags with the pre-stored database [7]. The master moves to the area of the particular slave bin and executes collection and disposal mechanism. The system comprises of multiple slaves in connection with single master wirelessly through X-bee movement to source location; the general closed-loop system to facilitate the master is shown in Fig. 1.

In the present scenario, there are many instances of untimely collection of garbage and its disposal creating garbage menace [8]. The garbage overflowing in locality/premises creates esthetically unpleasant environment and encourages epidemic and the garbage segregation [7, 8]. The mix of wet and dry waste makes the segregation difficult that entails the difficulty in recycling and reuse of garbage. The main objective of the paper is to automate monitoring of garbage pile-up level in localities/institutional premises via multiple slaves. Initiate automatic collection and disposal mechanisms via master garbage bin and slave bin and keep concerned parties informed, notify and alert authorities on violations of segregation norms or

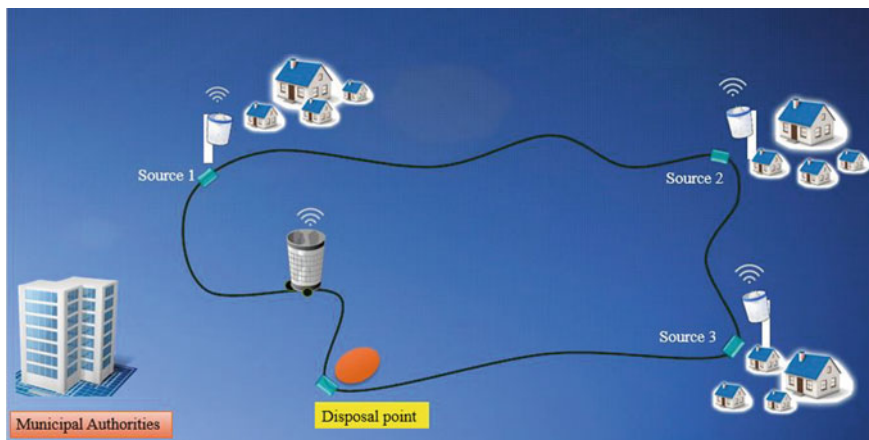


Fig. 1 Closed-loop system to facilitate the master

other issues, keep a record of garbage collection pile-up and collection on server, and keep the surrounding area clean, hygienic, and healthy [9, 10].

2 Test-Bed Requisites

The prototype of proposed system automates the collection and disposal mechanism of an area or institutional premises. The system comprises of mainly multiple slaves and one master as a participant. Hardware required is Arduino mega microcontroller, ultrasonic sensors, GSM/GPRS module, X-bee, IR sensors, DC motors, motor control board, FID reader, servomotors, LED, buzzer, and software required is Arduino IDE and XCTU.

- **Arduino Mega 2560 and Communication Network in X-bee:** Arduino is an open-source platform for creating electronic projects that combines several parts and interfaces into a single board [10]. It comes with it all you need to get started with the microcontroller. It can easily plug it to computer with a USB connection or battery with an AC-to-DC converter. The integrated development environment (IDE) is used to program the Arduino mega 2560, which is the same for all boards and works both online and offline. Almost all Arduino shields are also compatible with the mega. OEMs may employ X-bee RF modules to build a universal mark that can be used on a variety of platforms, including points, ZigBee/Mesh, and 2.4 GHz and 900 MHz solutions. OEMs that use the X-bee can switch out one X-bee for another according on the application's requirements, reducing development time, risk, and time-to-market. To be setup on a specific network or to serve as a master or slave X-bee, the X-bees must be programmed using software XCTU.
- **Ultrasonic Sensor (HCSR04) and IR Sensors:** Ultrasonic sensors are devices that take electrical–mechanical energy transformation to measure distance of target object from the sensor [11]. An infrared sensor is electronic equipment that emits and/or detects infrared radiation to perceive certain features of its surroundings. Infrared sensors are also effective in detecting motion and measuring the heat released by an item.
- **RFID Reader:** A radio frequency identification reader (RFID reader) is a device cast-off to collect information from an RFID tag, which is used to keep a track on individual objects. Radio waves are utilized to transfer data from the tag to a reader [12, 13].
- **Servomotor and SIM 900A:** A servomotor is a spinning or linear actuator that can regulate angular or linear position, velocity, and acceleration with pinpoint accuracy. It is constructed from a suitable motor, a gear, and a position feedback sensor. The servomotor's job is to take a control signal that indicates the desired output position of the servo shaft and drive its DC motor until the shaft rotates to that point. The motor which was used here has the torque rating of SIM900A is a GSM/GPRS module which could be used to send/receive messages and make a

Table 1 Specifications of radio frequency identification reader

System	Specification
Module	MF522-ED
Working frequency	13.56 MHz
Card reading distance	0 ~ 60 mm
Protocol	SPI
Working current	13–26 mA/3.3 V DC
Data communication speed	10 Mbits/Max

calls [14, 15]. It can also connect to the Internet using GPRS. The SIM900 Quad-band/SIM900A dual-band GSM/GPRS module includes a breakout board and a basic system. It uses AT instructions to interact with controllers (GSM 07.07, 07.05, and SIMCOM-enhanced AT commands). The software power on and reset functions are supported by this module.

- Motor Control Board (LM298) and DC Motor:** LM298 is a motor control board which controls the operation of DC motor. It has three ports, 5, 12 V power supply and one ground. It takes input from 4 ports for two DC motors to be controlled in either of the directions. The robot runs on wheels, and it has an arm which rotates; this is made possible by DC motors. In order to attain linear motion of the vehicle, DC motors are used. There are six DC motors fixed at the ends of the robot for the movement of the robot, and it requires three motors for the robotic hand to work properly.
- Arduino Nano Overview and XCTU:** The ATmega328-based Arduino Nano is a compact, comprehensive, and breadboard-friendly board. It functions in a similar way as the Arduino Duemilanove but in a different packaging. It just has a DC power connector and uses a Mini-B USB cable rather than a conventional one. XCTU is a free multi-stage program that lets developers utilize a graphical user interface to connect with Digi RF modules. It is simple to setup, configure, and test an X-bee module which includes new tools. The interface depicts the X-bee network graphically as well as the signal strength of each connection, and the X-bee API frame builder, which intuitively facilitates in the generation and interpretation of API frames for X-bees in API mode these are two unique features. XCTU may be configured on a variety of RF devices [16]. Frame generators, frames interpreter, recovery, range test, and firmware explorer are just a few examples of embedded tools that may be utilized without an RF module (Table 1).

3 Workflow: Master Bin and Slave Bin

The master bin is the central system in this prototype, and the block diagram is shown in Fig. 2. Master bin is mainly designed to execute to reach the destination of source of garbage, collect the waste, and dispose it to another site. Also, it is programmed and

designed to alert master bin comprises of MCU which is Arduino mega in this case. Arduino mega is interfaced with line following robot system, water sensors, X-bee, GSM/GPRS module, servomotors, buzzer, and ultrasonic sensors, master bin which follows the path and fixed on top of the line follower robot. This bin has an Arduino interfaced with ultrasonic sensors to check the pile-up level, X-bee to receive the wireless signal from slave bin when slave garbage source bin is full, servomotors for operating collection and disposal mechanism automatically. The master bin receives information of filling of source garbage bin through X-bee wirelessly from the slave or the source bin when it is filled with garbage up to brim.

After the interrupt, the collection mechanism is initiated in master bin. Once its initiates its movement by activating the line follower bot toward the slave bin, message is sent to the concerned authorities about its initiation for the collect of garbage from the particular slave bin. A message is also sent to public of corresponding slave bin area to refrain from adding more bin until cleanup has taken place, and the frequency of collection at different sites is uploaded to Web page using GSM/GPRS module; thus, the garbage collection rate at different sites can be monitored and studied. The ultrasonic sensors are incorporated in bin to monitor its own garbage filling level and give appropriate alert and indication of filling up. Water sensor senses whether there any wet waste present in the system, if presents, it alerts the concerned people and creates alarm. An actuator mechanism is set up using “servomotor” which allows control of angular position which is used to open the lids of bin for the purpose of collection and disposal. When the main bin arrives at the correct source bin to collect the garbage, the upper lid operates with the actuation of servomotor, and garbage is collected. And the lower lid opens at the disposal point when the bin has to dispose the garbage.

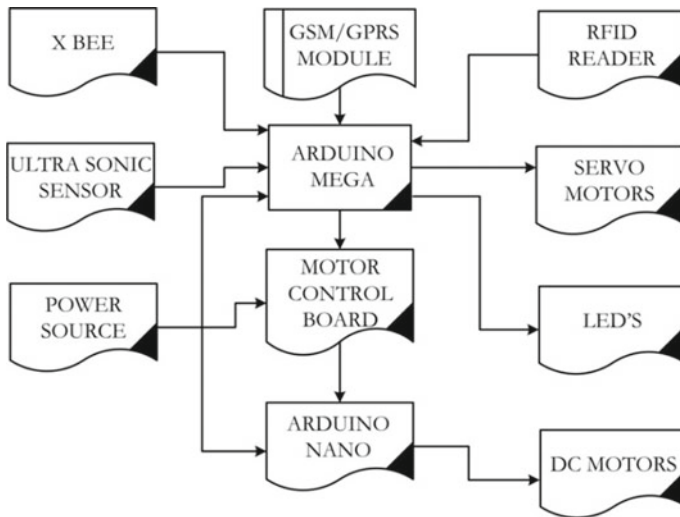


Fig. 2 Block diagram of master bin

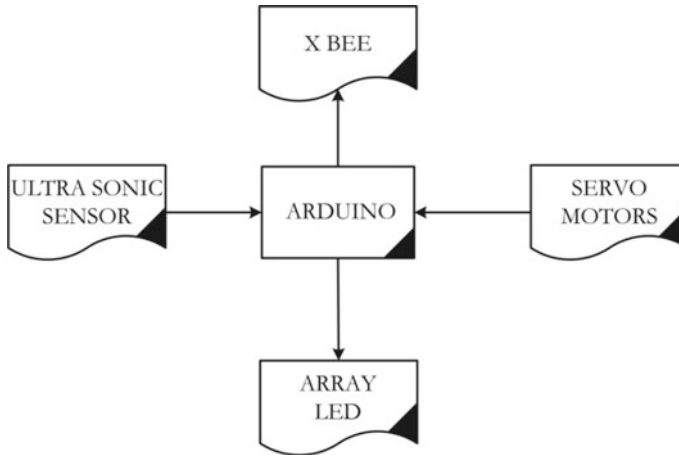


Fig. 3 Block diagram of slave bin

The block diagram of slave bin is shown in Fig. 3. The slave bins are the source of garbage generating points. These are to be stationed at tragic height and position to facilitate the automatic collection and disposal. The master arrives under the slave bin which is sensed by sensors incorporated in slave which then operates its disposal mechanism. Slave bin consists of array of led whose color raging from red to green. It is also interfaced with X-bee, ultrasonic sensors, and servomotors. The level of garbage in source bin is sensed by ultrasonic sensor and is transferred serially to Arduino, and when the sensed value reaches below the critical value, information is sent through X-bee to master bin for it to initiate the automatic collection. The pile-up levels of bin visually indicated using LEDs as the garbage pile-up level grow the led shifts from green to red. Bin pile-up levels are visually indicated by LEDs, which change from green to red as the garbage pile-up level increases. Once filled, the buzzer and red LEDs turned on. As soon as the master arrives, the ultrasonic sensors fixed at the bottom of the slave bin sense the change in distance thus open up the bottom lid and pushes the garbage out using servomotor mechanism as depicted in Figs. 4 and 5.

3.1 Control Mechanism

The robot that can follow a path is known as a line follower robot shown in Fig. 6. On a white surface, the route may be seen as a black line (or vice-verse). It is a combined design based on mechanical, electrical, and computer engineering understanding. A line follower is designed in order to carry the “Master bin” on top of it. A line follower has IR sensors which detect the intensity of light being reflected from the ground. While it is programmed to run on a black line on a white surface or any

Fig. 4 Master bin equipped with sensors

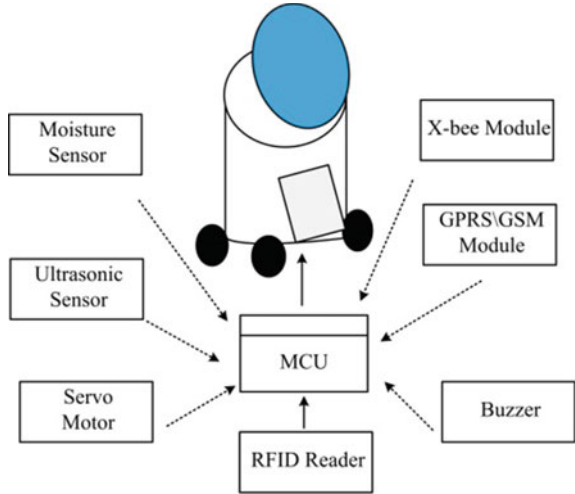
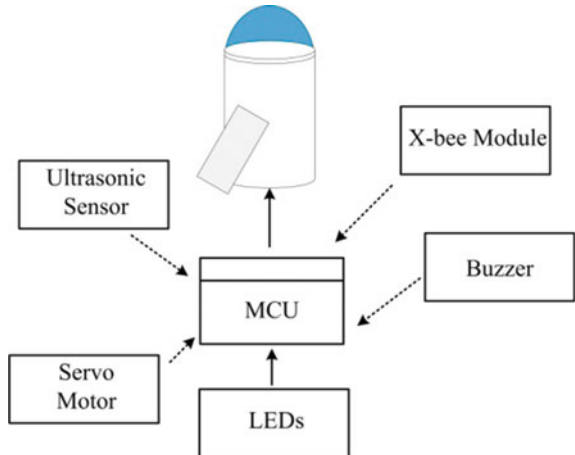


Fig. 5 Slave bin equipped with sensors



dark line on a light surface. Depending upon the output of the IR sensors, the line following robot is coordinated to move in appropriate direction.

The motors are driven in this line follower robot using motor driving circuit which depending upon the input from the Arduino runs or stops the motor to give

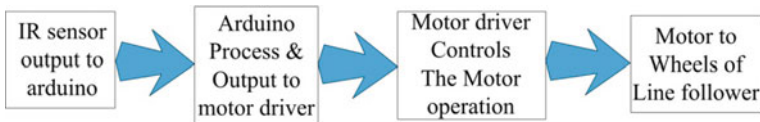


Fig. 6 Design of line follower robot

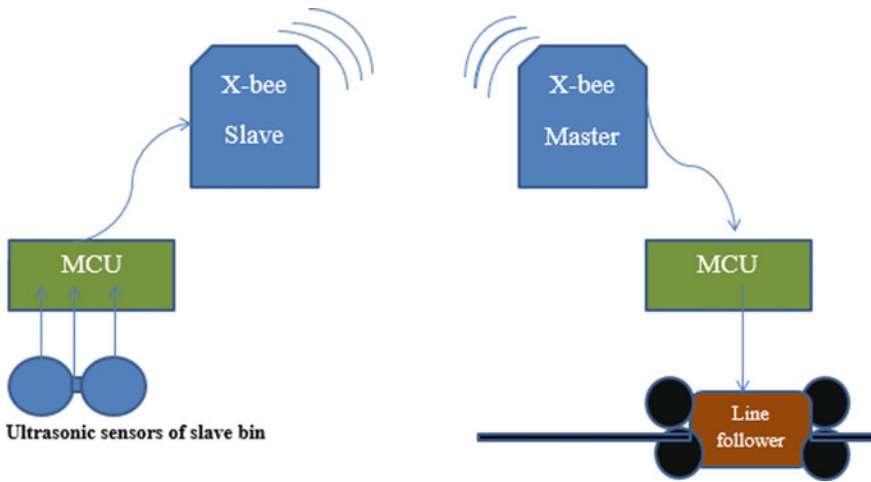


Fig. 7 Block diagram of communication through X-bee

line following robot an appropriate path to follow. A 12 V supply is provided to motor driving circuit (LM298) for it provide motors appropriate speed and torque. X-bee's are designed to have one master and multiple slaves, depicted in Fig. 7. All slaves communicate to its master through designated ID. The master and slave have same ID when the slave bins (Source garbage generating points) are full with garbage; the ultrasonic sensors fixed in the bin sense that, and this information is communicated to the master X-bee of master bin. The data are transferred in the form of data frame which contains the status of fill and open. Upon continuous reception of data frames from slave bins, the master X-bee of master bin interprets the messages, and upon finding of fullness of any bin, its starts its collection mechanism.

4 Detailed Discussions

A transceiver (transmitter/receiver) and an antenna, which are usually combined, make up an RFID reader. A transponder (transmitter/responder) and an antenna make up an RFID. The line follower robot with RFID is shown in Fig. 8. The RFID tag is read when the reader delivers a radio signal to the transponder, which activates the transponder and sends data back to the transceiver.

Here, we are using RFID reader to know the position of the slave bin. RFID cards are placed beneath the slave bins, and an RFID reader will be placed beneath the line follower robot. The master is programmed with database of RFID tags of corresponding slave bins, upon encountering any RFID tag on route toward slave bin to collect garbage, the master bin compares that with the database; if the tag matches with the appropriate, the master bin stops, and the collection mechanism is initiated.

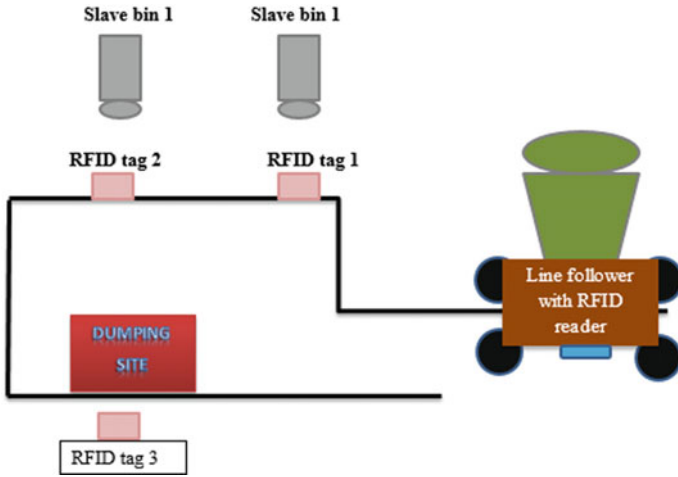


Fig. 8 Block diagram of line follower robot with RFID reader

A servomotor is used here in this prototype; it is fitted at the edge of the lids of both master and slave bins; the degree of rotation of motor is controlled to control the opening and closing of lids of both slave as well master bin. An ultrasonic sensor is fitted at the base of the slave bin so that it detects the main bin which comes to collect the garbage. After detecting the correct RFID tag, the master bin that is the line follower stops, and the upper lid of master opens, and the bottom lid of slaves open. The internal mechanism of slave bin facilitates the push of garbage from slave to master. Figure 9 shows experimental setup of line following robot attached with sensors.

Figures 10, 11 and 12 depict garbage collection statistics for a three-month period (from April 10, 2021 to July 10, 2021). The waste was disposed of after the bin

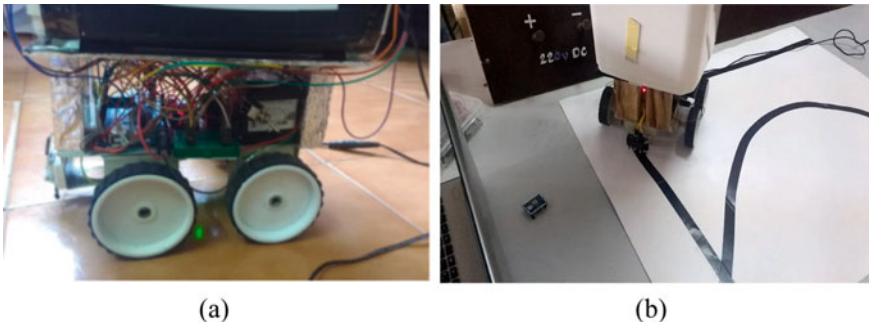


Fig. 9 Experimental setup: a line following robot attached with sensors and b master following the line to reach slave location

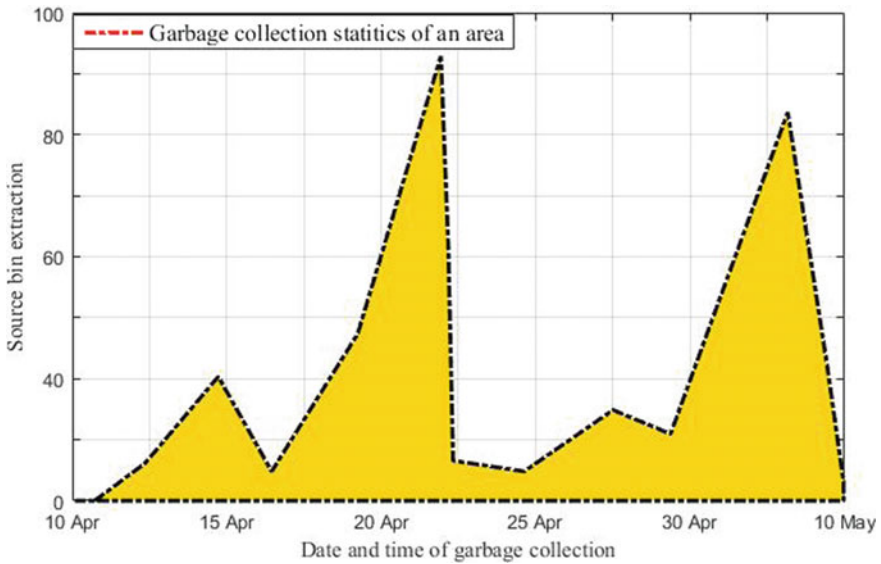


Fig. 10 Update of garbage collection data on server-ThingSpeak-channel status from 10th April 2021 to 10th May 2021

extraction and disposal data were obtained. The data on server-ThingSpeak-channel status show the area covered during the mentioned period.

5 Conclusion

The automatic collection and the disposal of the garbage could be achieved through an interconnected slaves and master garbage bins. This system could also reduce the unawareness among authorities and public regarding garbage bin piling and collection and disposal usage as the system could update the statics of collection and disposal on the server. The system also kept public and authorities informed and apprised by sending them periodic messages. This system also kept check on the public fowling segregation norms or not, hence maintaining the segregation of dry and wet waste efficiently. Having such system in place could reduce the un-hygienic conditions and could bring down the epidemic level as emanated from the untimely collection and non-segregation of waste. This system also facilitated the resource management in terms of labor management by making everything automated.

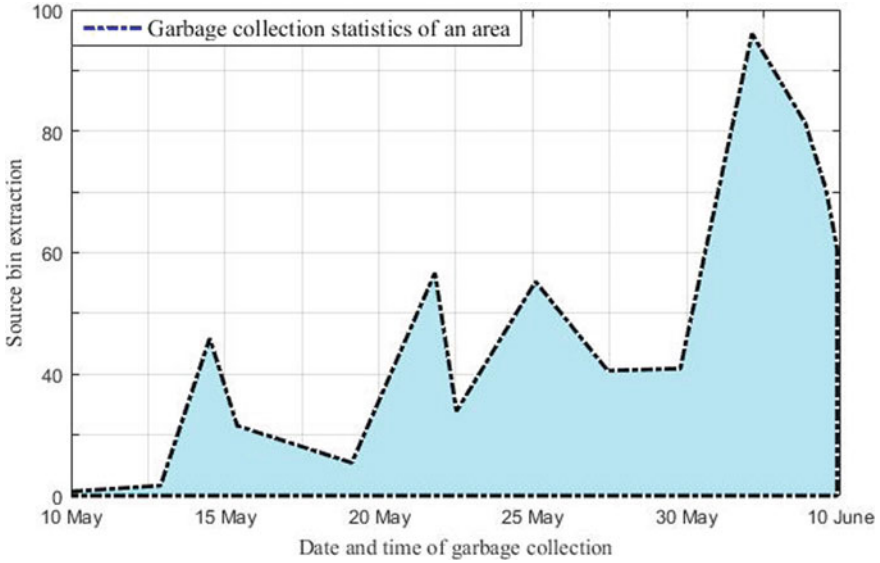


Fig. 11 Update of garbage collection data on server-ThingSpeak-channel status from 10th May 2021 to 10th June 2021

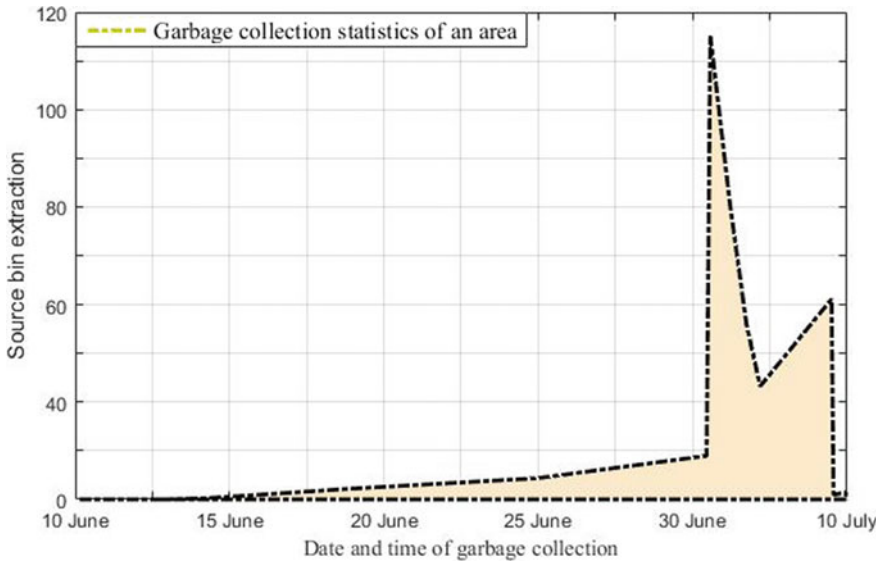


Fig. 12 Update of garbage collection data on server-ThingSpeak-channel status from 10th June 2021 to 10th July 2021

References

1. Arebey M et al (2009) Solid waste monitoring and management using RFID, GIS and GSM. In: 2009 IEEE student conference on research and development (SCOREd). IEEE
2. Arampatzis T, Lygeros J, Manesis S (2005) A survey of applications of wireless sensors and wireless sensor networks. In: Proceedings of the 2005 IEEE international symposium on Mediterranean conference on control and automation intelligent control, 2005. IEEE
3. Longhi S, Marzioni D, Alidori E, Di Buo G, Prist M, Grisostomi M, Pirro M (2012) Solid waste management architecture using wireless sensor network technology. In: Proceedings of the 2012 5th international conference on new technologies, mobility and security (NTMS), Istanbul, Turkey, 7–10 May 2012, pp 1–5
4. Narendra Kumar G, Swamy C, Nagadarshini KN (2014) Efficient garbage disposal management in metropolitan cities using VANETs. *J Clean Energy Technol* 2:258–262
5. Saji RM, Gopakumar D, Kumar SH, Sayed KNM, Lakshmi S (2016) A survey on smart garbage management in cities using IoT. *Int J Eng Comput Sci* 5:18749–18754
6. Younis O, Fahmy S (2004) HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Trans Mob Comput* 3:366–379
7. Al-Khatib IA, Monou M, Abu Zahra ASF, Shaheen HQ, Kassinos D (2010) Solid waste characterization, quantification and management practices in developing countries. A case study: Nablus district, Palestine. *J Environ Manage* 91(5):1131–1138
8. Eisted R, Larsen A, Christensen T (2009) Collection, transfer and transport of waste: accounting of greenhouse gases and global warming contribution. *Waste Manage Res* 27(8):738–745
9. Bhat VN (1996) A model for the optimal allocation of trucks for solid waste management. *Waste Manage Res* 14(1):87–96
10. Hannan MA, Arebey M, Basri H (2010) Intelligent solid waste bin monitoring and management system. *Aust J Basic Appl Sci* 4(10):5314–5319. ISSN 1991-8178
11. Ali ML, Alam M, Rahaman MANR (2012) RFID based e-monitoring system for municipal solid waste management. In: 7th international conference on electrical and computer engineering, Dec 2012
12. Singh T, Mahajan R, Bagai D (2016) Smart waste management using wireless sensor network. *Int J Innov Res Comput Commun Eng (IJIRCCE)* 4(6)
13. Healy M, Newe T, Lewis E (2008) Wireless sensor node hardware: a review. In: 2008 IEEE sensors, pp 621–624
14. Nithya L, Mahesh M (2016) A smart waste management and monitoring system using automatic unloading robot. *Int J Innov Res Comput Commun Eng (IJIRCCE)* 4(12):20838–20845
15. Bashir A, Banday SA, Khan AR, Shafi M (2013) Concept, design and implementation of automatic waste management system. *Int J Recent Innov Trends Comput Commun (IJRITCC)* 1(7):604–609
16. Al Mamun MA, Hannan MA, Hussain A, Basri H (2015) Integrated sensing systems and algorithms for solid waste bin state management automation. *IEEE Sens J* 15(1):561–567