R. Jagadeesh Kannan
S. Geetha
Sravanthi Sashikumar
Carl Diver   *Editors*

# International Virtual Conference on Industry 4.0

Select Proceedings of IVCI4.0 2021

Springer

# Lecture Notes in Electrical Engineering

## Volume 1003

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. \*\***

R. Jagadeesh Kannan · S. Geetha ·
Sravanthi Sashikumar · Carl Diver
Editors

# International Virtual Conference on Industry 4.0

Select Proceedings of IVCI4.0 2021

*Editors*
R. Jagadeesh Kannan
School of Computer Science
and Engineering
Vellore Institute of Technology
Chennai, India

S. Geetha
School of Computer Science
and Engineering
Vellore Institute of Technology
Chennai, India

Sravanthi Sashikumar
Department of Science and Engineering
Manchester Metropolitan University
Manchester, UK

Carl Diver
Academic Lead Industry 4.0
Manchester Metropolitan University
Manchester, UK

# Contents

# About the Editors

**R. Jagadeesh Kannan** is a Professor and Former Dean of the School of Computer Science & Engineering at Vellore Institute of Technology, India. He completed his Ph.D. degree in Handwritten Character Recognition using Hybrid Techniques from Anna University, Chennai, India. He got his M.E. degree in Computer Science & Engineering from National Engineering College, Tamil Nadu, and his B.E. in Instrumentation & Control Engineering from Madurai Kamaraj University, Tamil Nadu, India. Prof. Kannan has over 18 years of teaching and industrial experience in reputed organizations. Prof. Kannan has got several publications in conference proceedings and journals of national and international repute. His research interests are neural networks, fuzzy logic, neuro-fuzzy systems, soft computing tools, pattern recognition, natural language processing, image processing, networking, printed, handwritten and cursive character recognition, and artificial intelligence. Dr. Kannan is an Active Member of several Indian and international societies such as IEEE, ISTE, IACSIT, SDIWC, IFRSA, and IAENG.

**S. Geetha** is a Professor and Associate Dean (Research) of the School of Computer Science & Engineering at Vellore Institute of Technology, Chennai Campus, India. Dr. Geetha received her master's (Computer Science and Engineering) and doctoral (Information and Communication Engineering) degrees from Anna University in 2004 and 2011, respectively. Her areas of interest are computer vision, information and network security, and machine learning. Dr. Geetha has co-authored 06 books, 06 book chapters, and over 100 journal and conference papers.

**Sravanthi Sashikumar** is Deputy Head of the Department of Engineering at Manchester Metropolitan University, United Kingdom. She received a bachelor's degree in mechanical engineering from the University of Madras, India, in 2002 and a master's degree in Engineering Control Systems and Instrumentation from the University of Huddersfield, in 2005. She completed her Ph.D. from the University of Bolton in 2012, and her research interest areas are structural analysis of vehicle crashworthiness, passive safety of road vehicles, vehicle occupant safety, and impact biomechanics. Dr. Sravanthi has got several publications to her credit.

**Carl Diver** is Academic Lead on Industry 4.0 and Reader in Industrial Digitalization at Manchester Metropolitan University. Carl joined Manchester Metropolitan University in 2018 to lead the University's Industry 4.0 activities. Before academia, Carl worked in the manufacturing sector for over 20 years, firstly for a large multinational and then establishing his consultancy. At Delphi Automotive Systems, Carl's focus was on fuel injection equipment from both an R&D and manufacturing angle. More recently, Carl led Industry 4.0 activity, research, and teaching for the manufacturing group in the School of Mechanical Aerospace and Civil Engineering at the University of Manchester. He helped establish the academic conference at the Industry 4.0 summit in Manchester, and his Ph.D. students won international competitions such as the Siemens Open Space challenge at Hannover Messe in 2018. Carl has been speaking about Industry 4.0 at conferences and events in the UK, Europe, the Middle East, and Asia. He is an Associate Co-editor of the Virtual and Physical Prototyping Journal and continues helping businesses with more advanced, efficient, and controlled processes.

# Design of Web-Based Agile Meeting Dashboard

R. Dhanalakshmi, S. Sankar, V. Srinidhi, and K. Srividya

## 1 Introduction

In the corporate environment, managing the work routine is required to meet the goals and deadlines. It gives clarity of work and makes the team to be focused on the expected output. Agile management [1] methods provide greater interaction among the team members. It is responsive to any immediate changes thus making the task tracking more dynamic. It is also more flexible which helps the team to manage [2] their work effectively and efficiently.

In every organization, meetings are conducted daily where the team members discuss their doubts and put forward their opinions. Few questions like 'what is done', 'what is to be done', 'when to be done', and 'what are the issues faced' are asked during a meeting. To obtain feedback [3] for the work completed, a retrospective meeting is conducted at the end of each sprint which results in better delivery quality. This software provides better planning of activities based on the priority of work. The activities can be monitored constantly which improves the flow of work.

In the rapid development of technologies, this dashboard deals with understanding the individual's task status along with the time sheets and tabulating the work progress. It also plans the meeting according to the participant's time slot availability. The scrum board [4] helps in displaying the tasks and the progress can be easily monitored through graphical representation. This concept includes the features

S. Sankar · V. Srinidhi · K. Srividya
Department of Computer Science and Engineering, KCG College of Technology, Chennai, Tamilnadu, India
e-mail: sankar@kcgcollege.com

R. Dhanalakshmi (✉)
School of Computer Science and Engineering, Vellore Institute of Technology (VIT), Chennai, India
e-mail: dhanalakshmisai@gmail.com

such as a discussion forum, chatbox, meeting scheduler, screen share, calendar, document sharing, email notification, meeting recordings (in mp3 or mp4 format), and storyboard polls.

This paper is organized as follows. Section 2 analyzes some existing related work. Section 3 describes the system overview. Section 4 discusses the system prototype. In Sect. 5, the result is discussed. Section 6 concludes the paper.

## 2   Related Work

There are many existing agile meeting tools that easily manage the sprint. Each application has few features but till date none of them has incorporated all the possible features to make an efficient agile meeting dashboard. The main features and technologies of the existing systems are described here.

A.   Zoho Sprint

Zoho Sprint is an agile meeting solution that enables to design projects and provides the product without any delay [5]. This system includes time sheets, meeting scheduler [6], and the graphical representation of each task's results. The performance of this tool is fast and it has a simple configuration.

2   Axosoft

Axosoft is a project management tool that is used for managing projects that are complex. It helps in planning [7] the tasks particularly to the organizations which use the scrum approach. [8]. The main feature of this tool includes the creation of a proper backlog and automated progress generator which customizes the dashboard [9] to display the results. It also helps the user to monitor the project closely and accurately.

3   Scrum-It

Scrum-It is an open-source platform which is developed by the BS group [10]. The user can access this software without any license since it is free of cost [11]. Multiple projects can be managed by this software concurrently. The project-oriented data [12] is stored directly on the database server. In this platform, the scrum board displays the tasks and the burndown chart is used to monitor the work progress.

4   JIRA

JIRA [13] is a licensed project management tool. The main purpose of developing this tool is to identify the bugs and issues [14]. This tool can be used by organizations which have the iterative and incremental SDLC (Software Development Life Cycle) approaches in their project. Moreover, the status call can be scheduled and to notify members about their work, bots are configured.

5   Microsoft Azure

Microsoft Azure assigns the planned tasks based on their priorities. It has a user-friendly interface which manages the user stories and sprints. The status of the project is displayed on the scrum board [15]. To minimize the loss, Azure helps with project forecasting. Bugs can be easily detected and resolved.

## 3   System Overview

After analyzing various existing systems, we have come up with the idea of including additional features to our advanced software. They are further explained below.

A.  Discussion Forum

The Agile Meeting Dashboard provides a forum where the team members can discuss and ask their queries on the particular task assigned to them. For successive assignments, the new discussion forum is created automatically. In a decision forum, each person's knowledge [16] is shared which results in effective decision-making.

2   Chat Box

It has a chat box feature where the members can have a one-on-one conversation or group conversation through text. The user can send and receive messages only if they are connected to the same server [17]. To share one's knowledge and ideas, this chat box is integrated in this software.

3   Meeting Scheduler

The calendar displays the scheduled meetings along with time sheets. Since the calendar is visible publicly, the meeting scheduler [18] schedules the meeting according to the available slots. This is done using a conflict detection algorithm.

*Conflict Detection Algorithm*

$x$ = number of participants

$y$ = start time of meeting

$z$ = end time of meeting

$a = 1$

$b = 0$

while $a <= x$ do

while the start time and end time of the assigned meeting of every 'a' participant do

$c[start]$ = assigned meeting start time from database

$c[end]$ = assigned meeting end time from database

if$(y >= c[start]$ & $y <= c[end])$ or $(z >= c[start]$ & $z <= c[end])$ then

b = 1

break

a = a + 1

end if

end while

end while

if b = 1 then

Conflict

else

No Conflict

end if

The above pseudo code explains the algorithm that detects the freetime slots avoiding the conflict. This algorithm uses the participants list, meeting start and end time, and the meeting date to detect the conflict. For every participant in the meeting, the conflict is detected by the following steps. c[start] fetches the start time of all the scheduled meetings on the particular day from the database and stores it in an array. Likewise, c[end] fetches the end time of all the scheduled meetings from the database and stores it in an array. The currently scheduled meeting's start time is compared with each start time in the array c[start] and the end time is also compared in the same way. If there's any conflict, the process stops and indicates the user otherwise if no conflicts are detected, the meeting gets scheduled at the given time slot.

## 4 Email Notification

Once the meeting is scheduled, a notification is sent to the assignee through email. The email sent includes the meeting details, organizer details, and the participants list. It also sends a reminder 5 min prior meeting. This is done using Javax Mail Server (JMS) API. JMS is a platform and protocol-independent framework which uses SMTP (Simple Mail Transfer Protocol) for sending and receiving mails.

## 5 Content Share

During meeting, the participants can share their screen to make their contents visible to their team members. Even the participants can have control on their screen once the request is granted. Meeting notes can be generated date and time-wise along with the name of the meeting. The notes are saved in the database in encrypted form using the Base64 algorithm. The user can download and have the notes in their local system.

*Base64 Algorithm*

This algorithm deals with encryption in Java. RFC 4648 alphabet is used for encoding the text and RFC 2045 for decoding the text. The encoding text should be 76 characters per line. It has many methods to encrypt and decrypt the data. For accessing these methods, java.util.base64 must be imported in the file.

## 6 Meeting Recordings

The entire meeting session can be recorded in mp3 or mp4 format. If the user wants to refer to any topic or if he's not available at that time, he can refer to the recordings so that no knowledge is lost. It becomes easier and is time efficient. The recording will be saved once the meeting is ended which will be accessible to everyone only for a particular period of time. They can download it for longer use.

## 7 Storyboard

The storyboard automatically shifts the task status based on the user action. If the assigned task is viewed, the status will be shown as Opened. If the task is sent for prior review, it will be shown as In Progress. If it is submitted, the status will display Completed and if the task is not at all viewed, status will be Nil. This saves time by giving an overview of the project status. The chart displays the progress and estimated time to complete the task within the particular sprint.

## 8 Task Scheduler

This application allows users to quickly assign and update tasks according to the client's requirements. It uses a priority [19] based algorithm which sorts the task based on its priority. Once the task is assigned, a reminder is sent through email along with the task details which include task name and description, priority of the task, assigner name, and task deadline. If the task is not completed on time, a notification is sent automatically stating it as incomplete. The assigner can also modify the assigned task at any time. The modification will also be notified to the assignee.

*Priority-Based Sorting Algorithm*

PQ-Sort(T, A)

*INPUT* –task list T, priority comparator A for the tasks of T

*OUTPUT* – task list T sorted in increasing order according to A

PQ = priority queue with comparator A

while ¬T.isEmpty ()

r = T.remove (T.first ())

PQ.insert (r, Ø)

while ¬PQ.isEmpty()

r ← PQ.removeMin().getKey()

T.addLast(r)

The above pseudo code explains the algorithm that stores the tasks priority-wise. The priority is assigned by the following steps. If a new task is assigned, the priority of the scheduled task is compared with the priorities of every task in the database. In the comparison, if the priority of the newly added task is relatively high, the task gets inserted in the first place. Likewise, based on the priority weightage, all the tasks are sorted accordingly.

## 4 System Prototype

This system is a web-based application with responsive behavior. It uses markup language, CSS, JavaScript as front end technology and MySQL database as back end technology as it stores large amount of data [20] instantly and effectively. This software is integrated with JMS (Java Mail Server) API for sending and receiving emails.

In this application, various algorithms have been used. Base64 algorithm is used for encrypting and decrypting before storing in the database. Priority-based algorithm is used for displaying the task priority-wise. A conflict detection algorithm is used for detecting conflict to schedule the meeting without any conflicts (Fig. 1).
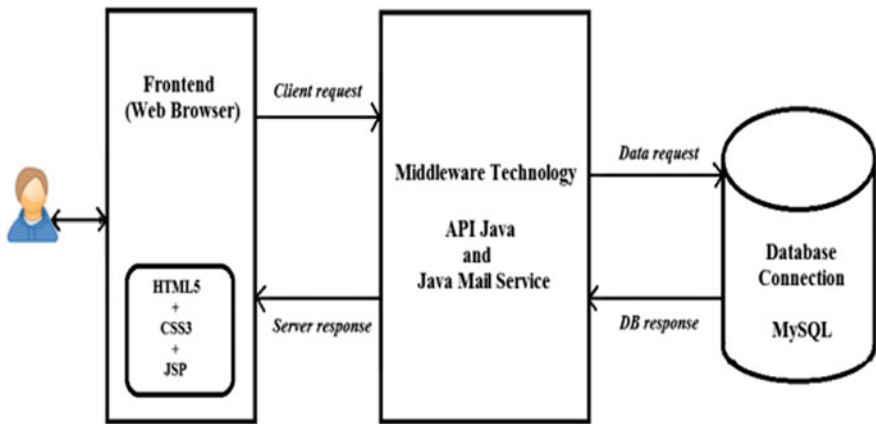


**Fig. 1** Architecture diagram

# 5 Result and Discussion

Let us discuss the result of few features and discuss the working function of each feature and their relationship.

Figure 2 explains the function of the task scheduler. If the user wants to add a task, he must fill in all the task details so that all the team members can know about them. Once he clicks the 'ADD' button, the task will be added and the details will be displayed in the task tab of both the assignee and assigner.

If the assigner wants to notify the assignee through the mail, he can send it using 'SEND NOTIFICATION' button. Once the mail is sent, a success alert box will be popped up. The assignee work progress can be monitored by the assigner at any time. Also a notification is sent after the task deadline gets expired.

Figure 3 shows the page on which the assigner can update or delete any assigned task. The assigned task details are fetched from the database and shown in this tab. After any modification, the assignee is notified through email.

Figure 4 displays all the tasks in a particular sprint. It is visible to all the members of the team. All the task details are fetched from the database and shown in this tab.

Figure 5 shows all the meetings scheduled for a day. The start and end time of the meeting is displayed here. It works using Java Mail Server (JMS) API. Using the login mail ID, the API filters the meetings and displays them here.



**Fig. 2** Task scheduler

**Fig. 3** Modify task



**Fig. 4** Action items

**Fig. 5** Scheduled meetings

Figure 6 shows the meeting notes taken during the meeting. The user can save the notes for later use. These are stored in the database in encrypted form. They can even download the notes and have them in their system.



**Fig. 6** Meeting notes

# 6    Conclusion

This system manages the work routine effectively in any organization. It helps in improving the productivity of the organization. After analyzing the related works in existing papers, we have added new features which increase the quality of the software. The added features benefit the following to this paper.

– At the end of each sprint, using the obtained feedback we can improve the task accordingly.
– Transparency in the work status so that monitoring becomes easy and the work will be completed effectively.
– This system supports high-quality and on-time deliveries.

# References

1. Špundak M (2014) Mixed agile/traditional project management methodology–reality or illusion? Procedia-Soc Behav Sci 119:939–948
2. Gandomani TJ, Tavakoli Z, Zulzalil H, Farsani HK (2020) The role of project manager in agile software teams: a systematic literature review. IEEE Access 8:117109–117121
3. Dhanalakshmi R, Sri DT (2020) Adaptive cognitive intelligence in analyzing employee feedback using LSTM. J Intell Fuzzy Syst 39(6):8069–8078
4. Alencar LH, Carneiro LB, Silva ACC (2018) Scrum agile project management methodology application for workflow management: a case study. In: 2018 IEEE international conference on industrial engineering and engineering management (IEEM). IEEE, pp 938–942
5. Mihalache A (2017) Project management tools for agile teams. Informatica Economica 21(4):85–93
6. Sugihara K, Kikuno T, Yoshida N (1989) A meeting scheduler for office automation. IEEE Trans Softw Eng 15(10):1141–1146
7. Wang X, Maurer F (2008) Tabletop Agile planner: a tabletop-based project planning tool for agile software development teams. In: 2008 3rd IEEE international workshop on horizontal interactive human computer systems. IEEE, pp 121–128
8. Özkan D, Mishra A (2019) Agile project management tools: a brief comprative view. Cybern. Inform. Technol. 19(4):17–25
9. Liechti O, Pasquier J, Reis R (2017) Beyond dashboards: on the many facets of metrics and feedback in agile organizations. In: 2017 IEEE/ACM 10th international workshop on cooperative and human aspects of software engineering (CHASE). IEEE, pp 16–22)
10. Schwaber K (2004) Agile project management with Scrum. Microsoft Press
11. Hu ZG, Yuan Q, Zhang X (2009) Research on agile project management with scrum method. In: 2009 IITA international conference on services science, management and engineering. IEEE, pp 26–29
12. Praveen Joe IR, Varalakshmi P (2019) An analysis on web-service-generated data to facilitate service retrieval. Appl Math Inform Sci 13(1):47–55
13. Ortu M, Destefanis G, Kassab M, Marchesi M (2015) Measuring and understanding the effectiveness of JIRA developers communities. In: 2015 IEEE/ACM 6th international workshop on emerging trends in software metrics. IEEE, pp 3–10
14. Subhiyakto ER, Astuti YP (2019) Design and development meeting schedule management application using the RAD method. In: 2019 international conference of artificial intelligence and information technology (ICAIIT). IEEE, pp 60–64

15. Souidi S, Boccio D, Mierzwa S, Aguilar J (2015) The feasibility of using Microsoft Azure infrastructure for a monitoring and evaluation system solution in sub-Saharan Africa. In: 2015 IEEE global humanitarian technology conference (GHTC). IEEE, pp 226–232

16. Korimbocus MA, Towokul TS, Nagowah SD (2019) Scrum meeting tool for knowledge capture and sharing. In: 2019 conference on next generation computing applications (NextComp). IEEE, pp 1–6

17. Siva Rama Rao AVS, Dhana Lakshmi R (2017) A survey on challenges in integrating big data. In: Deiva Sundari P, Dash S, Das S, Panigrahi B (eds) Proceedings of 2nd international conference on intelligent computing and applications. Advances in intelligent systems and computing, vol 467. Springer, Singapore. https://doi.org/10.1007/978-981-10-1645-5_48

18. Sugumaran M, Easwarakumar KS, Narayanasamy P (2003) A new approach for meeting scheduler using A*-algorithm. In: TENCON 2003. Conference on convergent technologies for Asia-Pacific region, vol 1. IEEE, pp 419–423

19. Anandhakumar P, Raj GD, Rajendran T (2013) Efficient priority queue algorithm and strainer mode technique for identification and eradication of duplications in XML records. In: 2013 fifth international conference on advanced computing (ICoAC). IEEE, pp 106–113

20. Patil PS, Patil SR, Rao S, Patil SB (2012) Customised approach for efficient data storing and retrieving from university database using repetitive frequency indexing. In: 2012 1st international conference on recent advances in information technology (RAIT). IEEE, pp 511–514

# Deep Learning Based Hybrid Approach for Crowd Anomalous Behavior Detection

**Aniruddha Prakash Kshirsagar and L. Shakkeera**

## 1 Introduction

With the rapid development of information and technology, surveillance video system has been widely used in public like highways and stations, and a large amount of abnormal activities has been recognized and analyzed in video data. In actual application, it is a significant direction to recognize various actual scenes with high accuracy and missing report rate. It is necessary to study the recognition method in video based on deep learning, which is helpful to reduce the safety hidden trouble caused by abnormal activities [1].

Computer vision [2] and other methods have been used to recognize the abnormal activity. At present, existing researches mainly combine human and intelligent video surveillance to monitor and warn against abnormal activity. Manual recognition is still the main method and is supplemented by automation and information technology, thus the standard of abnormal activity recognition needs to be improved. Because the SVM network model can accurately describe the semantic characteristics of video time series changes, and is suitable for identifying abnormal activity with relatively long intervals and delays in videos. So, the SVM network can be used to perceive the semantic characteristics of abnormal activities in videos, which is conducive to the early recognition of hidden security problems and effectively alleviating the problems caused by manual recognition. Dubey et al. [2] proposed a method based on the combination of trajectory and pixel analysis to measure the velocity and direction of the moving target trajectory and realized the recognition of abnormal activity through a clustering algorithm. The accuracy of trajectory feature extraction has a great influence on the result and is not applicable to video data with many noises. AI and deep learning are ideas that are regularly covered. There can be a slight disarray between the terms, Machine learning utilizes a bunch of calculations

A. P. Kshirsagar (✉) · L. Shakkeera
School of Computing Science and Engineering, VIT Bhopal University, Madhya Pradesh, India
e-mail: anipk2007@gmail.com

to dissect and decipher the information, gain from it, and in light of the learnings, settle on the most ideal choices. Then again, deep learning structures the calculations into various layers to make a "fake neural organization". This neural organization can gain from the information and settle on shrewd choices all alone.

## 1.1 Deep Learning

Customary AI strategies will in general capitulate to ecological changes while profound learning adjusts to these progressions by steady criticism and work on the model. Profound learning is worked with by neural organizations which mirror the neurons in the human cerebrum and installs numerous layer design (few noticeable and few covered up). It is a high-level type of AI, which gathers information, gains from it, and enhances the model. Regularly a few issues are mind boggling to the point that it is essentially outlandish for the human cerebrum to understand it, and subsequently programming it is an unrealistic idea. Crude types of Siri and Google Aides are a fitting illustration of customized AI as they are found compelling in their modified range. However, Google's profound psyche is an extraordinary illustration of profound learning. Profound learning implies a machine, which learns without anyone else through numerous experimentation strategies. Frequently a couple hundred million times.

## 1.2 Existing Approaches

In article [3], Sultani et al. combined histogram, PHOG and HMOEOF features to recognize abnormal activity through SVM. However, their method requires an amount of calculation and the final classification accuracy needs to be improved. Kavikuil and Amudha [4] proposed an anomaly activity recognition model based on the AlexNet network, but the imbalance of recognized data is an important factor that affects the algorithm's training feature.

Compared with the mentioned methods, the extracted feature's quality was affected by data noise, the video sequence information utilization rate is low, and poor classification results a multiple feature fusion based on CNN and SVM abnormal activity recognition method was proposed and introduced the attention mechanism [5] to SVM, then analysis the correlation between the features, which can effectively extract features to reduce the long sequence information and the information shortage.

## 2 Proposed Convolutional SVM Approach

A. *Activity representation with deep learning models*

We transform the issue of abnormal activity recognition into an outlier recognition problem of space–time sequence, and the output is divided into two types: normal and abnormal activities. The spatial–temporal features were extracted by CNN and SVM. SVM can effectively avoid long-term dependence problems, and the gradient will not disappear after time back-propagation training [6]. In addition, attention mechanism was introduced to effectively analyze the correlation between model input and output, avoiding the influence of background noise and long sequence, to obtain more information (Fig. 1).

Each frame is the input of CNN model for convolution operation in the video, and finally a 2048-dimensional feature vector $\mathbf{C_r}$ will be chosen as spatial output through the fully connected layer for transmission to the SVM Attention layer.

B. *SVM attention models*

In sequential tasks, it is critical to learn the time dependence between the inputs. As a special time recurrent network, SVM obtains higher level information by stacking together [7]. The cell structure is shown in Fig. 2.

SVM network is controlled and updated by input gate $i_t$, forgot gate $f_t$, and output gate $o_t$, where there is an input, if $i_t$ is activated, its information will be stored in the cell. Also, if $f_t$ is turned on, the unit state $c_t$ is forgotten. The latest feature in the fully connected layer. Next, the 1 * n-dimensional feature vectors are feeding into the SVM unit output $c_t$ is determined by whether $o_t$ is propagated to the final state $h$.
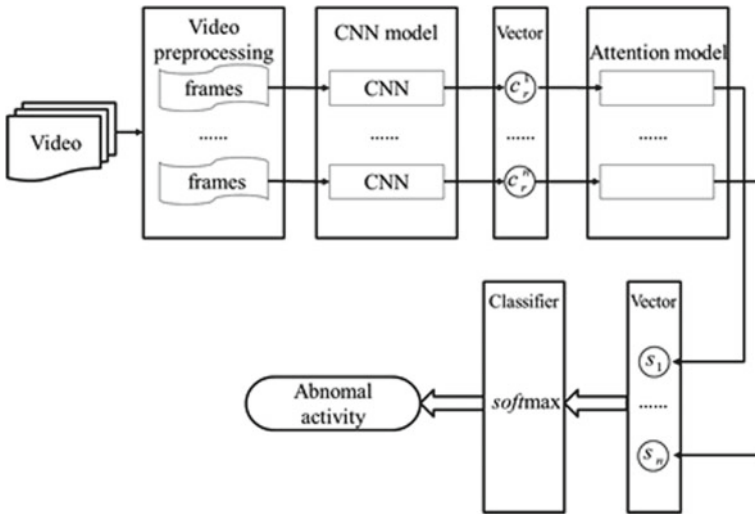


**Fig. 1** Method of abnormal recognition-based CNN SVM attention models
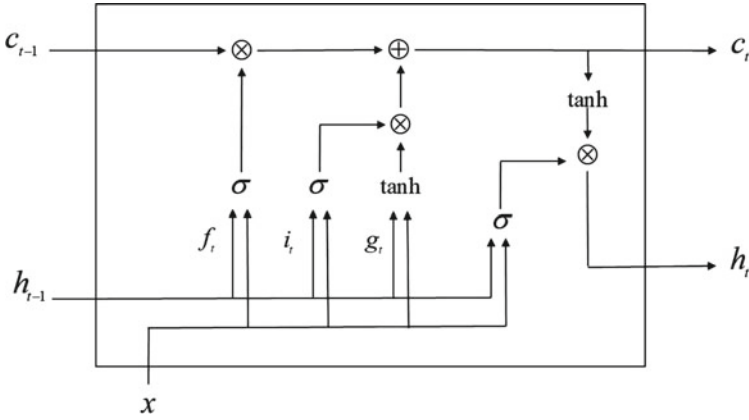
**Fig. 2** SVM cell structure

The state of each cell can be expressed by the attention model to train the time series features. The attention mechanism can distinguish key features from the hidden state output of the SVM layer.

C.  *CNN models*

The essence of CNN is to extract the visual features between data through convolution and pooling operations, and the extracted features will become more and more abstract with the increase of the number of layers, and finally converge at the full connection layer. Due to the good performance in the process of feature extraction, we chose inception-v3 model to extract features that are different from traditional CNN models, it convoluted images through different convolution verification operations, and then combined different convolution layers in parallel. The dataset used in the experiment is a publicly dataset UMN [8] with a resolution of 320 * 240, and it contains normal and abnormal activity in the crowd. Dataset contains 11 videos in 3 scenes where some people walking normally and suddenly running after some time, and all video scenes are taken in a permanent position with a static background. We trained on a normal section of 5 videos of all scenes and tested on all videos. The experimental parameters settings: experiment SVM super parameter of the model is obtained by cross-validation, and using the Singh and Mohan [9] optimization neural network model, it can weight vector update and set up according to the model, using the batch size of 64, every time training for the whole is represented by feature vector after pooling layer. CNN as input of SVM network, output vector used in this experiment single-layer SVM network and the attention of the input layer, in the attention layer, to compute the weight vector, and then the weight vector and the input vector to merge the current layer, a new vector **s** and as a weighted vector and all of the time step characteristics, its overall structure is shown in the Fig. 3.
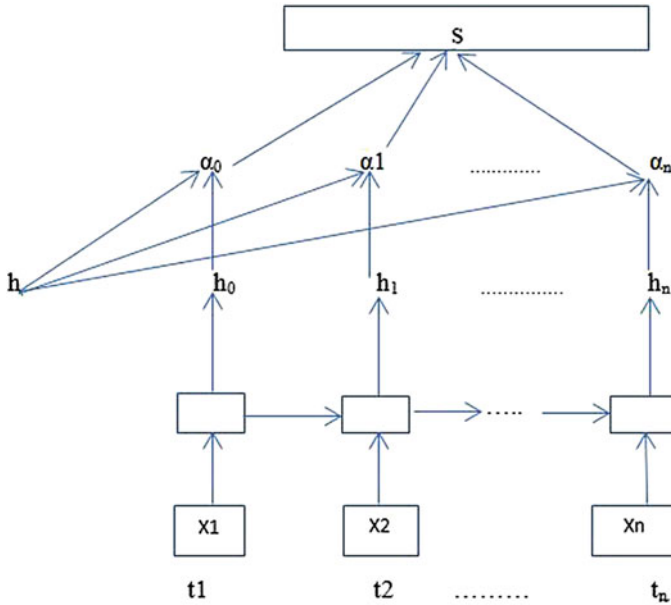
**Fig. 3** The model of SVM attention

## 2.1 C-SVM Working

After the multiple steps like data selection and data preprocess, the proper sorted and clean data are used for extracting the unused feature in the proposed approach which can be done by the SVM approach by considering the dataset into the vector by removing the grid of images. In the proposed hybrid approach with have used the combination of CNN and SVM so can find exact abnormal activity through a proper approach with automatic indication. Following some steps gives the detail overview of the approach.

1. New Trainingset {xi, yi, i=1…l+1}
2. New Coefficients qi, i=1…l+1
3. New Bias b
4. New Training Set Partition
5. New R Matrix

The output contains all the values given in the input updated. In the above steps, we have created and verified the dataset by maintaining the dataset and properly patinating the data to check the abnormal entry. The output contains all values given in the input updated.

**Forgetting Algorithm**

IF(c $\in$ REMAINING SET)
REMOVE SAMPLE FROM REMAINING SET
REMOVE SAMPLE FROM TRAINING SET
EXIT
IF(c $\in$ SUPPORT SET)
REMOVE SAMPLE FROM SUPPORT SET
IF(c $\in$ ERROR SET)
REMOVE SAMPLE FROM ERROR SET

In the above step check whether the normal activity is available in movement or not by cross-checking the activity with the store dataset if it's available and the activity is normal then it works as it is or is detected as abnormal activity in a particular area the following section gives the experimental analysis of real-time video.

## 3    Experiment and Analysis

A.   *Experiment environment and dataset*

The experiment used Python to program and TensorFlow for the training model, and the Python version is 3.6. Anocanda3 is used to build the experimental model in the Linux operating system server version Ubuntu 14.

B.   *Results and analysis*

The accuracy represents the proportion of samples correctly classified in all classifications. The precision rate indicates how many of the predicted samples (such as positive samples) actually samples of a certain type. Recall is how much of a sample is correctly predicted.

For Prediction and Accuracy
Algorithm 1 (Predict,Accur)=C-SVM(Train,Div,Test,Test-Final, ϴ)
ϴ=Termination condition Ensure: Predict->Predicted sentiment output
Accur->Accuracy
1. Net->Create Network
2. Network_initialize(Net)
3. for error>=do
4. error Network_Train(Net,Train,Div)
5. end for
6. /*Training completed*/
7. Featureopt->C-SVM(Train,Div)
8. HTrain->GetTop_HiddenLayer(Net,Train)
9. Train_combined<-HTrain + Featureopt
10. ModelSVM<-SVMLinear(Traincombined)
11. HTest<-GetTop_HiddenLayer(Net,Test)

**Table 1** Comparison of experiments

| Method | Accuracy (%) | Precision (%) | Recall (%) |
|--------|--------------|---------------|------------|
| CNN    | 83.13        | 76.63         | 99.32      |
| SVM    | 89.41        | 83.23         | 96.77      |
| C-SVM  | 94.30        | 89.24         | 96.61      |

12. Testcombined<-HTest + Featureopt
13. Predict<-SVMLinear(ModelSVM,Testcombined)
14. Accur<-Evaluation(Test-Final,Predict)
15. return(Predict,Accur)

The comparison of proposed approach with existing method in Table 1.

The result in Table 1 shows that the accuracy obtained from UMN dataset is higher than the other three existing methods. Figure 4 shows that the initial loss value caused by the increase in complexity increases and the convergence speed is fast after the attention mechanism is introduced into the SVM network during model training.

Figures 5, 6, 7 and 8 indicate that the attention mechanism improves the prediction of final classification to some extent. By introducing the attention mechanism into the hidden layer of the SVM network, feature loss caused by a long sequence can be effectively solved and important features can be highlighted, thus improving the performance of the model.
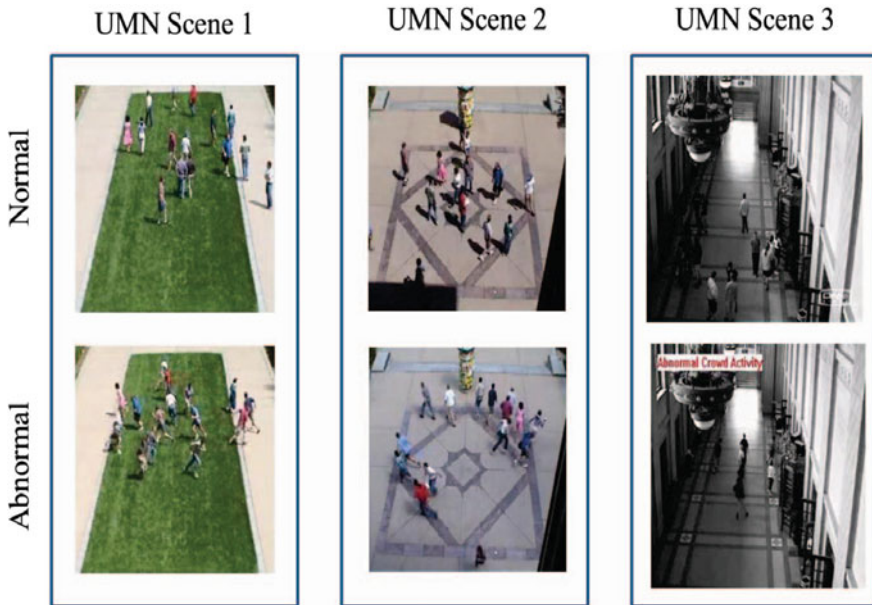


**Fig. 4** Normal and abnormal activity in dataset
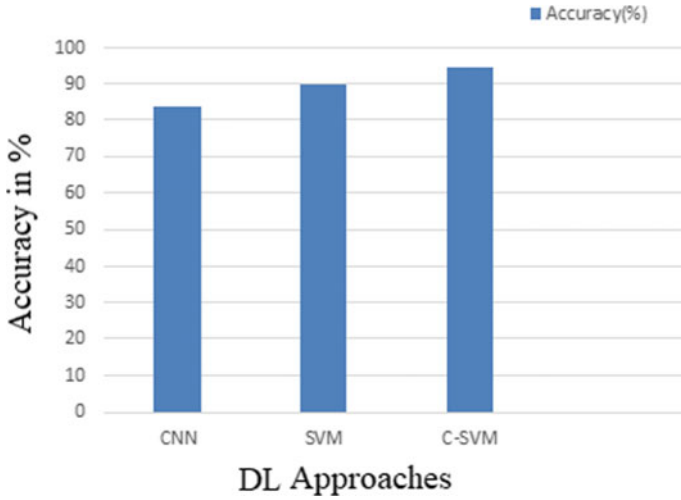
**Fig. 5** Analysis on basis of accuracy
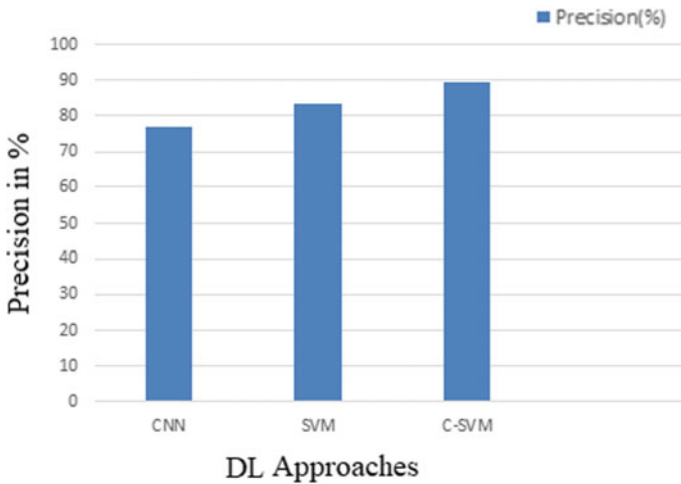


**Fig. 6** Analysis on basis of precision

## 4   Conclusion

We proposed a new method of abnormal activity recognition that applies the deep learning and attention mechanism to the issues of recognition successfully. Experimental result shows that the proposed method has been tested on the **UMN dataset** and outperforms the existing used methods, which proves the efficacy of the proposed method. The proposed approach cannot only fully extract the deep features of video
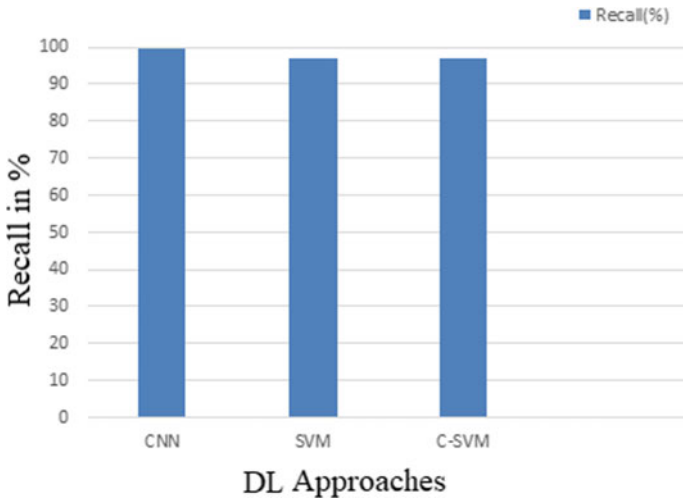
**Fig. 7** Analysis on basis of recall
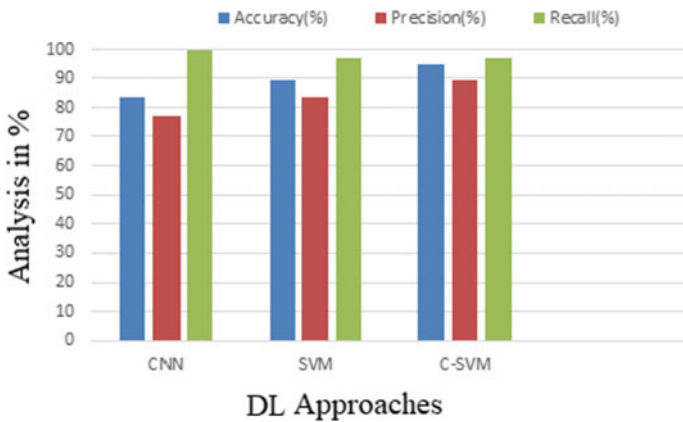


**Fig. 8** Overall performance analysis of the proposed system

frames, but also focus on behavioral features that have a greater impact on results. So, it has a greater potential compared with common deep learning and traditional manual feature extraction methods. However, due to the large amount of calculation, real-time performance of this method is difficult to be applied to the multi-channel recognition system with high real-time requirements, this will be the focus of our future research.

# References

1. Amrutha CV, Jyotsna C, Amudha J (2020) Deep learning approach for suspicious activity detection from surveillance video. 978-1-7281-4167-1/20/$31.00 ©2020 IEEE
2. Dubey S, Boragule A, Jeon M (2020) 3D ResNet with ranking loss function for abnormal activity detection in videos. IEEE
3. Sultani W, Chen C, Shah M (2019) Real-world anomaly detection in surveillance videos. In: Computer vision and pattern recognition (CVPR)
4. Kavikuil K, Amudha J (2019) Leveraging deep learning for anomaly detection in video surveillance. Advances in intelligent systems and computing
5. Pang H, Li H (2018) Intelligent detection simulation for crowded pedestrian abnormal behavior. Comput Simul 35:405–408
6. Cosar S, Donatiello G, Bogorny V, Garate C, Alvares LO, Bremond F (2017) Toward abnormal trajectory and event detection in video surveillance. IEEE Trans Circ Syst Video Technol 27:683–695
7. Mnih V, Heess N, Graves A. Recurrent models of visual attention. In: Advances in neural information processing systems, Montreal, pp 2204–2212
8. Ding L, Fang W, Luo H, Love PED, Zhong B, Ouyang X (2018) A deep hybrid learning model to detect unsafe behavior: integrating convolution neural networks and long short-term memory. Autom Constr 86:118–124
9. Singh D, Mohan CK (2017) Graph formulation of video activities for abnormal activity recognition. Pattern Recogn 65:265–272

# Quantum Machine Learning Algorithms for Diagnostic Applications: A Review

**Shruti S. Pophale and Amit Gadekar**

## 1 Introduction

Machine Learning (ML) is a fast growing discipline of computer discipline, powered by the huge amounts of data sent, saved, and analyzed every day [1]. In the real world, there is no shortage of machine learning and quantum computing applications and implementations. They are fascinating fields that cover a wide range of topics and have several real-world applications. Quantum computing with machine learning is, in reality, the fastest-growing field nowadays. Quantum computing procedures carry new concepts and methodologies to machine learning problems. Quantum computing is concerned with the creation of computer-based technologies constructed on the designs of quantum theory. Quantum theory describes the nature and comportment of energy and matter at the quantum level. Quantum computing is an assembly of bits that collaborate to solve issues. Quantum Machine Learning (QML) is an area that chains quantum physics and machine learning (ML). Quantum computing is utilized to construct quantum forms of ML algorithms, and standard ML algorithms are used to study quantum systems in a symbiotic relationship. The area of machine learning and deep learning has seen tremendous growth in recent years. Models created using these methods have been used in practically every industry imaginable, including military, aerospace, agriculture, banking, and healthcare [2]. Machine learning algorithms are tasked with retrieving important data from new samples and making predictions.

These algorithms, unlike all other mathematical techniques, develop and update their predictive model using information that is already available (training dataset).

S. S. Pophale (✉)
Department of CSE, Sandip University, Nashik, Maharashtra, India
e-mail: shruti.pophale@gmail.com

A. Gadekar
School of Computer Science and Engineering, Sandip University, Nashik, Maharashtra, India
e-mail: amit.gadekar@sandipuniversity.edu.in

Spam filtering, image processing, societal effect, picture identification, and signal processing are just a little of the jobs that machine learning approaches may help with. Many improvements in quantum data processing have been made in current years, indicating that certain quantum computing algorithms can outperform their classical analogs [3].

Quantum computers have been exposed to be proficient in handling such issues successfully with current technologies by processing several states concurrently. They exploit the three fundamental aspects of quantum physics like superposition, entanglement, and interference. Due to their genetic features (superposition), extreme correlation even when separated by great spaces (entanglement), and bias toward the preferred state, qubits (the essential unit of quantum computer) is capable in several states at the identical time (interference). As an outcome, quantum computing has the possibility to move science closer to artificial general intelligence [2]. The idea of a quantum bit underpins quantum computations (qubit). A qubit is a quantum type of a binary bit that is created using a two-state device. A qubit's measurement has two possible outcomes, similar to a binary digit: 0 or 1. While a classical bit's state can only be 0 or 1, a qubit's common state in quantum physics can be any logical superposition of both, allowing computation on both 0 and 1. This occurrence is called as quantum parallelism.

To resolve optimization problems, traditional algorithms use classical models and quantum machine learning algorithms, co-generative models of quantum machine learning algorithms, quantum-based support vector machines algorithms, quantum collaborative K-means, quantum computing, and diabetes guess using ML algorithms and DL algorithms. Quantum versus classical computation is used to apply machine learning to a diabetes dataset. It's also being studied in the research area of natural language processing (NLP), specifically, a machine translator for English to Hindi founded on quantum neural networks. All described methods investigated were used. Qubit states ranging from 3 to 5 qubits, and the computational complexity were compared to the classical approach. Quantum computers with more accessible qubits will be allowed to learn on larger datasets and achieve even higher accuracies. Many articles compared the results of classical machine learning algorithms and QML algorithms with deference to many characteristics such as accuracy rate, time complexity, and processing speed.

## 2 Literature Survey

Quantum operators must be linear, and finding alternate activation function solutions is a major research topic. Nonetheless, it has significant advantages over traditional models. A quantum network's storage capacity is increased exponentially by quantum superposition. Quantum parallelism can be used to train a neural network with multiple inputs at the same time. Despite the fact that quantum neural networks are a new field; many real-world applications have been proposed [4]. Quantum-based support vector machine identical to the classical model has contributed to discuss

quantum fundamentals, quantum algorithms, and then compare and contrast algorithmic growth with classical equivalent quantum support vector machines and their training algorithms [5]. Using quantum simulators or actual quantum computers, quantum computing can be used to solve a variety of machine learning and other optimization problems more efficiently. The topics and experiments presented here demonstrate quantum computers' suitability for a wide range of machine learning and deep learning applications [6]. Different specialized commitments, qualities, and similitudes of the examination work in this space were talked about, withdrawing from central ideas of AI and quantum registering. It additionally examines ongoing advances in quantum AI draws near, their intricacy, and applications in fields like material science, science, and normal language handling [7]. Deep reinforcement learning was investigated using variational quantum circuits. Exemplary profound support learning calculations, for example, experience replay and target network were reshaped into a portrayal of variational quantum circuits. Besides, when contrasted with old-style neural organizations, utilize a quantum data encoding plan to diminish the quantity of model boundaries [8]. The performance of a novel quantum neural network (QNN) training algorithm based on a variation of the quantum search algorithm was intensively researched in the literature. With a sublinear cost, the proposed algorithm trains a QNN to explore all possible weights. Initial experimental results show that the algorithm always computes to existing solutions, as well as having mean and maximum values that are almost always less than the theoretical maximum amount. Classification problems are solved using the training algorithm [9]. Researchers focused that groundbreaking thoughts that are not needed in the old style and probabilistic developers: (1) utilizing the essential Gleason's hypothesis in quantum mechanics to coordinate the selections of formats of LRSMs; and (2) a summed up Farkas' lemma as far as observables (Hermitian administrators) in quantum material science. The past shows an intriguing relationship between LRSMs and the exhibition of conditions of quantum frameworks [10]. Quantum Computing is advantageous to deal with training-related computational issues, MS (model selection), and EE (error estimation) phases in ML. Furthermore, the same authors demonstrated how QC can inspire new theoretical approaches that are not possible with the traditional computation paradigm [11]. A heuristic and an exact template matching algorithm were presented in the literature for finding new quantum templates with more than 3 qubits [12]. A generative adversarial learning algorithm for learning a latent variable generative model is replaced by samples from a graphical model is proposed in [13]. A Boltzmann machine is used to learn this graphical model, and feature representations of data extracted by the discriminator Sheng and Zhou [2]. Distributed secure quantum machine learning (DSQML) enables a traditional client with limited quantum technology to delegate remote quantum machine learning to a quantum server while maintaining privacy data is discussed in [14]. The client–server protocol is the first DSQML protocol, and the client–server-database protocol is the second, which is more general and practical [14]. The research focused on the quantum version of the k-NN algorithm, which allows researchers to understand the fundamentals of converting classical machine learning algorithms to quantum versions [15]. Overall quantum calculation proposed

for AI is dependent on a quantum generative model by the authors in [16]. It is shown that the proposed model is more equipped for addressing likelihood circulations than traditional generative models and that it has outstanding learning and induction speedups, basically sometimes, when a quantum PC can't be effectively mimicked traditionally. The outcome unites devices from different fields to shape a captivating connection between the quantum many-body physical science quantum computational intricacy hypothesis, and the outskirts of AI [16]. Authors in [17] have proposed quantum-inspired information retrieval (QIR) as one such region, where the quantum theory (QT) numerical system is used to build up IR representation and user models required to more readily line up with human intellectual data preparation. It differs from quantum computing in that it does not depend on physical quantum states to perform computations [17]. The discoveries in the investigations in Maheshwari et al. [18] are (i) Gradient-based analyzers discover marginally preferred arrangements over angle free streamlining agents, and the versatile learning rate or inclination technique likewise helps half-breed quantum old-style models perform better. (ii) Even when the quantity of emphases is more noteworthy than that of slope-based enhancers, angle free analyzers regularly utilize less calculation time until they combine. (iii) COBYLA, a slope free based streamlining agent, has the best reasonable exhibition in limited scope variable space order assignments [18]. Quantum versus classical implementation of Machine learning (ML) algorithm applied to a diabetes dataset in Ding et al. [19]. A different approach to comprehending the disease's complex patterns was discussed. The proposed system addresses a binary classification problem in which patients with diabetes are divided into two groups: those with acute diseases and those without acute diseases [19]. A quantum SVM calculation was proposed to address the huge information challenge, professing to accomplish dramatic speedup for least squares SVM (LS-SVM). At the point when a low-position estimation is suitable, the calculation functions admirably on low-position datasets or datasets that can be all around approximated by low-position grids, which is like the quantum SVM calculation [20]. Small quantum computers, as well as larger special-purpose quantum simulators, annealers, and other devices, appear to have potential applications in machine learning and data analysis, according to reviews [21]. Perspectives for a mixed distribution of classical ML and quantum computation experts based on the literature in quantum ML has reviewed [22]. The limitations of quantum algorithms will be clarified, as well as how they compare to their best classical counterparts and why quantum resources are expected to provide advantages for learning problems [23]. A method for allowing cluster interaction based on the collaboration of several clusters has been proposed in Kerenidis et al. [24]. As a result, the dataset's underlying structures and regularities could be identified [24]. Clustering is a canonical problem in unsupervised machine learning, and a new quantum algorithm has been developed to solve it. Like k-means, the q-means algorithm guarantees convergence and precision, and it produces a good approximation of the K-cluster centroids with a high probability, just like the classical algorithm [25]. With the help of Adiabatic Quantum Learning, researchers have formalized the insights of Quantum Machine Learning and show that the algorithm's execution time is greatly reduced [26].

## 3   Limitations

It is found that in some papers work done on quantum simulators has been implemented instead of real-world quantum computers. However, extending the work on quantum computers may increase the performance. In some of the approaches, quantum 3 qubits were implemented which gives less accuracy in the results obtained. In most of the approaches, it is found that QML algorithms take more execution time that can be reduced with improved methodologies. Complexity measures and analysis are missing in some of the methods that can be rectified in improved approaches.

## 4   Conclusion of Literature Survey

The literature survey mainly focuses on four different domains where quantum computing is discovered in four flavors as shown in Fig. 1. Quantum computing is integrated with different advanced techniques to form the various approaches. Quanta computing with machine learning presents a new field as QML in which classical algorithms and quantum approaches are combined and interesting results are found out. The approaches are quantum support vector machine, quantum supervised learning, quantum unsupervised learning, quantum classified learning, quantum reinforcement learning. All approaches prove that the results obtained by the quantum computing approach are better as compared to the classical machine learning method. Quantum computing with deep learning has new advances in an artificial neural network as quantum deep learning, Quantum natural language processing, mode quantum artificial neural network, quantum neural network, global search for QNN and adiabatic algorithm.

These prove the more flexible and precise result oriented compared to the classical neural network and deep learning methods. Quantum computing with artificial intelligence and its applications laid down the new benchmark for the output acquired with modified methods in QML diabetic detection, English to Hindi translation, Game semantic, non-algorithmic, and deep-learning diabetic approach. In the final category, the quantum computing emerges to solve optimization problems such as optimization for quantum machine learning, a generative model for optimization, template matching, quantum theory for information retrieval, and quantum circuit optimization.

In quantum machine learning, nearby are a number of issues that must be addressed on both the hardware and software sides. The two primary complexity measures for computational models are time and space. Because of the scourge of dimensionality, old-style frameworks with countless levels of opportunity are tough to display, however, the quantum parallelism impact permits us to stay away from this issue. Accordingly, quantum computational assets are amazingly helpful for tackling an assortment of issues with a surprisingly enormous number of measurements (AI, for instance). There is an assortment of ways to the transaction with information
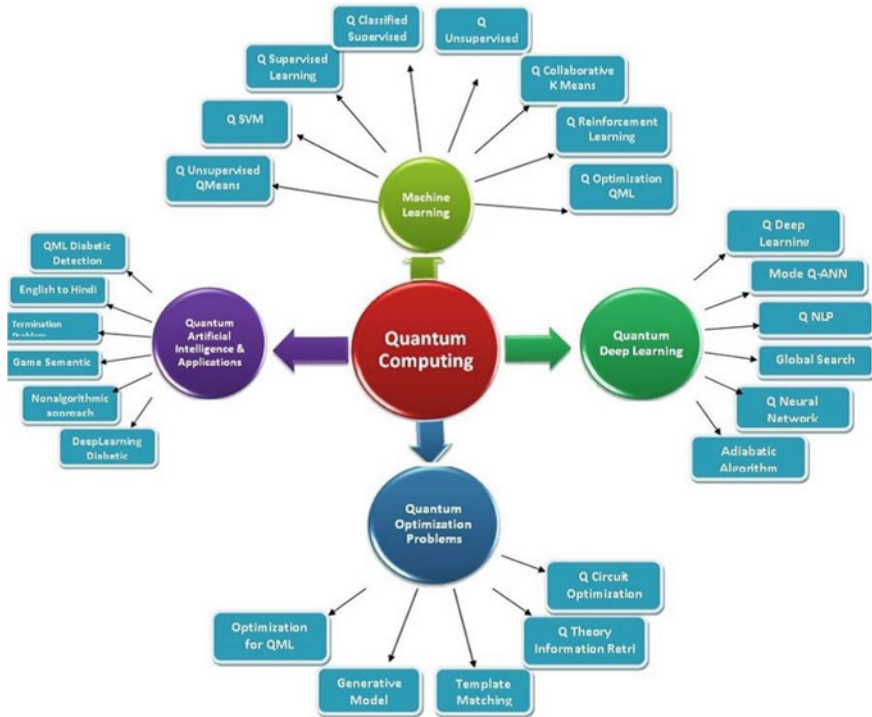
**Fig. 1** Approaches in quantum computing

handling and AI. Quantum approaches can use to both essential and modern strategies (k-closest neighbor calculation, k-implies calculation, head part examination, etc.) (variational autoencoders, acquainted ill-disposed organizations) [27]. Need for more research and challenges.

## 4.1 Need

The most common method of creating a quantum neural network appears to be the direct translation of classical neural network components into quantum physics language using some basic quantum properties like linear quantum gates, quantum unitary operators, quantum superposition, and quantum entanglement, to name a few. The issue with this method is that it introduces many edge cases, which can lead to erroneous quantum states. Additional complexity is introduced to deal with this, which may have an impact on network performance. Novel, non-traditional concepts are required to overcome the obstacles. The field is new and promising, but it will take a lot more research to realize its full potential in practice. Future research in this field will focus on applying these ideas to a real-world quantum

computer. When using quantum principles and algorithms to develop algorithms and error correction processes, one can concentrate on reducing the contents of the data matrix, which can help to reduce time complexity. The following problems can be exactly solved using the provided algorithm. Traveling salesman, traffic optimization problem, quantum war game. With more accessible qubits, quantum computers will be able to learn on larger datasets and accomplish higher accuracy. These research and implementations have thus demonstrated the actual potential of quantum computers in enabling very efficient machine learning systems in the future. Quantum computing can help with high-temperature superconductor modeling, molecule selection for organic battery production, and drug modeling and testing, among other scientific challenges. Quantum computing can help with high-temperature superconductors modeling, organic battery molecule selection, and drug modeling and testing, among other scientific challenges.

## 4.2  Challenges

There are two ways to make the model improved: quantum terms may improve the learned latent variable generative model; or graphical models, which are traditionally difficult to train, such as fully connected, integrated into the latent spaces, may improve the learned latent variable generative model. Before devoting a significant amount of time and effort to integrating quantum models into latent spaces, consider the following. It's critical to note that these improvements are reflected in the overall model of the dataset. Backpropagation, in other words, does not cancel out any gains in latent space performance. In the future, the developed DL model will be tested on various diabetic datasets to ensure its robustness, and a user-friendly web application will be developed. In order to outperform existing models and state-of-the-art methods, the proposed QML model must be integrated with a deep learning framework. This discovery paves the way for the use of quantum computation to solve difficulties in ML and AI, which could have far-reaching implications in the future. In the face of uncertainty and vibrant changes in context, the QT framework excels at simulating human decisions. Combining them with deep neural networks can improve their ability to represent complex human behavioral data, laying the groundwork for more human-centered artificial intelligence. The output layer of a deep learning model can combine several statistical hypotheses remembered and learned by the hidden layer during training. Quantum superposition is similar to the nature of simultaneously reserving several hypotheses. This opens up some exciting future research possibilities.

## *4.3   Objectives of Proposed Research*

As a result, it is recommended that approaches be created to address the following challenges in order to improve accuracy, time complexity, and computational speedup. Enhancement can be expanded in a few factors to improve results when compared to present state-of-the-art techniques. The results of the presented algorithms can be acquired in real-time applications such as aviation system control, agricultural forecasting, stock market analysis, and retail market analysis. The purpose of the research will be to learn more about

1. To create a framework that will enable effective qubit quantum machine learning for i. accurate diagnostic decision-making and ii. Deal with dynamic decision-making scenarios.
2. This study will look into a number of difficult challenges in the diagnostic field.
3. For applications such as aircraft control systems, this research will undoubtedly be useful from a social, economic, and technical standpoint.
4. It will produce a beneficial solution to the problems and may lead to the discovery of new research areas.

## 5   Conclusion

In this paper, we reviewed quantum machine learning, quantum computing with deep learning quantum natural language processing, quantum neural network, global search for QNN, and adiabatic algorithm. These algorithms can be applied to many real-world applications in medical, healthcare, flight delay prediction, share market analysis, and agriculture field. In the medical field for cancer detection, Covid-19 detection quantum machine learning can be applied. In agriculture field for crop disease identification, crop prediction quantum machine learning can expose. Quantum AI can possibly alter software engineering. It can possibly accelerate data handling much past what is as of now conceivable. Quantum calculations have as of late been fostered that could be utilized as establishments for AI applications. Notwithstanding its gigantic guarantee, quantum AI has generous equipment and programming impediments that should be defeated before it becomes functional. Quantum registering, as recently showed, can help in the arrangement of specific logical issues, for example, the displaying of high-temperature superconductors, the choice of atoms for the creation of natural batteries, and prescription demonstrating and testing. Quantum AI has various issues that should be settled on both equipment and programming levels. To start, quantum equipment should be sensible to receive the rewards of quantum calculations talked about in this survey. Second, to encode old-style data in the quantum mechanical structure, QML requires the joining of interface gadgets like qRAM. These equipment issues are not immaterial and should be tended to. At long last, to completely acknowledge QML methods, the limits of quantum calculations' application should be tended to. Quantum calculations have

four significant issues to manage: input, yield, cost, and benchmarking. There is right now minimal comprehension of the genuine number of entryways needed to execute a calculation in QML. Since these systems are only applied at that point, their reconciliation intricacy is likewise simply hypothetical. This likewise implies that foreseeing the viable productivity increment among quantum and traditional methodologies is troublesome.

# References

1. Ray S (2019) A quick review of machine learning algorithms. In: 2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon), pp 35–39
2. Gupta H, Varshney H, Sharma TK, Pachauri N, Verma OP (2022) Comparative performance analysis of quantum machine learning with deep learning for diabetes prediction. Complex Intell Syst 8:3073–3087
3. Shah A, Shah M, Kanani P (2020) Leveraging quantum computing for supervised classification. In: Proceedings of the international conference on intelligent computing and control systems (ICICCS 2020). IEEE Xplore, pp 257–261
4. Chakraborty S, Dasy T, Sutradharz S, Dasx M, Deb S (2020) An analytical review of quantum neural network models and relevant research. In: Proceedings of the fifth international conference on communication and electronics systems (ICCES 2020), IEEE Conference, pp 1395–1400
5. Uke D, Soni KK, Rasool A (2020) Quantum based support vector machine identical to classical model. In: IEEE conference, 11th ICCCNT 2020, July 1–3, 2020
6. Barabasi S, Tappert CC, Evans D, Leider AM (2019) Quantum computing and deep learning working together to solve optimization problems. In: 2019 international conference on computational science and computational intelligence (CSCI). IEEE, pp 493–498
7. Khan TM, Robles-Kelly A (2020) Machine learning: quantum vs classical. IEEE Access 8:219275–219294
8. Yen-Chi Chen S, Huck Yang C-H, Qi J, Chen P-Y, Ma X, Goan H-S (2020) Variational quantum circuits for deep reinforcement learning. IEEE Access 8:141007–141024
9. de Paula Neto FM, Ludermir TB, de Oliveira WR (2019) Quantum neural networks learning algorithm based on a global search. In: IEEE 2019 8th Brazilian conference on intelligent systems (BRACIS), pp 842–847
10. Li Y, Ying M (2018) Algorithmic analysis of termination problems for quantum programs. In: Proceedings of the ACM on programming languages, vol 2, no POPL, Article 35. Accessed January 2018
11. Oneto L, Ridella S, Anguita D (2017) Quantum computing and supervised machine learning: training, model selection, Anderror estimation. In: Chapter 2—Quantum computing and supervised machine learning, © 2017 Elsevier, pp 33–83
12. Mazder Rahman Md, Dueck GW, Horton JD (2014) An algorithm for quantum template matching. ACM J Emerg Technol Comput Syst 11(3), Article 31. Accessed December 2014
13. Wilson M, Vandal T, Hogg T, Rieffel EG (2021) Quantum-assisted associative adversarial network: applying quantum annealing in deep learning. Springer, Quant Mach Intell 3:19
14. Sheng Y-B, Zhou L (2017) Distributed secure quantum machine learning. Sci Bull 20 June 2017:S2095-9273(17)30325-0
15. Mezquita Y, Alonso RS, Casado-Vara R, Prieto J, Corchado JM (2020) A review of k-NN algorithm based on classical and quantum machine learning. In: International symposium on distributed computing and artificial intelligence, DCAI 2020: distributed computing and artificial intelligence, Special Sessions, 17th international conference, pp 189–198

16. Gao X, Zhang Z-Y, Duan L-M (2018) A quantum machine learning algorithm based on generative models. Sci Adv 4(12):eaat9004
17. Uprety S, Gkoumas D, Song D (2020) A Survey of quantum theory in-spired approaches to information retrieval. ACM Comput Surv 53(5):98:1–98:39, Article 98. Accessed September 2020
18. Maheshwari D, Garcia-Zapirain B, Sierra-Soso D (2020) Machine learning applied to diabetes dataset using quantum versus classical computation. In: 2020 IEEE international symposium on signal processing and information technology (ISSPIT)
19. Ding C, Bao T-Y, Huang H-L (2021) Quantum-inspired support vector machine. In: IEEE transactions on neural networks and learning systems. IEEE
20. Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N, Lloyd S (2017) Review: quantum machine learning, vol 549, pp 195–202. © 2017 Macmillan Publishers Limited, part of Springer Nature
21. Ciliberto C, Herbster M, Ialongo AD, Pontil M, Simone A-R, Severini, Wossnig L (2018) Quantum machine learning: a classical perspective, January 23, 2018. royalsocietypublishing.org
22. Huang H-Y, Broughton M, Mohseni M, Babbush R, Boixo S, Neven H, Mcclean JR (2021) Power of data in quantum machine learning. Nat Commun 12:2631
23. Benlamine K, Bennani Y, Grozavu N, Matei B (2020) Quantum collaborative K-means. In: 2020 international joint conference on neural networks (IJCNN), IEEE conference
24. Kerenidis I, Landman J, Luongo A, Prakash A (2019) q-means: A quantum algorithm for unsupervised machine learning. In: Advances in neural information processing systems (NeurIPS 2019), vol 32. arXiv: 1812.03584 [quant-ph]
25. Fastovets DV, Bogdanov YI, Bantysh BI, Lukichev VF. Machine learning methods in quantum computing theory. Quant Phys (quant-ph). arXiv: arXiv:1906.10175
26. Shrivastava P, Soni KK, Rasool A (2020) Classical equivalent quantum un-supervised learning algorithms. In: International conference on computational intelligence and data science, Science direct procedia computer science, vol 167, pp 1849–1860
27. Gupta S, Mohanta S, Chakraborty M, Ghosh S (2017) Quantum machine learning—using quantum computation in artificial intelligence and deep neural networks. In: 2017 8th annual industrial automation and electromechanical engineering conference (IEMECON), pp 268–274

# Minimizing Latency While Transferring IoT Data to Cloud Using Gap Optimization Algorithm

**Siddhant Roy, Eesha Saxena, and Abdul Quadir Md**

## 1 Introduction

In this fast-paced world, many organizations are shifting to the concept of cloud environment, while embracing the concept of virtualization. This underlying growth and demand of Cloud systems have developed a competitive edge among other companies, and different cloud service providers. Companies majorly use these services to store and manipulate data that they gathered over time using IoT devices that are mostly set up and stagnant in terms of geographical movement. These data sets are then utilized to predict patterns, solve corporal problems, analyze a situation, or derive a better tangible solution. Data manipulation using cloud and IoT devices gives an upper hand to the advancement of technology as there is no manual involvement applied and it is cost-effective. With all these advantages come certain setbacks such as latency delays. Latency delay is understood as the time elapsed between sending and receiving the data, in this case, between IoT devices as source and cloud servers as destination receivers. In other words, latency can be defined as the measure of the responsiveness of an application. This can cost multinational companies a great loss and delay data manipulation at the zeroth hour. High latency can constrict the network, and thus reduce its performance. Although using the concept of cloud as a means of data storage and processing platform to decrease data detainment has been proven quite efficient in keeping the obstruction under control.

---

S. Roy · E. Saxena (✉) · A. Quadir Md
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
e-mail: eesha.saxena2019@vitstudent.ac.in

S. Roy
e-mail: siddhant.roy2019@vitstudent.ac.in

A. Quadir Md
e-mail: abdulquadir.md@vit.ac.in

Various factors still hinder the use of services to a great extent. Multiple attributes are influencing the degree of latency including the bandwidth of the internet connection majorly operating from the data source's side, geographical distance, contention ratio as well as traffic on the server. The rest of the paper is structured as follows: Sect. 2 presents the related work in the area of Latency and its reduction with problem statements. In Sect. 3, the most important factors of latency are discussed. Section 4 describes a few methods of reducing latency. Section 5 discusses the proposed work and Sect. 6 describes the data analysis and Sect. 7 concludes the paper with future directions.

## 2   Literature Review

There have been multiple attempts to reduce latency. From [1], we get to learn two things. First, finding the paradigmatic batch of servers that curtail latency with the aid of characterizing can be achieved through the means of intelligent servers. Second, by using redundancy, performance can be achieved with straightforward policies. The authors of [2] propose a distributed service broker policy-based algorithm for logistics information systems. In comparison to the previously specified service broker policies of proximity-based and virtual machine-based, the suggested policy will provide an improvement. A stable matching-based algorithm has been implemented in [3]. It reduces latency in polynomial time and then utilizes local search to improve its efficiency further. On evaluating further, this algorithm offers a better time complexity. For [4], the major focus of the paper is on how to reduce the recovery time after a disaster strikes on the cloud platforms, which causes costly disturbances in their services, thereby a need for Disaster Recovery (DR) [4]. The paper talks about the advantages of CloudSim and an escape from the impacts of long latencies on cloud platforms through the use of 'Bulk data transfer'. Malvinder Singh Bali and Shivani Khurana [5] proposed a practical method that can be used to calculate the network latency which hampers the performances of the cloud services. Their work also revealed the service disruption effect due to DDoS attacks on cloud networks. In [6], the authors propose a joint computation-plus-communication model for VM allotment that cuts down energy expenditure in a Cloud data center. First, they captured the heterogeneous connections between the VMs, stated within the same data center networks considering the data traffic exchange between them. The energy consumption due to VMs migrations was patterned by surveying both data transfer and computational expenses. And thirdly, weight is the aim. This is not a parameter for the proposed VMs allocation process. The paper [7] highlights an external factor- geographical location as one of the major factors in producing latency effects for cloud servers. The authors concluded that the consideration of Cloud computing should account for the entire system, from Cloud to the end-users. Redundant cloud architecture, redundant data center infrastructure.

Content caching to the end-users can be tremendously improved by using technologies such as TCP acceleration and CDNs. From [8] we can understand the

different issues that are faced due to latency, and possible solutions to these problems. This paper provides a universal outlook of the categories of latency, techniques, findings, and fixes. In present times, there is a disclosure of latency-sensitive and mass data in the case of IoT applications. The broad summary of the paper [9] is that it deals with the estimation of latency while a transfer of data from IoT sources to the cloud and back to the sources takes place. The paper focuses on the concept of IIoT, that is, the Industrial Internet of Things. IIoT being referred to as worldwide applications, the paper assesses some cases where interconnected nodes are deployed in separate continents. The experiments conducted by the authors concluded that simple lightweight protocols such as MQTT and inexpensive industrial-grade IoT devices are feasible. The paper revolves around the idea of an 'Industry 4.0' scenario where the need for short-latency would flood the market. To alleviate high latency when processing industrial data, the Industrial Internet of Things cloud–fog hybrid network (ITCFN) structure is presented in [10]. But, in [11], the authors have proposed an algorithm for data packet allocation and selection. It employs a fuzzy inference method in combination with reinforcement learning and neural network evolution methodologies. To optimize the performance of Cloud Computing, the authors in [12] have discussed different kinds of approaches for lessening latency. The paper has discussed various aspects through which other external factors affect the performance of cloud services and how latency, one of the major factors, can be reduced in different heterogeneous domains. Concepts like Intelligent virtual placements, Binary ATM Placement, and Euclidean Steiner Tree-based approach have been approached to resolve the negative impact of latency on cloud services/platforms.

## 2.1  Problem Statement

The highly in-demand domain of computer science, i.e., Cloud Computing, forms the base of this paper. Cloud Computing is one of the most recent technologies in which the research community has shown interest and can be perceived as an on-demand, and sometimes a pay-as-you-go service, which allows access to a reserve of numerous resources, without direct active management from the user. Most vendors of cloud computing provide applications and enable technology, infrastructure, hardware, software, and much more for their clients. In simple real-life terms, it is used by companies and/or users to rent data servers rather than investing in their servers which can be costly as well as difficult to manage. By using the Cloud, users and companies can access applications and data from anywhere in the world. There are various factors involved in making the process of cloud computing services a successful one. The factor that we will be focusing on is latency in cloud computing services.

In the language of networking, latency is intended to express the time difference between the dispatch of data/signal from the respective source and arrival at the destined service. The internet may seem like a place where services are delivered instantaneously but that is not the case and in reality, there is a delay of a few

milliseconds. This might be frustrating for users and financially burdening for the service providers and the users who have invested in multiple cloud services. With the exponential increase in cloud service demands and higher expectations for speed and connectivity, latency is among the most influential facets in this compass.

Services that are highly affected by latency require high amounts of CPU computation, huge amounts of storage, or numerous server requests. High latency is a huge problem for companies, and reducing latency may reduce costs for all services, and improve the user experience and performance of the services. High latency causes the wastage of resources since a higher latency means a longer usage of resources for a task. And also, a lower latency can sometimes mean a lower requirement of resources, which can save costs for the client. We aim to write a comprehensive paper that will describe the effects of latency on cloud computing services and therefore its impact on economics and performance.

## 3   Factors Contributing to Data Delay

High latency can adversely affect the performance of the network and greatly reduce how quickly an application communicates with users. This may affect the user experience adversely, thus affecting the productivity and usability of the application. Some factors which contribute to this are

### 3.1   Geographical Distance

Defined as the distance between the data trading platforms. Simply, the distance between consumer devices making requests, and servers responding to those requests. Data is limited by means of the laws of physics, and it cannot travel quicker than the speed of light.

### 3.2   Contention Ratio

Stated as the ratio of actual demand to the available bandwidth. The lower the contention ratio, the lower the quantity of users that may be seeking to use the bandwidth at any time frame, and, consequently, the higher the effective bandwidth offered. In simple words, the lower the contention ratio, the higher the QoS of the network. This means that if you have a 20 Mbps service with a 10:1 contention ratio, and all other 9 subscribers you are sharing with are heavily utilizing their services, each party's connection speed could drop to as low as 2 Mbps

### 3.3   Bandwidth of Internet

Bandwidth is a dimension of the quantity of data that can be exceeded through a connection, in a given time. It can be thought of as a measurement of either speed or usage. Bandwidth is like a pipe, which can be both wide and narrow. A wider pipe will allow more data to be passed through it.

### 3.4   Server Load

The number of processes that are waiting in the queue to access the processor, in a certain period of time. The higher the load, the greater amount of CPU and disk space is required by the server.

### 3.5   Quality of Routing Devices

Latency is impacted by the router's efficiency in processing incoming data. The main job of routers is to forward data packets over various networks of computers, and are in charge of traffic direction caused by these data packets. Routers spend some time examining a packet's address and other data information related to its destination and, in some instances, insert specific data. The lag of transmission rises with each jump from one router to another.

### 3.6   Transmission Medium

Defined as the palpable path between the starting point and the ending point. The nature of the medium can heavily influence latency. For instance, fiber optic cables offer significantly much more bandwidth than traditional copper cables. Since fiber optic cables are made out of glass strands instead of metal, they are unaffected by interference from other wires and radio signals. They also have the capacity to carry more data. Nowadays, upheavals in fiber optic technology allow data to travel at about two-thirds of the speed of light.

## 4   Methods to Reduce Latency

Latency has been increasingly costing network society and companies a great deal of pain, but historically networks have been designed to maximize throughput and utilization. Shaky video or audio, slower loading times, or timed-out requests may

be the result of excessive latency. To reduce latency, we can use various methods such as Traffic Shaping, Edge Computing and Fog Computing. We have explained these in detail:

## 4.1  Traffic Shaping

On computer networks, traffic shaping is a bandwidth management approach that involves delaying some or all packets of data in aims to introduce them into line with a specified traffic profile. By delaying some packets, traffic shaping can enhance or ensure execution, reduce latency, or enhance bandwidth utilization for those packets. By adjusting the burst size and employing a leaky bucket method to balance the production capacity, traffic shaping works to minimize delay, distortion, and interruption. Only packets departing a connection can be shaped, not data approaching the network.

## 4.2  Edge Computing

Edge computing focuses on relocating the data processing servers closer to the source so that the data, which can be processed locally, does not have to travel long distances to its native cloud servers. This gives an undue advantage of reducing the distance and proportionally its latency. The edge or local servers are temporary storage for newly published data and compute fewer processes, the rest being sent to the cloud for further evaluations. Edge computing has been recognized as a centralized system for managing and handling data, which is also closer to the device root since it is located at the edge of the network system. The analysis of data figures in an edge computing model is done in either of the ways; in the IoT device itself(on the location where the data is collected) or on one of the edge network servers, both of which effectively reduces latency and bandwidth of the whole system significantly diminishing operating costs for companies and organizations. With the growth of the IoT industry, the demand for edge architecture has also rhythmically escalated and so needs to incorporate artificial intelligence in the architecture namely *Edge AI*. Justifying the above statement, since it is well known that artificial intelligence heavily depends on data processing and transmission of the same, latency in the domain is a big backdrop. Thus, a new architecture which is born through the amalgamation of both of these powerful technologies, *Edge AI* has taken the industry by a snap, including the high degree processing powers of powerful technologies incorporating artificial intelligence with a well handled and efficient latency reduction technology of edge computing. The importance of this accommodation will directly affect how data is processed at a surface level, dividing the data sets into categories and channelizing the intelligence into prioritizing data actions and proceedings. This in turn will assure substantial Edge AI facilities and life-cycle governance of end devices.

### *4.3   Fog Computing*

Fog computing, very much like edge computing, is a branched version of cloud computing that has been routed all over the internet. It is a distributed network environment. Fog computing is based on the concept of fog nodes. Fog nodes are typically found near the data source. This offers an advantage of more processing and storage capacity. It creates low-latency network connections between various devices. Fog computing provides the choice of the location where the computation will happen. Edge computing is a subset of Fog computing. Fog computing is edge processing and the network connections that bring the data from the edge to its endpoint. Fog nodes can quickly process the data. This architecture significantly reduces the amount of bandwidth needed, compared to if that data had been transported to the Cloud for processing. It can be used in scenarios where there is a very low bandwidth connection to send data. Fog computing can be used when there is an enormous number of devices so that the Cloud doesn't get cluttered. Fog computing facilitates the control of privacy.

## 5   Proposed Architecture Using Gap Optimization Algorithm

IoT devices generate millions of data per second, data flow from IoT devices is almost as quick as the speed of light, it suffers from latency delays which hurdles the processing of the data on time. Of all the factors impacting latency, the distance between the cloud servers and IoT devices seems to be the biggest factor. Local source internet bandwidth might be up for blame as well but cannot always be controlled. The most efficient way to minimize latency, looking at our available resources, is to minimize the distance as well. We propose an algorithm that is based on geographical conditions ergo, the less the distance between the cloud server and data source, the lesser will be the chances for low latency (Fig. 1).

There is one source of data and multiple cloud servers available, provided the distance (in km) of each from the IoT data source is stated. Wielding this data, the distances of the respected servers are stored in an array which will be used in the gap optimization algorithm. The algorithm is divided into two sections; the first one will be focused on choosing a viable cloud data server to curtail the surface latency. The second section focuses on reducing latency through the process of fog computing and edge computing taking into account the architectures are well versed in artificial intelligence and data prioritizing. We propose an algorithm, where we focus on reducing the distance between the source and the designated server. The distance between the servers and the source is taken into an array (dis[]) and hence are sorted in increasing order To reduce the computational time, we have used a sorting algorithm whose worst time complexity will be O(log(n)). This algorithm will have to be applied on the Cloud provider's side. After the sorting, the user can select and
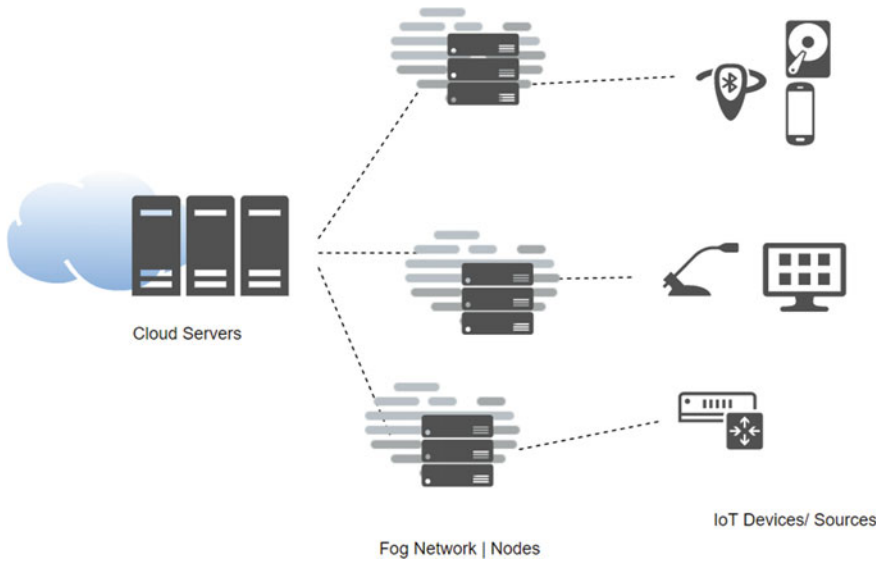
**Fig. 1** IoT data transfer to cloud servers using fog networks

prioritize the initial elements (x) in the sorted array and create a request-response for the same. If the external factors like bandwidth are justifiable the user will be able to select the server (M) whose data delay will be least compared to the other servers (Figs. 2, 3 and Table 1).

After an appropriate cloud server is chosen for the IoT device, the device needs a fog and edge network to not let distance affect data transitions while transferring data to the designated server. The IoT application is written or ported according to the network requirements. The IoT data is then to be sent to the nearest edge network where the architecture will decide to do one of the following

1. Process the data—happens almost instantly (milliseconds) and sends the results back. Usually, M2M mechanisms are adopted.
2. Store the data—could take minutes to process it locally on the node/network.
3. Proceed the data over to the fog nodes registered in LAN systems and finally over to the cloud servers where big data is processed and stored globally.
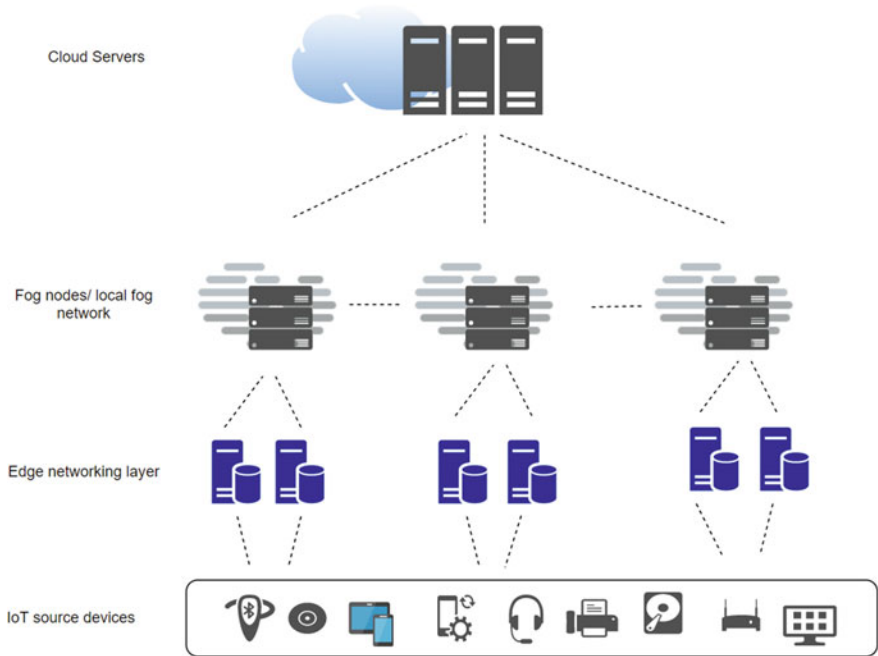
**Fig. 2** Hierarchy of architecture of network models

---

**Gap Optimization Algorithm**

---

*Algorithm Begin:*

    Process **start**

    a: **if** ( size of dis[] >1) **then**

    divide dis[] into two halves by the midpoint **m=n//2**

        **store** -> ($i_0$, $i_m$) in left[]

        c: **store** -> ($i_{m+1}$, $i_n$) in right[]

    **Repea**t steps [a,c]  for left[] and right[]

        sort and merge left[] and right[]

    Choose  and  prioritize  the  top  x(depends  on  the  user) comparable(in terms of distance) servers - [$i_0$ , $i_x$]

    Send a response request to the selected data centers

        **if** ( $t_{Mi+1}$ < $t_{Mi}$), **then**

        **replace** $M_i$ with $M_{i+1}$ in dis[]

        Choose $M_0$ as the cloud server for the device

    **End**

---

**Fig. 3** Proposes gap
optimization algorithm
working
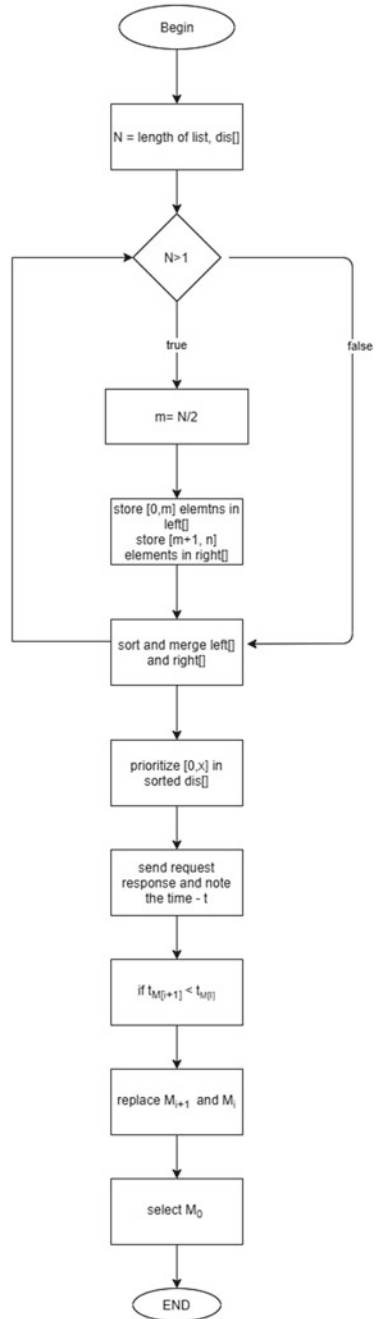
**Table 1** Parameter of gap optimization algorithm

| Notations | Description |
|---|---|
| n | Total number of available servers |
| M | Data server center |
| dis[] | Array storing the distances |
| i | i ∈ [0,n]; signifies the ith element of the array |
| m | The middle point of the list/array: dis[] |
| left[] | Empty list to store the leftmost values of the list dis[] |
| right[] | Empty list to store the rightmost values of the list dis[] |
| t | Response time of M |

## 6 Data Analysis

Latency majorly depends on distance. Latency is measured in milliseconds. During testing speed, it can be referred to as a ping rate. It is preferable to have as low latency as possible. Low latency is an extremely important part of IoT. It can improve both the user experience and also unlock the potential of an IoT device.

The above data in Table 2 and Fig. 4 is a projection of ping tests taken for AWS cloud service provider and Table 3 and Fig. 5 is for Azure cloud service provider [13]. A ping test determines the time in milliseconds in which a request-response reaches and is reverted. We tested out four major areas of the world and as it can be very clearly seen that latency is directly proportional to the distance between the client and the server. From the graph, we can assume that major outliers are present in the higher end of distribution since in most cases the median is lower than the mean. Both the above tables are sorted in ascending order of their mean latencies (top to bottom).

## 7 Conclusion and Future Work

In this paper, we have mapped a solution for the reduction of latency and its effects in the domain of transferring IoT data to cloud servers. The proposed algorithm considers the various factors of latency and uses these dependencies to filter out and choose a cloud server that will have minimal impact on the speed of the data transfer involved through the transfer process. The paper takes into account the already existing technologies and architecture; fog and edge computing, for increasing the output of the algorithm drafted. In the future, we aim to implement and analyze this algorithm and generate experimental results using cloud simulation tools. We also aim to increase the dynamic nature of the existing algorithm with constantly changing values. We wish to further branch our knowledge over other aspects and technological advancements on cloud-based subjects and related matters.

**Fig. 4** Data projection of AWS mean and median
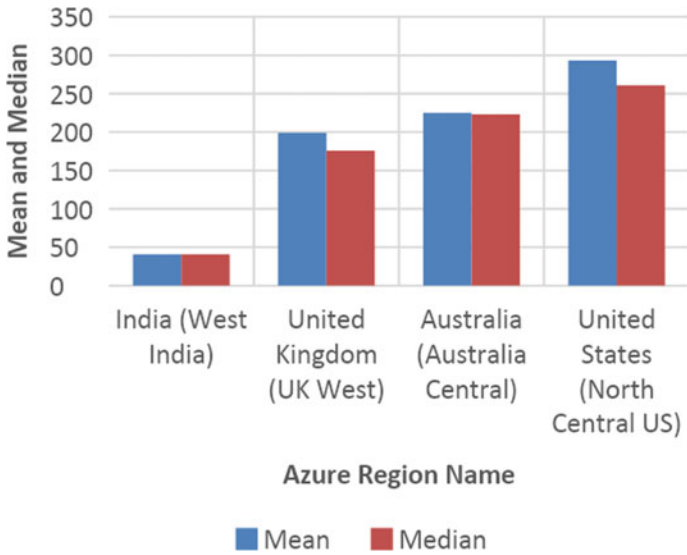


**Fig. 5** Data projection of AWS mean and median

**Table 2** Data projection of AWS cloud server

| AWS region name | Region code | Mean | Median |
|---|---|---|---|
| Asia Pacific (Mumbai) | ap-south-1 | 117 | 87 |
| Europe (London) | eu-west-2 | 585 | 577 |
| Asia Pacific (Sydney) | ap-southeast-2 | 720 | 469 |
| US East (N. Virginia) | us-east-1 | 725 | 902 |

**Table 3** Data projection of azure cloud server

| Azure region name | Region code | Mean | Median |
|---|---|---|---|
| India (West India) | West India | 41 | 41 |
| United Kingdom (UK West) | UKwest | 199 | 176 |
| Australia (Australia Central) | Australia central | 225 | 223 |
| United States (North Central US) | Northcentralus | 293 | 261 |

# References

1. Kwon M, Dou Z, Heinzelman W, Soyata T, Ba H, Shi J (2014 Use of network latency profiling and redundancy for cloud server selection. 2014 IEEE 7th international conference on cloud computing, Anchorage, AK, USA, pp 826–832. https://doi.org/10.1109/CLOUD.2014.114
2. Dubey S, Jain S (2015) A new algorithm for improving latency in distributed data center for logistics information system over cloud. Int J Comput Appl 130:16–20. https://doi.org/10.5120/ijca2015906844
3. Ghai KS, Choudhury S, Yassine A (2020) Efficient algorithms to minimize the end-to-end latency of edge network function virtualization. J Ambient Intell Human Comput 11:3963–3974. https://doi.org/10.1007/s12652-019-01630-6
4. Sharma S, Sharma P (2016) Impact of latency on the economics of cloud computing. (IJCSIT) Int J Comput Scie Inf Technol 7(3):1414–1416
5. Bali M, Khurana S (2013) Effect of latency on network and end-user domains in cloud computing, 777–782. https://doi.org/10.1109/ICGCE.2013.6823539
6. Alicherry M, Lakshman TV (2013) Optimizing data access latencies in cloud systems by intelligent virtual machine placement. 2013 Proceedings IEEE INFOCOM. https://doi.org/10.1109/infcom.2013.6566850
7. Singh A, Hemalatha (2012) Comparative analysis of Low latency on different bandwidth and geographical locations while using cloud-based applications. Head department of Software systems, Kalpagam university Coimbatore: IJAET ISSN: 2231-1963
8. Tamizan M (2020) Latency issues in internet of things: a review of literature and solution. Int J Adv Trends Comput Sci Eng 9:83–91. https://doi.org/10.30534/ijatcse/2020/1291.32020
9. Ferrari P, Sisinni E, Brandao D, Rocha M (2017) Evaluation of communication latency in industrial IoT applications. 2017 IEEE international workshop on measurement and networking (M&N). https://doi.org/10.1109/iwmn.2017.8078359
10. Liu W, Huang G, Zheng A, Liu J (2020) Research on the optimization of IIoT data processing latency. Comput Commun 151:290–298, ISSN 0140-3664. https://doi.org/10.1016/j.comcom.2020.01.007
11. Shukla S, Hassan MF, Khan MK, Jung LT, Awang A (2019) An analytical model to minimize the latency in healthcare internet-of-things in fog computing environment. PLoS ONE 14(11):e0224934. https://doi.org/10.1371/journal.pone.0224934

12. Srivastava S, Singh SP (2016) A survey on latency reduction approaches for performance optimization in cloud computing. 2016 second international conference on computational intelligence & communication technology (CICT). https://doi.org/10.1109/cict.2016.30
13. https://cloudpingtest.com

# Man-in-the-Middle Attack Mitigation in IoT Sensors with Hash Based Multidimensional Lamport Digital Signature

**T Bebin Josey and D. S. Misbha**

## 1 Introduction

The largest network design interconnected by small sensors is the wireless sensor The Internet of Things (IoT) has had a direct impact on a variety of domains recently, including industries, smart cities, health care systems, and smart homes, by enabling devices to connect and exchange data easily [1, 2]. Personal data is securely stored in memory on laptops and cell phones, whereas communication is wireless, making data security a challenging task. The Internet of Things has a tiered architecture with four layers: an application layer, an infrastructure layer, a network layer, and a device layer. Each of these levels makes use of a variety of software, devices, and Internet of Things applications, posing a variety of security concerns and risks [3]. Hence, IoT devices must follow strong security protocols in order to protect data from malicious network attacks and to ensure security features such as confidentiality, integrity, and availability. Numerous industry protocols and guidelines for device manufacturing, communication protocols, security services, and solutions have been developed for safely integrating the IoT into industrial processes. Digital Signature is considered to be the most popular security protocol of these [4, 5]. The general functionality of a digital signature is depicted in Fig. 1.

As illustrated in Fig. 1. A digital signature authenticates and validates a signature using an asymmetric cryptographic technique. The fundamental concepts behind the operation of a typical digital signature security mechanism [6, 7].

T. B. Josey (✉)
Research Scholar, Department of Computer Science, NMCC Marthandam, MS University, Tirunelveli, India
e-mail: bebin_josey_csa@nmcc.ac.in

D. S. Misbha
Research Guide, Department of Computer Applications, Nesamony Memorial Christian College, Marthandam, MS University, Tirunelveli, India
e-mail: misbhasatheesh4@gmail.com
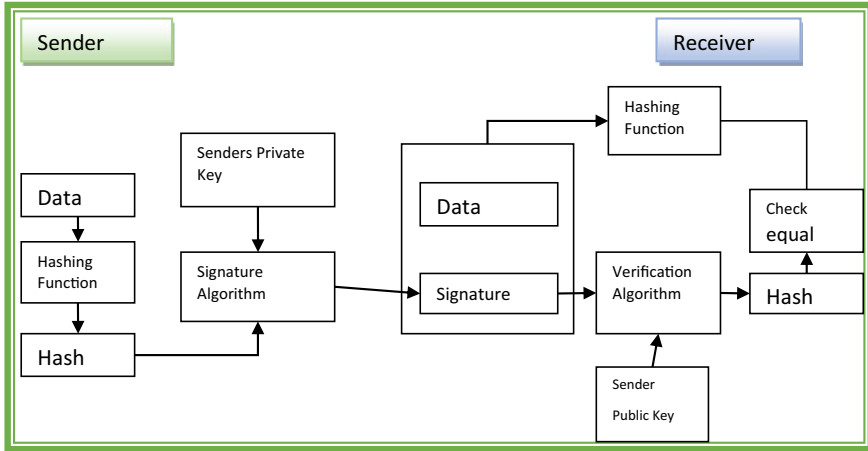
**Fig. 1** Working procedure of digital signature

- Public Key: A user can share his or her public key with anyone who requests to authenticate his or her signature, whether or not they wish to share it with them.
- Private Key: This is the cryptographic key that is used to generate signatures. The private key is safeguarded. The private key of a user should be kept secure.
- Signing Process: To begin, we will determine the message digest of a simple message using hashing algorithms. The digest is encrypted with the signer's private key, referred to as the digital signature. For encryption, we used a variety of cryptographic encrypting approaches.
- Authentication process: When two hash values match after receiving the message, it indicates that the digital signature is valid [7].

Numerous digital signature algorithms have been developed for IoT device authentication, the most widely used of which is the Lamport digital signature scheme. The primary disadvantage of this method is, its private keys may be used to securely sign only one message. In addition, it requires high computation costs and complicated residual operators. The goal of this research is to develop a high-performance Lamport digital signature capable of analyzing and resolving man-in-the-middle attacks that occur when resource-constrained sensor nodes are employed in IoT networks. We propose the modified Lamport signature system in this article, which is a variant of the original Lamport signature scheme. This suggested digital signature enables the use of a private key to securely encrypt an unlimited amount of messages while maintaining a shorter signature with the hash function but at the expense of increasing the length of the public and private keys. This proposed protocol is mainly developed to address the threat of man-in-the-middle attacks on IoT networks.

The following is a synopsis of this article. The second section addresses the advantages and disadvantages of newly established authentication solutions against man-in-the-middle attacks. Section 3 describes the key modules of the suggested security

protocols. Section 4 compares the existing security protocols with the proposed protocol using experimental analysis. Finally, the proposed strategy is summarized.

## 2 Literature Review

This section examines the newly designed Middle Attack Mitigation techniques in depth.

FarouqAliyu et al. [8]. investigated the many possibilities for deploying an intrusion detection and prevention system in the event of a Man-in-the-Middle (MitM) attack utilizing IDS nodes. The IDS has IDS nodes that interrogate nodes one hop away at each interval and examine their response in terms of arrival time, context, and content. And by utilizing lightweight encryption, IPS avoids the problem of Man-in-the-Middle attacks and their variants. IPS encryption and decryption are performed using AES (128-bit key and block size). The system can be enhanced in the future by changing the delay from static to dynamic.

Kore et al. [9]. presented IC-MADS, a fast cross-layer trust computing technique for MIMA attack detection. IC-MADS employs a probabilistic computation and assessment approach in order to create an effective Cluster Head clustering algorithm (CH). The likelihood of a sensor node is computed using the degree parameters of the node, the leftover energy, and the distance to the base station. By allocating CH as the node with the highest probability, the energy imbalance and load balancing problems are resolved. MIMA attackers are recognized globally using a multi-layer trust evaluation technique. Trust is computed for each node based on the sensor node's parameters. Following that, the attacker node is determined by comparing the aggregate trust value of all sensor nodes to the threshold value. The approach can be enhanced by using a lightweight cryptographic-based solution for user authorization and authentication.

Zhang et al. [10]. described a method based on tag verification and embedding for authenticating a lightweight PHY-layer for IoT devices in smart cities. The tag signal is encrypted and incorporated in the device's signal in such a way that it is independent of the message in the transmitter. The tag signal is collected and validated at the receiver end using signal detection techniques. The analytical model, matrix analysis, and composite hypothesis testing theories are used to illustrate the framework's authentication performance. The system's disadvantages include the fact that selecting a tag signal that is dependent on both the transmitter and the receiver introduces the issue of standard impersonation attacks. Additionally, using a shared secret key for the encryption process reduces the likelihood of unwanted manipulation and detection of the man-in-the-middle attack.

Kim et al. [11]. proposed a method for detecting attacks in the Internet Of Things through the use of Machine Learning techniques. The strategy focuses solely on assaults against IoT devices and creates models based on machine learning in response. The approach makes use of the N-BaIoT dataset. The N-BaIoT dataset is generated as a result of botnet attacks being injected into many sorts of IoT devices,

including security cameras, doorbells, webcams, and baby monitors. Then, for each device, a botnet detection model is constructed utilizing a variety of machine learning (ML) and deep learning (DL) models. The model with the highest F1 score is then assessed to determine if it is the most effective model by doing multiclass and binary classification on each model. The method's drawback is that it involves consideration of many protocols utilizing TCP and UDP.

Rehman et al. [12]. proposed the secure health fog system (SHFS), a prototype that aggregates data from many Internet of Things-enabled sensors to retain a customized database for adaptive model tuning (IoT). This adaptive model is then modified on a regular basis via the process of generating personalization. Moreover, a Zigbee Secure Health Fog (ZigbeeSHF) method is suggested here that mitigates cyber-attacks through the use of symmetric and public key encryption. Whenever data is transferred, it is encrypted and combined with an encrypted digital signature to ensure its integrity, secrecy, and authentication. Additionally, computerized reliability assessment systems such as Scyther and AVISPA are used to evaluate protocols in terms of the adversary model and the adversary environment's numerous possibilities. The model's reliability and effectiveness are evaluated using the smart studio apartment.

Although various security schemes exist, the drawbacks of these cryptographic signatures, whose trustworthiness is contingent on their computational difficulty, are that they are particularly vulnerable to being broken by a supercomputer.

## 3   Proposed Methodology

Because of the rapid advancement in the field of practical supercomputing, such signature systems are compromising their effectiveness. Lamport Signature Scheme is a highly safe and efficient method of authenticating digital signatures. However, due to the large number of hash functions used in its implementation, it is considered to be quantum resistant. The sender generates 256 pairs of 256-bit values at random in a Lamport Signature Scheme implementation. These numbers are generated securely with the use of a random number generator. These pairs total 16 KB in size. This is the private key of the signer. The sender then encrypts each of the previously created integers, yielding 512 hashes, each 256 bits long and covering a total of 16 KB. The sender selects the hash sum's n-th bit. When the n-th bit is set to 0, the sender selects the n-th number from the very first list; when the $n + 1$th bit is set to 1, the sender selects the $n + 1$th number from the second list, and so on. This procedure generates $256 \times 256$-bit integers, each of which covers an 8-kilobit space. The sender's signature is composed of these 256 numbers. The recipient generates a 256-bit hash sum when it receives the message. The receiver may select a number from one of the recipient's two public key lists for each n-th bit in the message, just as the sender did for the private keys that matched the hash sum bits. Following that, the recipient hashes each integer in the signature. If these two lists of numbers match, you have received a real message. It is possible that the content of the message or the
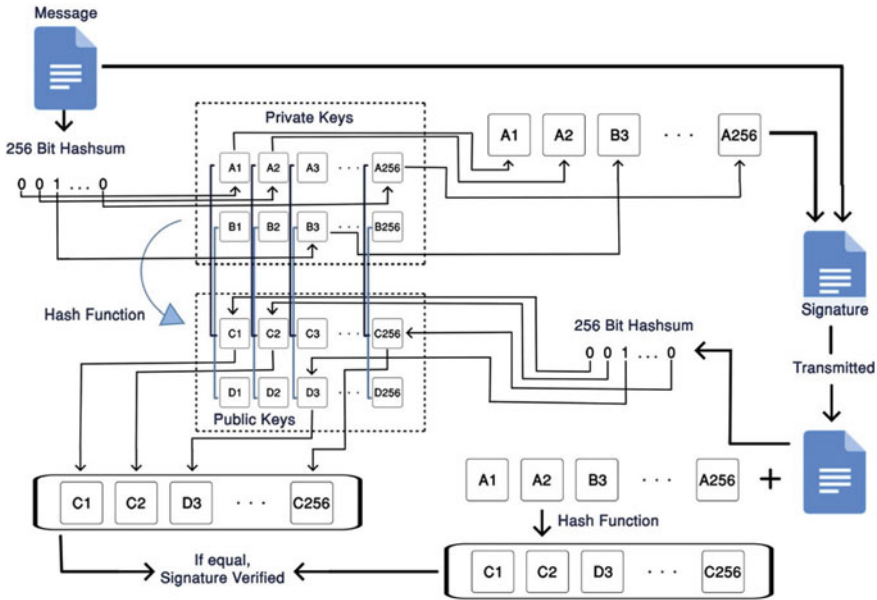
**Fig. 2** İllustrates the procedure for implementing the lamport signature scheme

signature was altered during transmission. Figure 2 illustrates an implementation of the classic Lamport digital signature mechanism [13, 14].

The fundamental shortcoming of existing Lamport techniques (one-time signature schemes) is that they can only be used to securely sign a single message.

## 3.1 Proposed Lamport Signature Scheme for IoT Sensors Security

The traditional Lamport approach uses the private key to sign a single message, which is a significant disadvantage [14, 15]. This research addressed this issue by utilizing n-dimensional messages sign with a shorter signature. This proposed digital signature permits the secure encryption of an endless number of messages using a private key while preserving a shorter signature. In this upgraded Lamport approach, the hash function is applied to the message that will be signed. Second, the signed message is divided into a predetermined number of blocks (n number of hashed blocks with fixed size). The signature is constructed using length numbers extracted from the private key matrix in the following manner: the line indicates the block location, while the column indicates the numerical value of the bit block. Currently used Lamport one-time signature schemes produce the private key using randomly generated numbers and the public key based on the private key and the number of hashes. The signed

value is first hashed, and each bit of the hash is signed in the following manner. To protect a 128-bit long hash value, for example, $2 \times 128 = 256$ numbers are necessary for the private and public keys, as each bit has two potential integrals (1 and 0). Each bit in the existing technique required key values, which significantly increased the computing complexity and cost. The group of bits is enclosed in hashed blocks and a key pair is assigned in this research. This significantly reduces the number of key pairs and the computational cost. The message is signed using the algorithm 1 in the proposed hashed multi-dimensional Lamport signature scheme.

---

***Pseudo code 1 : Signing usinghash based multi-dimensional Lamport signature scheme***

| | |
|---|---|
| ***Input*** | $Message, Private_{Key}, Block_{Size}$ |
| ***Step 1*** | $Sign \leftarrow [\ ]$ |
| ***Step 2*** | $Hash_{Message} \leftarrow Hash(Message)$ |
| ***Step 3*** | $Blocks \leftarrow Split(Hash_{Message}, Block_{Size})$ |
| ***Step 4*** | ***for*** $(i = 0\ to\ n\ blocks)$ |
| ***Step 5*** | $Sign \leftarrow Private_{Key}$ |
| ***Step 6*** | ***End for*** |
| ***Step 7*** | Return Sign |

---

Prior to verifying the signature, the hash function is applied. Second, the total number of blocks is determined. Finally, each block is subjected to key verification. If a key verification error occurs, the signature is denied. Algorithm 2 details verifying operation.

---

***Pseudo code 1 : Verifying usinghash based multi-dimensional Lamport signature scheme***

| | |
|---|---|
| ***Input*** | $Message, Public_{Key}, Block_{Size}$ |
| ***Step 1*** | $Hash_{Message} \leftarrow Hash(Message)$ |
| ***Step 2*** | $Block_{count} \leftarrow Count(Hash(Message))$ |
| ***Step 4*** | ***for*** $(i = 0\ to\ Block_{count}\ do)$ |
| ***Step 5*** | ***if*** $\left( verify \left( Public_{Key}(Block) \right) \right)\ ! = true$ |
| ***Step 6*** | *return false* |
| ***Step 7*** | ***End if*** |
| ***Step 8*** | ***End for*** |
| ***Step 9*** | *Return true* |

# 4 Experimental Analysis

## 4.1 System Configuration and Software Tools

The machine is equipped with an Intel Core i5-10200H processor, 16 GB DDR4 RAM, and an NVIDIA GeForce RTX 3050 4 GB GDDR6 graphics card to prove the proposed multi-dimensional Lamport signature scheme's efficacy. Further, the multi-dimensional Lamport signature scheme's was implemented using JDK 1.8, the Eclipse integrated development environment, and Windows 10.

## 4.2 Efficiency Analysis

In the Existing Lamport one-time signature scheme, every bit is verified by a key pair, which enhances security but also dominates computational complexity and computational cost. In this research, the bit values are converted into blocks of a certain size and all the blocks are encoded by a hash function to reduce the computational cost without compromising security. This significantly reduces the IoT sensors' computational complexity. The following table compares the computational complexity of the traditional and proposed Lamport Digital Signatures. Computational time is specified in milliseconds in Table 1.

**Table 1** Computational complexity comparison between traditional and proposed Lamport Digital Signature

| Lamport digital signature | | | | Hash-based multidimensional lamport digital signature | | | |
|---|---|---|---|---|---|---|---|
| Data size | Key size | Total key size | Computational time (MS) | Block size | Key size | Total key size | Computational time |
| İ bit | 128 | $2 \times 128 = 256$ | 1181 | 4 bit | 128 | $2 \times 128 = 256$ | 1108 |
| 2 bit | 256 | $4 \times 256 = 1024$ | 2600 | 8 bit | 256 | $2 \times 256 = 512$ | 2690 |
| 4 bit | 256 | $8 \times 256$ | 4567 | 12 bit | 256 | $2 \times 256 = 512$ | 3100 |
| 6 bit | 256 | $12 \times 256$ | 6900 | 18 bit | 256 | $2 \times 256 = 512$ | 3412 |
| 8 bit | 256 | $16 \times 256$ | 8500 | 24 bit | 256 | $2 \times 256 = 512$ | 3516 |
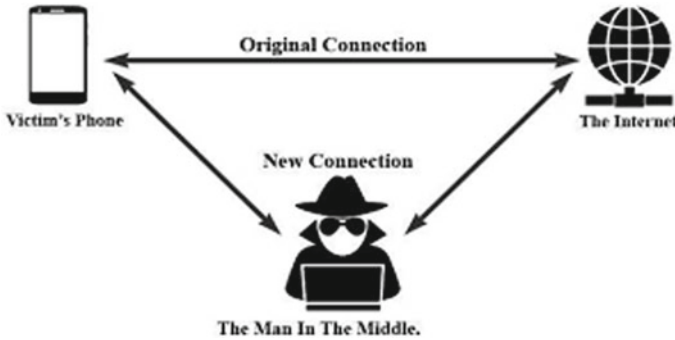
**Fig. 3** Man-in-the-middle attack model

## 4.3 Analyses of Security Against Man-In-The-Middle (MITM) Attacks

Consider the following scenario, in which A is the recipient. A third party, let's call him C, can act as a go-between for the two parties who rely on signatures for verification. After A has added the digital signature to the message, B decrypts it using the public key. If a message is signed using C's private key, it must be decoded using C's public key, which he claims to possess. In this case, there is no guarantee that the public key genuinely belongs to the A; it simply ensures that the public key belongs to the individual. There is no direct relationship between A and B in Fig. 3. Rather than that, A sends a transmission to B, which C intercepts while impersonating as A. This occurs because there is no mechanism to determine if the sender claiming possession of his public key in fact owns it [9]. This is why a new solution is necessary to validate the sender's public Key appropriately.

The owner must obtain the digital signature certificate (DSC) from a third party trusted entity, the certificate authority (CA). When the recipient receives the communication, they can trust the owner's authenticity because the owner used this certificate to produce a digital signature. This certificate contains the sender's public key in addition to other information that has been verified by a trusted third party. The sender-receiver communication can be made secure against a man-in-the-middle attack by utilizing the digital certificate.

As illustrated in Fig. 4, digital certificates serve the same purpose as a passport or other form of identification in the digital world. Even if the adversary has detailed information about the security system (Key payers and signatures), he or she will be unable to conduct an MITM attack. This is because a digital signature can protect against MITM attacks through the use of a hashing algorithm and a key distribution mechanism.
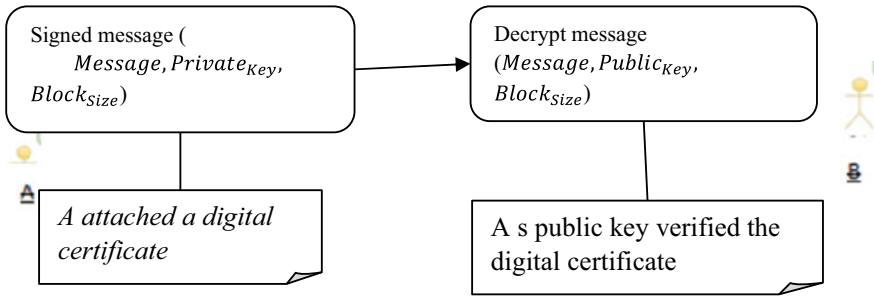
**Fig. 4** Procedure of digital certificate works

## 5 Conclusion

This study examines the usage of the Lamport digital signature technique to defend against man-in-the-middle attacks. The Existing Lamport one-time signature system verifies each bit using a key pair, which increases security but also significantly increases computing complexity and costs. The bit values are converted into blocks of a specified size in this research, and each block is encoded using a hash function to lower the computational cost without sacrificing security. This drastically reduces the computing complexity of IoT sensors. The proposed method's computing efficiency is demonstrated experimentally.

## References

1. Kumar S, Tiwari P, Zymbler M (2019) Internet of things is a revolutionary approach for future technology enhancement: a review. J Big Data 6:111. https://doi.org/10.1186/s40537-019-0268-2
2. Khanna A, Kaur S (2020) Internet of Things (IoT), applications and challenges: a comprehensive review. Wireless Pers Commun 114:1687–1762. https://doi.org/10.1007/s11277-020-07446-4
3. Zaheeruddin, Gupta H (2020) Foundation of IoT: an overview. In: Alam M, Shakil K, Khan S (eds) Internet of things (IoT). Springer, Cham. https://doi.org/10.1007/978-3-030-37468-6_1
4. Fang W, Chen W, Zhang W et al (2020) Digital signature scheme for information non-repudiation in blockchain: a state of the art review. J Wireless Com Network 2020:56. https://doi.org/10.1186/s13638-020-01665-w
5. Howgrave-Graham NA, Smart NP (2001) Lattice attacks on digital signature schemes. Des Codes Crypt 23:283–290. https://doi.org/10.1023/A:1011214926272
6. Lizama-Perez LA (2019) Digital signatures over hash-entangled chains. SN Appl Sci 1:1568. https://doi.org/10.1007/s42452-019-1618-6
7. Micali S, Reyzin L (2002) Improving the exact security of digital signature schemes. J Cryptology 15:1–18. https://doi.org/10.1007/s00145-001-0005-8
8. Aliyu F, Sheltami T, Shakshuki EM (2018) A Detection and prevention technique for man in the middle attack in fog computing. Procedia Comput Sci 141:24–31, ISSN 1877–0509, https://doi.org/10.1016/j.procs.2018.10.125

9. Kore A, Patil S (2020) IC-MADS: IoT enabled cross layer man-in-middle attack detection system for smart healthcare application. Wireless Pers Commun 113:727–746. https://doi.org/10.1007/s11277-020-07250-0

10. Zhang P, Liu J, Shen Y, Li H, Jiang X (2020) Lightweight tag-based PHY-layer authentication for IoT devices in smart cities. IEEE Internet Things J 7(5):3977–3990. https://doi.org/10.1109/JIOT.2019.2958079

11. Kim J, Shim M, Hong S, Shin Y, Choi E (2020) Intelligent detection of IoT botnets using machine learning and deep learning. Appl Sci 10:7009. https://doi.org/10.3390/app10197009

12. Rehman UU, Park S-B, Lee S (2021) Secure health fog: a novel framework for personalized recommendations based on adaptive model tuning. IEEE Access 9:108373–108391. https://doi.org/10.1109/ACCESS.2021.3101308

13. Alboaie S, Cosovan D, Chiorean L, Vaida MF (2018) Lamport n-time signature scheme. 2018 IEEE ınternational conference on automation, quality and testing, robotics (AQTR), pp 1–6. https://doi.org/10.1109/AQTR.2018.8402796

14. Abdullah GM, Mehmood Q, Khan CBA (2018) Adoption of lamport signature scheme to implement digital signatures in IoT. 2018 ınternational conference on computing, mathematics and engineering technologies (iCoMET), pp 1–4. https://doi.org/10.1109/ICOMET.2018.8346359

15. Kalach K, Safavi-Naini R (2016) An efficient post-quantum one-time signature scheme. In: Dunkelman O, Keliher L (eds) Selected areas in cryptography—SAC 2015. SAC 2015. Lecture Notes in Computer Science, vol 9566. Springer, Cham. https://doi.org/10.1007/978-3-319-31301-6_20

# Irrigation System Based on IOT and Machine Learning Approach

Ancy Stephen, A. Punitha, and A. Chandrasekar

## 1 Introduction

In agriculture, irrigation plays a vital role. The available water resources in India should be used wisely to avoid water scarcity during cultivation. The growth of the plant gets affected due to improper irrigation, which affects the yield of the forms. In India most of the formers do manual irrigation which might lead to wastage of water, i.e., the excess flow of water to the crop, overflow of water from the agriculture field. These can be avoided by using an automatic irrigation system. In automatic irrigation the water flow can be controlled based on soil moisture and temperature.

Many research works are already performed with automatic irrigation systems using various sensors. Android-based intelligent irrigation system which consists of several parameters that can be selected according to the stage of the crop, climatic condition, and the amount of moisture content. Temperature sensor to find the climatic change that takes place in the atmosphere [1]. Automated irrigation model with ANN-based controller [2]. The corn yield irrigation systems based on the GA algorithm [3]. The cost-effective smart irrigation with Rasp-berry Pi [4]. Designs an Internet of things based real-time monitoring system for smart agriculture [5].

A. Stephen (✉)
Research Scholar, Department of Computer Science and Engineering, Faculty of Engineering and Technology, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India
e-mail: ancys@stjosephs.ac.in

A. Punitha
Department of Computer Science and Engineering, Faculty of Engineering and Technology, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India

A. Chandrasekar
Department of Computer Science and Engineering, St. Joseph's College of Engineering, OMR, Chennai, Tamil Nadu, India

Intelligent irrigation monitoring by thermal imaging using the Cloud of Things [6]. Arduino-based smart irrigation system using wireless Robot [7]. Precision agriculture for an intelligent irrigation scheduling system [8]. De-signed an Arduino-based automatic irrigation system [9].

This paper focuses on providing farmers with automatic water supply based on the crops. The focus is to build a system that includes providing water according to requirements and activates the relay motor automatically. Implementing this system provides real-time sensing and self-controllable eliminates the requirements for manpower.

Finally, this paper provides an automatic irrigated system that calculates soil moisture and temperature and helps to take necessary action on turning ON/OFF the water supply automatically by saving water and environment. To achieve this system, the data can be collected from cloud-based IOT model and apply k nearest neighbor algorithm and Naive Bayes algorithm to find the dryness of soil and for user-friendly access, the android application is also accessible for automation from a remote location.

## 2    Problem Statement

This paper helps in monitoring soil moisture level and powers ON/OFF motor automatically. It will also help to control water wastage. The proposed model can be operated using Web and Android mobile application.

It is not needed for humans to visit the form directly to carry out the irrigation process. The dryness of the soil is determined by using machine learning algorithm. Build a system that performs an automated irrigation model which performs automatic ON and OFF the pumping motor based on the prediction of water requirements or predicted for the next water cycle.

## 3    Proposed System

For an Automatic irrigation system, the IOT model is constructed with Arduino board and sensors like soil moisture, dht11, Node MCU as shown in Figs. 1 and 2.

The moisture of soil can be measured with the help of a moisture sensor. The DHT11 sensor helps to monitor the temperature and humidity values. The water vapor is detected by measuring the electrical resistance between two electrodes.

NodeMCU which is available at free of cost runs on the ESP8266MOD. NodeMCU is like other microcontrollers such as Arduino UNO and Nano. It has built-in Wi-Fi and Bluetooth with better speed and flash space. The data are collected real time using these IOT devices.

Machine learning algorithms can be applied to find how much soil is dry and check or compare two ML algorithms KNN and Naive Bayes.
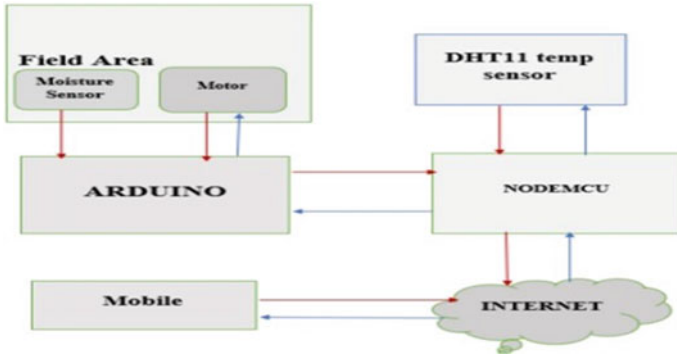
**Fig. 1** Proposed architecture for smart irrigation
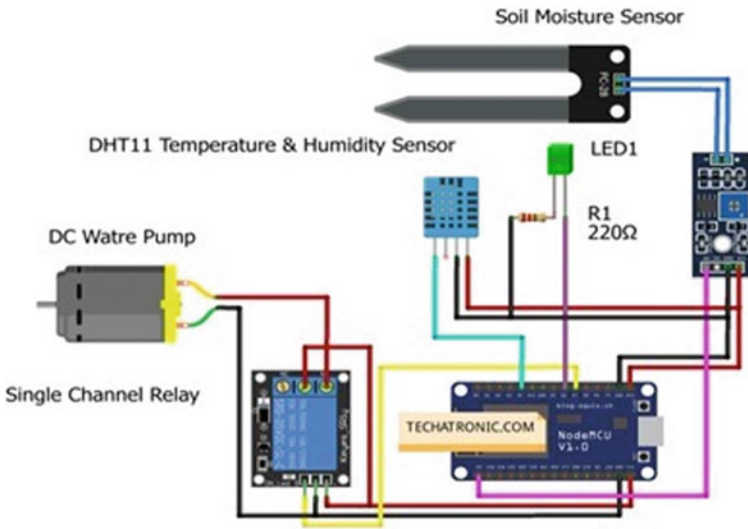


**Fig. 2** Circuit diagram

## 3.1 KNN (K-Nearest Neighbors) Algorithm

Step 1: Find the K-most similar instance from the given data set when prediction is required.

Step 2: The similarity between real value data is measured using Euclidean distance and Humming distance is used for categorical or binary data.

Step 3: Return average predicted attributes from the classification.

## 3.2  Naive Bayes Algorithm

The Naive Bayes is one of the supervised learning models for making a prediction using the probabilities of individual attributes from each class.

Step 1: Probability calculation is simplified by considering the probability of each attribute from the given class is independent of other attributes.

Step 2: The product rule is applied on all conditional probabilities for each attribute for a given class value.

Step 3: Conditional probability for each class attribute is derived using Bayes rule and prediction is made by using the class value with the highest probabilities.

## 3.3  Linear Regression Algorithm

Linear regression algorithm is used to count the water quantity requirement and the prediction for the next water cycle. Multiple linear regression is used when there are more input variables which affects the output, as shown in Eq. (1).

$$Y = \beta 0 + \beta 1 x 1 + \beta 2 x 2 + \cdots + \beta k x k + \in \tag{1}$$

## 3.4  Artificial Neural Network (ANN)

ANN algorithm is implemented for on-off prediction. Figure 3 shows the Irrigation System performed with ANN Controller.

**Input from Sensors**: The different parameters like temperature and soil moisture are collected in this stage. Then these parameters are used as the input for the next stage.

**Evapotranspiration Model**: Both input parameters are converted into actual soil moisture.

**Required Soil Moisture**: Provides information about the quantity of water required for the proper growth of plants.

**ANN Controller**: Decision is made dynamically based on the comparison of required with actual soil moisture.
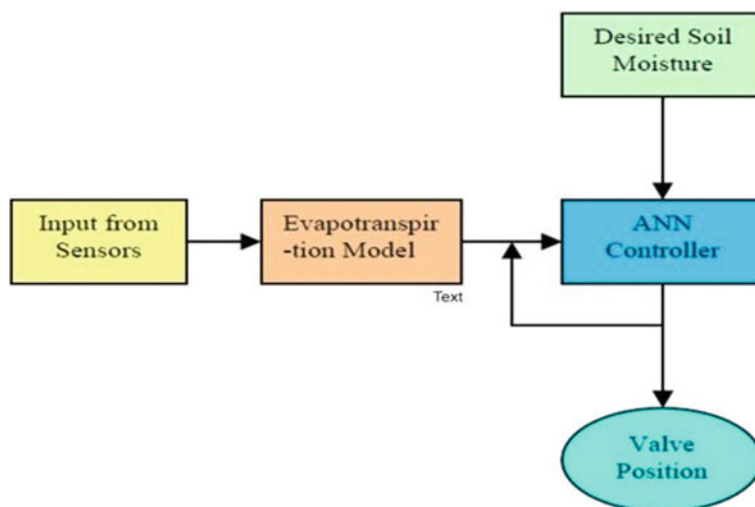
**Fig. 3** ANN-based irrigation control system

## 4   Results and Discussion

The proposed system consists of a watering system for plant farming that includes the formers being satisfied by automatic water supply according to their crops. This system provides water according to requirement and activates relay motor automatically.

### 4.1   Dataset Preparation

The real-time data is collected using IOT sensors and processed using a trained model of machine learning algorithms like KNN and Naive Bayes, the characteristics of sensor values are shown in Fig. 4.

### 4.2   Prediction of Dryness of Soil

The experiment was conducted using Arduino board with humidity and temperature sensors. The connections were established between the sensors and Arduino board. The dryness of the soil is calculated using machine learning algorithms such as K-means and Naive bayes algorithm. Table 1 shows the performance of the machine learning algorithm. K-means algorithm produced an accuracy of 91.17% and Naive

**Fig. 4** Data measured from the IOT model

**Table 1** Performance of machine learning algorithm

| S. no | Machine learning algorithm | Performance |
|-------|----------------------------|-------------|
| 1     | KNN                        | 91.17%      |
| 2     | Naive Bayes                | 81.43%      |

Bayes algorithm produced an accuracy of 81.43%. K-means produces high accuracy while comparing with Naive Bayes shown in Fig. 5.
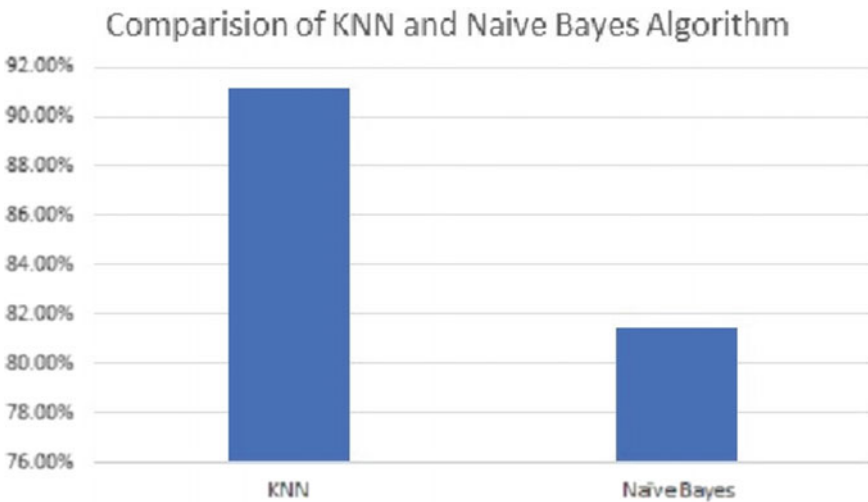


**Fig. 5** Comparison of KNN and naive Bayes algorithm

Water needed after=13.7696181388101

Water needed=2335.2153987167735 liters
Duration of Irrigation=11.12007332722273 Min

(i) Finding water requirements

(ii) Amount of water needed and duration of irrigation
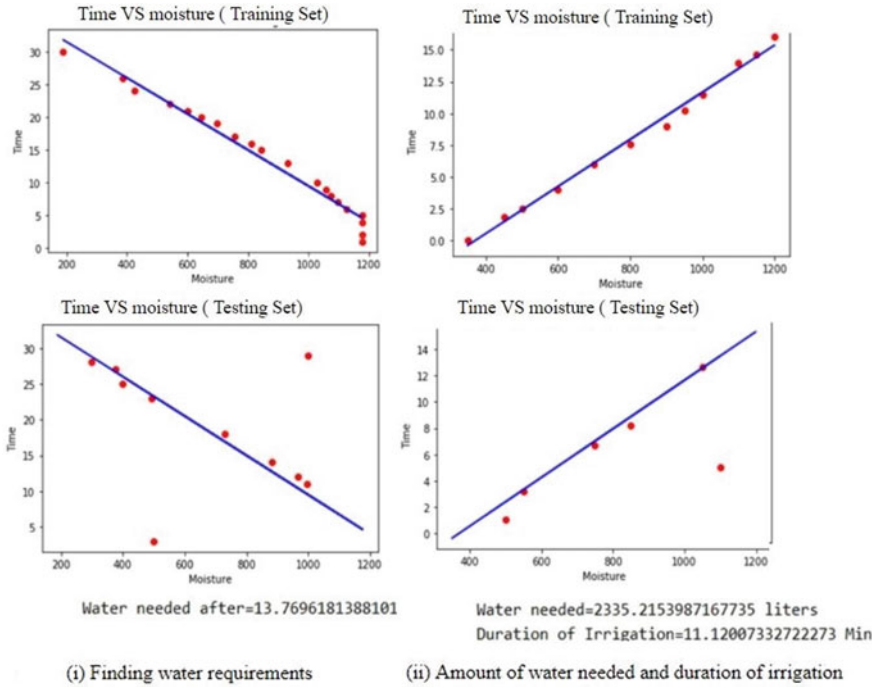
**Fig. 6** Prediction of water requirement and next water cycle

## 4.3 Finding Water Quantity Requirement and Prediction for the Next Water Cycle

Linear regression algorithm used to find the requirements of water quantity and predict when the water is required again. It predicted the water needed after 13.76 days based on the values of moisture level in the dataset and then the amount of water needed, and irrigation intervals are shown in Fig. 6.

## 4.4 Prediction of Pump ON/OFF

Figure 7 shows the prediction of pump ON/OFF based on the ANN model it contains an input layer, four hidden layers, and output layer. It performs Relu operation. It provides an accuracy of 97.22% and loss of 19.68%.

(i) Correlation matrix          (ii) scatter and density plot
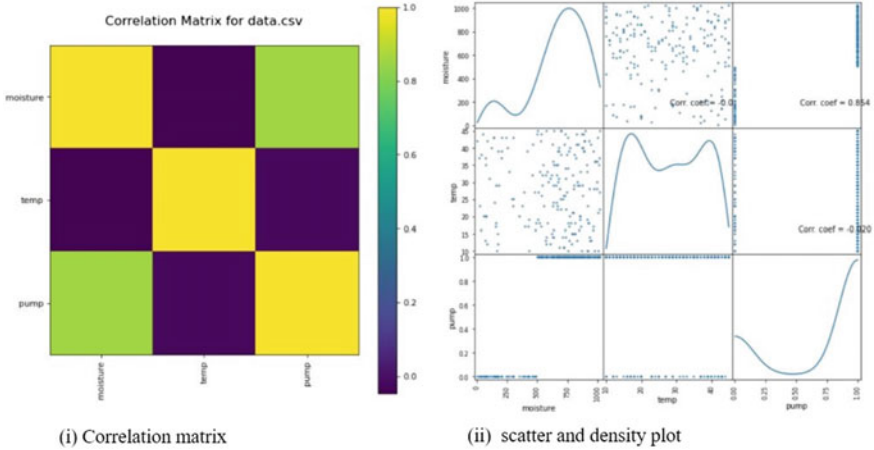
**Fig. 7** Prediction of pump ON/OFF

## 5 Conclusion and Future Enhancements

We proposed an automatic irrigation system in three modules. In the first module, we constructed an IOT model. We used the sensors like soil moister sensor and dht11 sensor and NodeMCU and collected corresponding values. In the second module, we applied machine learning algorithms like KNN and Naive Bayes model for identifying the dryness of the soil and linear algorithm is applied for identifying the amount of water required. In the third module, we used artificial neural network algorithm for pumping ON/OFF.

In future we have the plan to integrate the system along with rain forecasting, which leads to more effective utilization of power and water.

## References

1. Seenu (2018) Android based intelligent irrigation system. Journal
2. Umair M (2015) Automation of irrigation system using ANN-based controller. Int J Electr Comput Sci IJECS-IJENS
3. Gu J (2017) An improved back propagation neural network prediction model for sub-surface drip irrigation system. Elsevier Ltd, Computers and Electrical Engineering
4. Koprda S (2017) A low cost irrigation system with Raspberry Pi—own design and statistical evaluation of efficiency. Journal
5. Praveen (2019) Based multiple sensor (DHT11, Soil moisture sensor) monitoring system. IJRAR
6. Roopaer M (2017) Cloud of things in smartagriculture: intelligent irrigation moni- toring by thermal imaging. IEEE Cloud Comput
7. Hassan A (2021) A wirelessly controlled robot-based smart irrigation system by exploiting Arduino. J Robot Control (JRC)

8. Jamroen C (2020) An intelligent irrigation scheduling system using low-cost wireless sensor network toward sustainable and precision agriculture. IEEE Ac- cess
9. Akte S (2019) Developing a smart irrigation system using Arduino. Int J Res Stud Sci Eng Technol V6

# Water Quality Risk Analysis for Sustainable Smart Water Supply Using Adaptive Frequency and BiLSTM

**Uma N. Dulhare and Syeda Talha Ali Taj**

## 1 Introduction

Water is a vital commodity for use in households, industries, and agriculture. Growing population and global ecosystem degradation have put compelling stress on fulfilling the natural resources associated with energy, water, and food. Water quality is usually measured in terms of chemical characteristics, physical characteristics, and biological characteristics such as E.coli, Coliform, etc., as shown in Fig. 1 based on the standards of its usage. The quality of water is classified into four types—potable water which is safe for drinking and is used for everyday activities; palatable water has a chemical present in it, which is not harmful to the health; contaminated water which consists of impurities harmful for domestic use; and infected water which is unfit to drink. The necessity for water quality differs based on its usage. The requirement for water quality needs to comply with the standards set up by the government. The World Health Organization (WHO) has established standards for water quality to ensure safe drinking water that all countries are supposed to comply with.

Physical indicator constitutes temperature, turbidity, etc. Chemical indicator involves pH, dissolved oxygen, Mg, fluorine, etc. Biological indicator includes algae, phytoplankton, flora, and fauna. These parameters are not only confined to surface water studies but also to groundwater and industrial processes. With the advancements in technology, the researchers have aimed towards the utilization of machine learning techniques to analyze the water quality as well as to predict the risks associated with them.

Advanced ubiquitous sensing technologies have also been used in various researches including industries and daily life. In today's world, numerous sensors are used to collect zettabytes of data in order to perform stronger data analysis. As the quality indicators of water are spatiotemporal variables. For spatiotemporal variables,

U. N. Dulhare · S. T. A. Taj (✉)
Muffakham Jah College of Engineering and Technology, Hyderabad, India
e-mail: talhaalitajsyeda@gmail.com
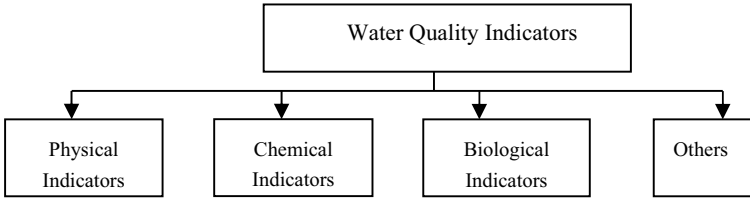
**Fig. 1** Classification of water quality indicators

either correlation analysis or numerical prediction analysis is used and, for multiple variable analysis, Principal Component Analysis (PCA) is usually used. The kernel correlation analysis was used by Hardoon et al. [1] for web page image analysis and associated text analysis. The classical PCA and other new algorithms were reviewed by Jolliffe et al. [2]. In gait recognition, a tensor flow model was applied in correlation analysis without considering the correlations in the time domain by Luo et al. [3].

The most desirable range of pH for drinking water lies between 6.5 and 8.5. The range of total dissolved solids is considered excellent if it is less than 300 mg/l and it is considered unacceptable if it lies above 1200 mg/l. The value of temperature must be 20 °C/68° F for normal drinking water and $6^0$ C/$43^0$ F for cool drinking water. The range of conductivity for drinking water must lie between 50 and 800 $\mu$S, the acceptable value for nitrate is 10 mg/l, and the acceptable range for coliform must lie below 1 per liter. Incompliance to the various parameter ranges for safe drinking water might lead to severe and drastic health issues such as chronic health disease, cholera, typhoid, jaundice, and others [4].

This research proposes a BiLSTM model to analyze the water quality by taking into consideration the threshold limits set up for each indicator by the WHO in order to prevent the spread of contagious diseases and to check the efficiency of the developed model it is utilized on two different datasets.

## 1.1 Objective

The major contribution of this work is to analyze the risk associated with water quality in order to curb the changing trends in water parameters and spread of contagious diseases. The rudimentary objective is to analyze the water quality risk and support decision-making in water supply systems by considering all the four indicators (biological, physical, environmental, and chemical) including frequency and time domains. The study proposes a Bidirectional Long Short-Term Memory (BiLSTM) model for analyzing and predicting the risks of water quality. This work utilizes real-time industrial data collected from different cities and states. Finally, the results obtained are compared with Artificial Neural Network (ANN) and Adaptive Frequency Analysis (Adt-FA) algorithm. The statistical metrics are calculated for each algorithm for result analysis.

## 2 Literature Survey

Approximately 681 million individuals around the world are attempting to get safe drinking water as per the WHO report [5]. The quality of water is a vital factor for all the processes in the DWS system. Yagur-Kroll et al. [6] utilized certain sensors for biological parameters monitoring in 2015. For the quality prediction of water, an artificial neural network model for analyzing the salinity level in the River Murray, Australia, was proposed by Maier et al. [7]. However, this entire work focused on a specific water quality parameter for prediction, neglecting the interrelationship between them. F.J. Chang et al. utilized systematic analysis schema, and the model was developed for water quality prediction using two static ANNs (i.e., backpropagation neural network and artificial neuro-fuzzy inference system) and one dynamic ANN (NARX) combined with Gamma Test (GT) using hydrological and economic data [8]. The study was carried out on the Dahan River, Taiwan, for nine water quality parameters: acidity (pH), electroconductivity (EC), dissolved oxygen (DO), biochemical oxygen demand (BOD), chemical oxygen demand (COD), suspended solid (SS), coliform, and ammonia nitrogen ($NH_3$-N). The results demonstrated that these three models fit efficiently the variability of $NH_3$-N concentration. This model was designed specifically for the detection of ammonium nitrate and neglected the relationship existing between different indicators. Hadi Mohammed et al. built an adaptive neuro-fuzzy inference system (ANFIS) [9]. This model was built to predict the overall amount of norovirus in water considering pH, turbidity, conductivity, and temperature. The study was conducted on the Nødre Romrike, Oslo, Norway. The results obtained by the model have shown high accuracy of prediction for the presence of norovirus. The developed model was limited to single parameter identification and neglected several major water quality indicators. D. Wu et al. proposed a smart data analysis method for biological indicators [10]. A Zero-Inflated Poisson Regression (ZIP) model was built for risk prediction. The Zero-Inflated Negative Binomial Regression (ZINB) model was also built. The research considered 11 indicators in total, and they are 4 physical indicators (temperature, conductivity, color, and turbidity), 2 chemical indicators, 4 microbial indicators (coliform, E.coli, Int.coli, Clostridium perfringens), and time. The study was limited only to biological indicator prediction and the results obtained weren't reliable and the prediction of water quality was done based on biological indicators only. Bozorg-Haddad et al. utilized two data-driven methods—genetic programming (GP) and least-squares regression (LSSVR) [11]. The parameters were selected from the Sefidrood River, Iran. The models were fit considering sodium, potassium, magnesium, phosphate, chlorine, pH, EC, and Total dissolved solids (TDS). The accuracy of the LSSVR is dependent on the coefficient values and has obtained better accuracy compared to the GP. This method is limited to a few parameters and the approach didn't consider the interrelationship between different parameters making the model to be less reliable for utilizing in real-time monitoring systems. Mahmoudi et al. have proposed the SVR to analyze the water quality [12]. The area considered here for study is the

Astane station, Sefidrood River, which covers the approximate length of 670 km. The GP and SFLA-SVR model were applied for Na, K, Mg, sulfates, Cl, pH, EC, and TDS and then the models are compared for their prediction accuracy. The RMSE values obtained from the hybrid model have improved for training and testing with regards to GP. This model has been critical to parameter selection and distorted relationships between various indicators of the quality parameters. Najfzadeh has done research using GEP, EPR, and MT to estimate three indicators [13]. To develop the proposed models, input parameters that have been considered are calcium, sodium, magnesium, $NO_2^-$, $NO_3^-$, $PO_4^{3-}$ EC, PH, and turbidity. Karun River is the focus of the study. The results demonstrated that the EPR has shown superiority over the other two models. The performance evaluation parameters RMSE, $R^2$, MAPE were used and the EPR model has shown efficient results. This study was limited to few parameters and unacknowledged the relationship between them. The LSTM NN is proposed by Wang et al. for water quality prediction [14]. The dataset from Taihu Lake is used for the model training. The series of simulations are performed. The models were built to predict total phosphorus (TP) and DO. The model built is compared with the Backpropagation Neural Network (BP NN). Another model online sequential extreme learning machine (OS ELM) was also used for comparison with LSTM. The LSTM model has gained much better accuracy than the other two models. The limitation it possessed was time-consuming since it takes lots of time for model training and is limited to only two parameters. Liu et al. analyzed the quality of water by considering pH, dissolved oxygen, and temperature [15]. The Pearson correlation coefficient method is applied for correlation analysis. The study was carried out on the Xincun Town, China. The deep Bi-S-SRU learning network is utilized. Bi-S-SRU is a two-step calculation process: forward pass and backward pass. The results showed that this method has obtained better prediction accuracy than the RNN or LSTM model. The drawback of this study is it focused only on the biological parameters for prediction.

By examining the background and related work critically, this study focuses on identifying the risks by considering the physical indicators, the environmental indicators, the biological indicators, and the chemical indicators with a total of 19 parameters along with the time and frequency domain in order to prevent the spread of contaminated diseases. It utilizes two different datasets collected from two different countries for checking the efficient functioning of the system. The present study attempts to model a system using BiLSTM by taking into consideration various indicators along with their interrelationship and the threshold values setup by the W.H.O. for each indicator. This study is considered to be one of its kinds that consider several major water quality parameters belonging to various indicators that were neglected in the previous research.

## 3  Data Description

The dataset utilized for the study consist of biological indicators (Coliform, Fecal Coli., Int Coli., Termotol Coliform), Physical indicators (pH, Turbidity, Conductivity, Color, Temperature), Chemical indicators (Alkalinity, B.O.D, Nitrate, D.O) including the geographical indicators. The sample snapshot of the dataset 1 is as shown in Fig. 2.

The 19 indicators that are used for study are.

- Physical indicators

    - Turbidity
    - Conductivity
    - Color
    - Temperature

- Chemical Indicators

    - pH
    - Alkalinity
    - Biological Oxygen Demand (BOD)
    - Nitrate
    - Dissolved Oxygen (DO)

- Biological indicators

    - Escherichia coli (E. coli)
    - Intestinal enterococci (Int)
    - Coliform
    - Clostridium perfingens (CIPerf)
    - Termotol Coliform

| STATIONC | LOCATION | STATE | MONTH | Temp | D.O._(mg) | PH | CONDUCT | TURBIDITY | COLOR_(n | ALKALINIT | B.O.D._(m | NITRATE | FECAL_CO | Int_(cfu/1 | CIPerf_(cf | TERMOTO | TOTAL_CC | year |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1393 | MARIDAS¹ | OSLO | JAN | 30.6 | 6.7 | 7.5 | 203 | 0.96 | 5 | 4.4 | 0 | 0.1 | 11 | 0.12 | <1 | <1 | 27 | 2015 |
| 1399 | MARIDAS¹ | OSLO | JAN | 29.8 | 5.7 | 7.2 | 189 | 0.94 | 9 | 4.4 | 2 | 0.2 | 4953 | 0.63 | <1 | <1 | 8391 | 2015 |
| 1475 | MARIDAS¹ | OSLO | JAN | 29.5 | 6.3 | 6.9 | 179 | 0.93 | 5 | 5.7 | 1.7 | 0.1 | 3243 | 0.65 | <1 | <1 | 5330 | 2015 |
| 3181 | MARIDAS¹ | OSLO | JAN | 29.7 | 5.8 | 6.9 | 64 | 0.93 | 5 | 6 | 3.8 | 0.5 | 5382 | 0.66 | <1 | <1 | 8443 | 2015 |
| 3182 | MARIDAS¹ | OSLO | FEB | 29.5 | 5.8 | 7.3 | 83 | 0.95 | 8 | 6.3 | 1.9 | 0.4 | 3428 | 0.67 | <1 | <1 | 5500 | 2015 |
| 1400 | MARIDAS¹ | OSLO | FEB | 30 | 5.5 | 7.4 | 81 | 1.08 | 12 | 4.2 | 1.5 | 0.1 | 2853 | 0.63 | <1 | <1 | 4049 | 2015 |
| 1476 | MARIDAS¹ | OSLO | FEB | 29.2 | 6.1 | 6.7 | 308 | 0.9 | 20 | 7.1 | 1.4 | 0.3 | 3355 | 0.63 | <1 | <1 | 5672 | 2015 |
| 3185 | MARIDAS¹ | OSLO | MAR | 29.6 | 6.4 | 6.7 | 414 | 1 | 20 | 8.1 | 1 | 0.2 | 6073 | 0.64 | <1 | <1 | 9423 | 2015 |
| 3186 | MARIDAS¹ | OSLO | MAR | 30 | 6.4 | 7.6 | 305 | 0.92 | 20 | 7.5 | 2.2 | 0.1 | 3478 | 0.7 | <1 | <1 | 4990 | 2015 |
| 3187 | MARIDAS¹ | OSLO | APR | 30.1 | 6.3 | 7.6 | 77 | 0.91 | 7 | 6.5 | 2.3 | 0.1 | 2606 | 0.59 | <1 | <1 | 4301 | 2015 |
| 1543 | | OSLO | APR | 27.8 | 7.1 | 7.1 | 176 | 1 | 6 | 6.6 | 1.2 | 0.1 | 4573 | 0.59 | <1 | <1 | 7817 | 2015 |
| 1548 | MARIDAS¹ | OSLO | APR | 27.9 | 6.7 | 6.4 | 93 | 0.98 | 8 | 6.3 | 1.4 | 0.1 | 2147 | 0.65 | <1 | <1 | 3433 | 2015 |
| 2276 | | OSLO | MAY | 29.3 | 7.4 | 6.8 | 121 | 1.02 | 19 | 5.4 | 1.7 | 0.4 | 11633 | | <1 | <1 | 18125 | 2015 |
| 2275 | MARIDAS¹ | OSLO | JUN | 29.2 | 6.9 | 7 | 620 | 0.96 | 12 | 2.2 | 1.1 | 0.1 | 3500 | 0.68 | <1 | <1 | 6300 | 2015 |
| 3189 | MARIDAS¹ | OSLO | JUN | 30 | 6 | 7.5 | 72 | 0.94 | 11 | 5.2 | 1.6 | 0.2 | 4995 | 0.17 | <1 | <1 | 9517 | 2015 |
| 1546 | MARIDASVANNET | | JUL | 29 | 7.3 | 7 | 247 | 1.06 | 13 | 5.6 | 1.5 | 0.2 | 1095 | 0.68 | <1 | <1 | 2453 | 2015 |
| 2270 | | OSLO | JUL | 29.1 | 7.3 | 7 | 188 | 0.86 | 23 | 5.5 | 1 | 0.1 | 1286 | 0.67 | <1 | <1 | 3048 | 2015 |
| 2272 | MARIDAS¹ | OSLO | JUL | 28.7 | 7 | 6.9 | 224 | 0.95 | 17 | 3.6 | 1.2 | 0.3 | 3896 | 0.64 | <1 | <1 | 6742 | 2015 |
| 1545 | MARIDAS¹ | OSLO | AUG | 28.7 | 7.3 | 6.7 | 144 | 0.85 | 14 | 5.7 | 1.5 | 0.1 | 1940 | 0.6 | <1 | <1 | 3052 | 2015 |
| 2274 | | OSLO | AUG | 29.5 | 5.3 | 6.8 | 319 | 0.93 | 15 | | 1.8 | 0.3 | 6458 | 0.6 | <1 | <1 | 10250 | 2015 |
| 2271 | | OSLO | SEP | 29 | 6.3 | 6.4 | 79 | 0.85 | 8 | | 1.6 | 1.4 | 7592 | 0.62 | <1 | <1 | 12842 | 2015 |
| 2273 | MARIDAS¹ | OSLO | SEP | 29.4 | 5.4 | 7.6 | 39 | 0.93 | 5 | | 1.4 | 0.1 | 3176 | 0.64 | <1 | <1 | 6367 | 2015 |

**Fig. 2**  Snapshot of the dataset 1

| STATION C | LOCATION | STATE | Temp | D.O. (mg/l | PH | CONDUCT | B.O.D. (mg | NITRATE0 | FECAL COL | TOTAL CO | year |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1393 | DAMANGA | DAMAN & | 30.6 | 6.7 | 7.5 | 203 | 0 | 0.1 | 11 | 27 | 2014 |
| 1399 | ZUARI AT I | GOA | 29.8 | 5.7 | 7.2 | 189 | 2 | 0.2 | 4953 | 8391 | 2014 |
| 1475 | ZUARI AT I | GOA | 29.5 | 6.3 | 6.9 | 179 | 1.7 | 0.1 | 3243 | 5330 | 2014 |
| 3181 | RIVER ZUA | GOA | 29.7 | 5.8 | 6.9 | 64 | 3.8 | 0.5 | 5382 | 8443 | 2014 |
| 3182 | RIVER ZUA | GOA | 29.5 | 5.8 | 7.3 | 83 | 1.9 | 0.4 | 3428 | 5500 | 2014 |
| 1400 | MANDOVI | GOA | 30 | 5.5 | 7.4 | 81 | 1.5 | 0.1 | 2853 | 4049 | 2014 |
| 1476 | MANDOVI | GOA | 29.2 | 6.1 | 6.7 | 308 | 1.4 | 0.3 | 3355 | 5672 | 2014 |
| 3185 | RIVER MAI | GOA | 29.6 | 6.4 | 6.7 | 414 | 1 | 0.2 | 6073 | 9423 | 2014 |
| 3186 | RIVER MAI | GOA | 30 | 6.4 | 7.6 | 305 | 2.2 | 0.1 | 3478 | 4990 | 2014 |
| 3187 | RIVER MAI | GOA | 30.1 | 6.3 | 7.6 | 77 | 2.3 | 0.1 | 2606 | 4301 | 2014 |
| 1543 | RIVER KAL | GOA | 27.8 | 7.1 | 7.1 | 176 | 1.2 | 0.1 | 4573 | 7817 | 2014 |
| 1548 | RIVER ASS | GOA | 27.9 | 6.7 | 6.4 | 93 | 1.4 | 0.1 | 2147 | 3433 | 2014 |
| 2276 | RIVER BICI | GOA | 29.3 | 7.4 | 6.8 | 121 | 1.7 | 0.4 | 11633 | 18125 | 2014 |
| 2275 | RIVER CHA | GOA | 29.2 | 6.9 | 7 | 620 | 1.1 | 0.1 | 3500 | 6300 | 2014 |
| 3189 | RIVER CHA | GOA | 30 | 6 | 7.5 | 72 | 1.6 | 0.2 | 4995 | 9517 | 2014 |
| 1546 | RIVER KHA | GOA | 29 | 7.3 | 7 | 247 | 1.5 | 0.2 | 1095 | 2453 | 2014 |
| 2270 | RIVER KHA | GOA | 29.1 | 7.3 | 7 | 188 | 1 | 0.1 | 1286 | 3048 | 2014 |
| 2272 | RIVER KUS | GOA | 28.7 | 7 | 6.9 | 224 | 1.2 | 0.3 | 3896 | 6742 | 2014 |
| 1545 | RIVER MAI | GOA | 28.7 | 7.3 | 6.7 | 144 | 1.5 | 0.1 | 1940 | 3052 | 2014 |
| 2274 | RIVER MAI | GOA | 29.5 | 5.3 | 6.8 | 319 | 1.8 | 0.3 | 6458 | 10250 | 2014 |

**Fig. 3** Snapshot of the dataset 2

- Environmental indicators

  - Location
  - Station
  - State
  - Month
  - Year

The sample snapshot of the dataset 2 is shown in Fig. 3. The dataset 2 consists of 12 parameters that belong to different categories of water quality indicators.

## 4 System Architecture

The architecture of the system designed for prediction of risk is as shown in Fig. 4. Broadly, it consists of six steps: data collection, data pre-processing, feature selection, building water quality risk recognition models, risk evaluation, and risk prediction. Various machine learning concepts are being used in each step [16]. A detailed explanation of each step is discussed further.

**Data Collection**

In this step, the raw real-time data consisting of noise, missing values, etc. is collected from the water sources from several water supply systems.

For indicator domain, we considered indicators as physical: conductivity, turbidity, temp, color, for chemical indicators: pH, DO, Alkalinity, B.O.D, Nitrate and biological: Coliform, Ecoli, Int, ClPerf, Termotol coliform.
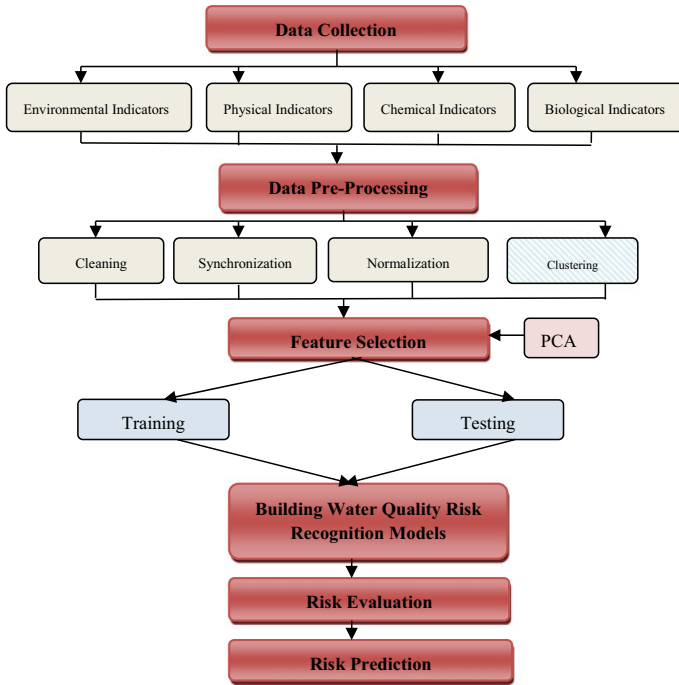
**Fig. 4** System architecture for water quality risk prediction

As the time varies from city to city for the time domain, the yearly data was collected from different cities and states of two different countries which also constitutes to geographical domain.

**Data Pre-processing**

As the data collected usually consists of missing values, out-of-range values, noise, etc. Therefore, data pre-processing transforms the improper data to computable format by cleaning, synchronizing, and normalizing. Clustering and declustering can also be done but is optional.

- Cleansing is done to remove the values that are of no use and creates a mess while analysing using a model as it consists of missing values, out-of-range values, etc. Hence, this step is needed.
- Synchronization is to be done since the indicators collected are different in a hybrid frequency environment.
- To deal with diverse units normalization is done. It is the process of rescaling the attributes to the range of 0 or 1. In this study, z- score normalization is used by transforming all the parameter values of water quality to have a mean of 0 and a standard deviation of 1.

**Z-Score Normalization**

We calculate the z-score by deducting the mean of the population from a raw data value divided by the standard deviation as shown in (1), resulting ideally in between $-3$ and $+3$.

$$z\text{-}score = \frac{(x - \mu)}{\sigma} \tag{1}$$

where, x represents the value of a specific sample, $\mu$ represents the mean and $\sigma$ represents the standard deviation.

In this study, k-means clustering is being used which partitions n observation into k clusters. The k-means algorithm works as follows.

1. Initialize randomly k points.
2. Classify each item to its closest mean and update the mean's coordinates.
3. Repeat it for a given number of iterations and clusters are obtained as an end result.

**Feature Selection**

Once the data is pre-processed we proceed ahead with the feature selection. Principal Component Analysis is being used for feature selection in our study. PCA is generally used to reduce the dimensionality of the data while retaining as much of the information as possible. The main objective of PCA is to reduce the variables for prediction and thereby detecting the structure that might exist between the variables. In this, the p initial variables $(x_1, x_2, \ldots \ldots x_n)$ linear combinations are created to produce $(PC_1, PC_2, \ldots \ldots PC_n)$ [12]. Each of these components is expressed by the equation below

$$PC_1 = w_{11}x_1 + w_{12}x_2 + \ldots \ldots w_{1n}x_n$$

$$PC_2 = w_{21}x_1 + w_{22}x_2 + \ldots \ldots w_{2n}x_n$$

$$PC_n = w_{n1}x_1 + w_{n2}x_2 + \ldots \ldots w_{nn}x_n \tag{2}$$

In (2),
$PC_i$ = principal component of ith term,
$W_{ij}$ = principal component coefficient of ith term and jth initial variable, and
$x_i$ = initial variable of ith term.

**Building Water Quality Risk Recognition Models**

An BiLSTM model is built for water quality risk analysis and prediction considering varying parameters of different indicators on two different datasets then an ANN model and an Adaptive frequency analysis model is generated for comparison with the proposed model.

**Bidirectional Long Short-Term Memory (BiLSTM)**

Bidirectional Long-short-term memory is a special kind of Recurrent Neural Network (RNN) as it overcomes vanishing gradient and explosive gradient limitations of RNN. The BiLSTM model processes the data in both forward and backward direction. The RNN takes the form of chained modules recursively of the neural network whereas LSTM uses a purpose-built memory cell for storing information that are chained in a similar structure, but each of these modules is structured differently, a study done by Liu [17]. The equation computations done by each cell are as follows:

1. Forget gate layer.

$$F_t = \sigma(w_t[S_{t-1, x_t}] + b_f) \tag{3}$$

2. Input gate layer.

$$I_t = \sigma(w_t[S_{t-1, x_t}] + b_I) \tag{4}$$

3. New memory cell.

$$\sim N_t = \tanh(w_t[S_{t-1, x_t}] + b_F) \tag{5}$$

4. Final memory cell.

$$N_t = \sigma(F_t * N_{t-1} + I_t * \sim N_t) \tag{6}$$

5. Output gate layer.

$$O_t = \sigma(w_0[S_{t-1, x_t}] + b_0) \tag{7}$$

$$S_t = O_t + \tanh(N_t) \tag{8}$$

In this study, a BiLSTM model was designed to predict the water quality using Keras and Tensorflow by means of python 3.7. The structure of LSTM cell is shown in Fig. 5. The flowchart of proposed BiLSTM model is shown in Fig. 6
The workflow of the BiLSTM model is as follows.
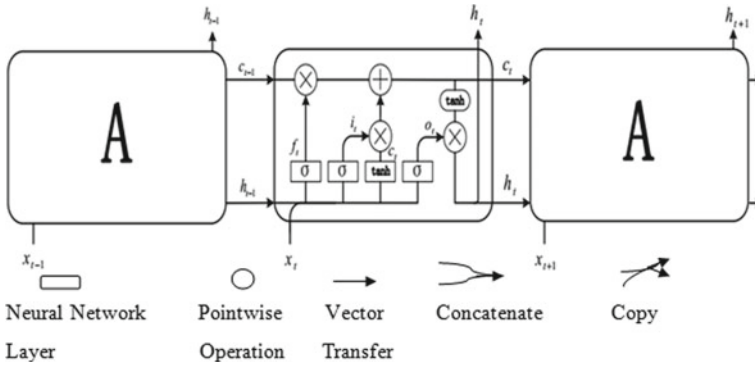
**Fig. 5** Structure of LSTM

**Fig. 6** Flowchart of
proposed BiLSTM model
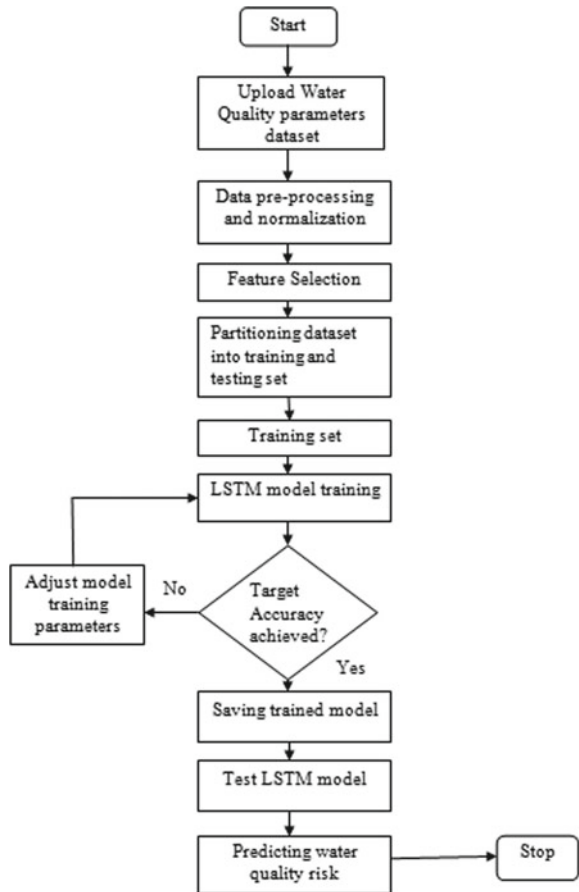
1. Transforming and loading the CSV file of water quality data to a pandas dataframe.
2. BiLSTM model building.
3. Model Training.
4. Running a trained model.
5. Testing the trained model.
6. Making water quality risk predictions using BiLSTM.

After developing the BiLSTM model, the efficiency of the developed model on two different datasets is evaluated and the results are then compared with the ANN and Adaptive frequency analysis models.

**Risk Evaluation**

The statistical evaluation metrics such as accuracy, Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R Square ($R^2$) are used.

**Accuracy**

The accuracy is calculated for each algorithm. The formula to calculate accuracy is as follows

$$Accurancy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ observations\ taken\ into\ consideration} \quad (9)$$

**Root Mean Square Error (RMSE)**

It is a unit of measure that has been used to assess the error rate of the algorithms and is calculated using (10).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(P_i - A_i)}{N}} \quad (10)$$

where, $P_i$ belongs to predicted values, $A_i$ stands for actual values and N stands for total number of observations.

**Mean Absolute Error (MAE)**

The mean absolute error is used to compare and measure the accuracy of the model which is being calculated for each model implemented is in the system. The lower the MAE value the better is the algorithm. The equation of calculating MAE is as given in (11).

$$MAE = \sum \frac{|y_i - x_i|}{N} \quad (11)$$

where, N is the number of observations, $x_i$ actual value, and $y_i$ predicted value.

**R Squared Error ($R^2$)**

The R squared error is used for the analysis of the result which shows how well a model fits the given dataset. It is calculated using (12).

$$R^2 - 1 = \frac{\sum(z_i - \hat{z})^2}{\sum(z_i - \overline{z})^2} \tag{12}$$

where, $\hat{z}$ is the predicted value of z, $\overline{z}$ is the mean value of z.

**Risk Prediction**

The proposed model BiLSTM is applied to predict the risk by taking into consideration all the parameters selected while feature selection and the results are generated displaying whether the risk is predicted or not. The prediction results are compared with ANN and Adt-FA [18]. The same process is repeated for both the datasets.

## 5 Result Analysis

The result analysis is done considering accuracy, RMSE, MAE, and $R^2$. The model with highest accuracy is considered to be the best suitable model for water quality risk prediction. The accuracy is calculated for each algorithm using formula (9) and the comparison graph is shown in the Fig. 7 were one can see that BiLSTM has gained highest accuracy compared to Adaptive Frequency Analysis and ANN for water quality risk prediction.
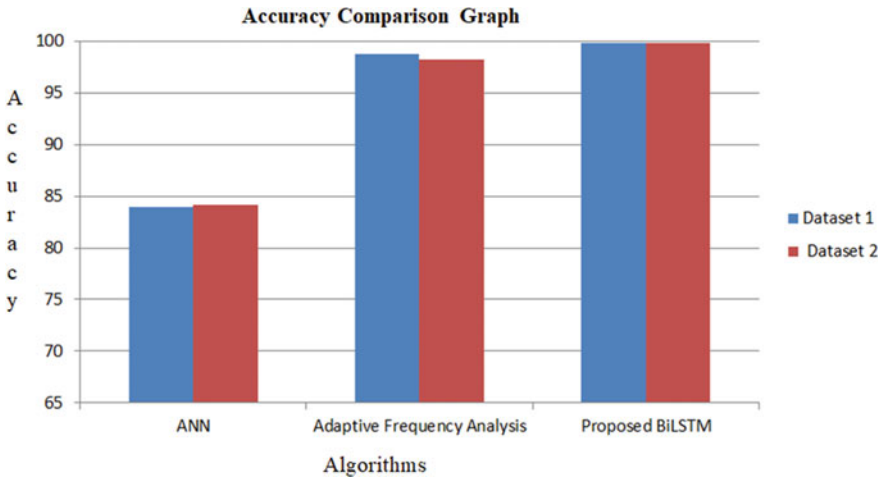


**Fig. 7** Accuracy of different algorithms

The RMSE values and graph for the algorithms is given in Table 1 and Fig. 8. The lower the RMSE error rate the better the model for risk prediction. From the Fig. 8 one can depict that BiLSTM has obtained lowest error rate than other two algorithms. The BiLSTM model has obtained better RMSE error rate for both the datasets.

The MAE for each algorithm is calculated and listed in the Table 1. The graph representing the MAE and algorithms is shown in Fig. 9. The model with lowest MAE is considered to be best fit for water quality risk prediction. From the results obtained BiLSTM has shown lowest MAE value compared to ANN and Adt- FA.

The calculated values of $R^2$ are listed in Table 1. The comparison graph representing the R square error values and algorithms is shown in Fig. 10. The model with highest $R^2$ value is considered to be the most efficent model. From the results it is observed that BiLSTM has shown better $R^2$ value compared to ANN and Adt-FA.

**Table 1** Statistical evaluation of proposed model with other algorithms

|  | Algorithm | Accuracy | RMSE | MAE | $R^2$ |
|---|---|---|---|---|---|
| Dataset 1 | ANN | 83.95 | 0.40 | 0.1604 | 0.13 |
| Dataset 2 |  | 84.21 | 0.39 | 0.157 | 0.12 |
| Dataset 1 | Adaptive Frequency Analysis | 98.74 | 0.11 | 0.0125 | 0.94 |
| Dataset 2 |  | 98.24 | 0.13 | 0.018 | 0.91 |
| Dataset 1 | Proposed BiLSTM | 99.79 | 0.05 | 0.0051 | 0.98 |
| Dataset 2 |  | 99.89 | 0.04 | 0.002 | 0.98 |
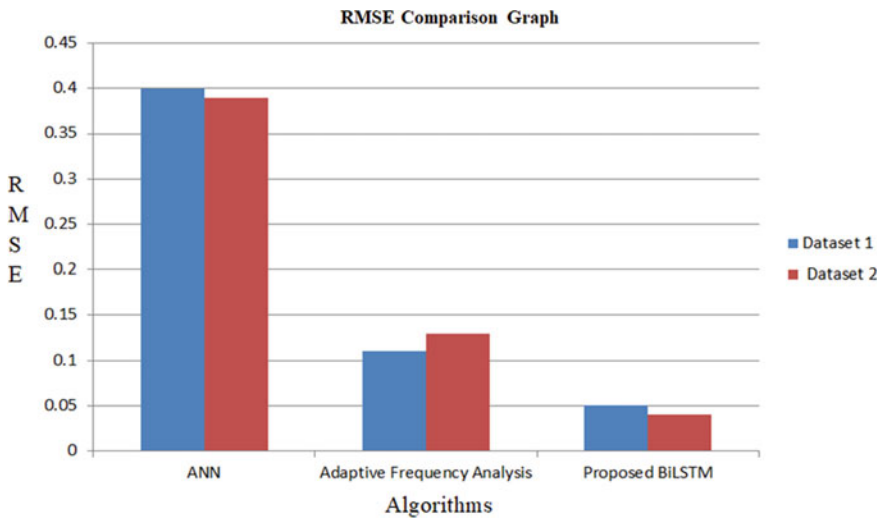


**Fig. 8** RMSE of different algorithms
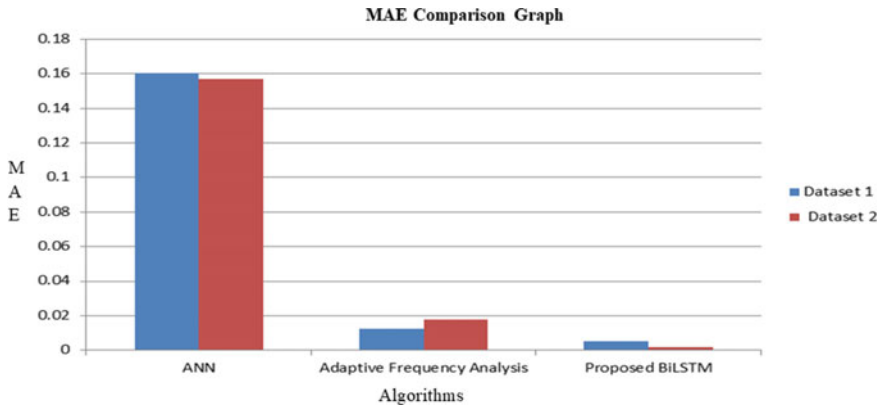
**Fig. 9** MAE of different algorithms
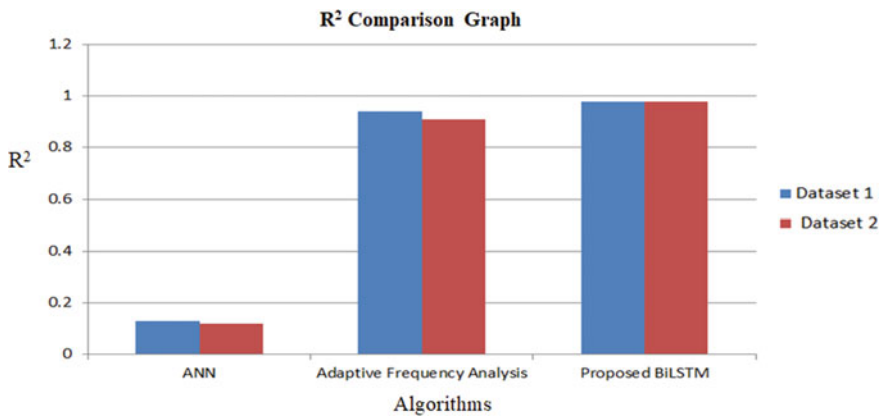


**Fig. 10** $R^2$ of the algorithms

## 6  Conclusion

The proposed work suggests the use of BiLSTM for water quality risk analysis considering biological indicators, physical indicators, environmental indicators, and chemical indicators along with frequency and time domain. The study considered major water quality parameters that belong to various indicators of water quality. BiLSTM is a long short-term memory that utilizes memory units for short-term memorization and overcomes the limitations of the classical RNN. To analyze the efficiency of the proposed model it is utilized on two different datasets collected from real-time water quality monitoring systems. The results obtained from BiLSTM are than compared with the adaptive frequency analysis algorithm, and ANN. The results were analyzed using statistical metrics such as accuracy, RMSE, MAE, and $R^2$. The

results demonstrated that the proposed BiLSTM model has obtained highest accuracy compared to Adt-FA and ANN. The proposed BiLSTM model showed better results for all the metrics used for evaluation. This model was built for tackling the spread of contagious diseases caused due to poor water quality. This model can be deployed in a real-time water quality risk monitoring system as this system showed reliable results.

For future enhancement, quality risk analysis can be computed using cost-less data analysis techniques for providing early warning mechanism from the water source areas. It can be incorporated with the real-time sensors in the water distribution systems to detect the risks and to generate the alarms indicating risks. Further, this work can be employed to real-time industrial water distribution systems in order to monitor the quality of water.

# References

1. Hardoon DR, Szedmak S, Shawe-Taylor J (2004) Canonical correlation analysis: an overview with application to learning methods. Neural Comput 16(12):2639–2664
2. Jolliffe IT, Cadima J (2016) Principal component analysis: a review and recent developments. Philosoph Trans R Soc A Math Phys Eng Sci 374(2065), Art. no 20150202
3. Luo Y, Tao D, Ramamohanarao K, Xu C, Wen Y (2015) Tensor canonical correlation analysis for multi-view dimension reduction. IEEE Trans Knowl Data Eng 27(11):3111–3124
4. Dulhare UN, Ayesha M (2016) Extraction of action rules for chronic kidney disease using Naïve Bayes classifier. In: 2016 IEEE international conference on computational intelligence and computing reasearch (ICCIC), pp 1–5
5. World Health Organization (WHO) (2004) Guidelines for drinking water quality: recommendations
6. Yagur-Kroll S, Schreuder E, Ingham CJ, Heideman R, Rosen R, Belkin S (2015) A miniature porous aluminium oxide-based flow-cell for online water quality monitoring using bacterial sensor cells. Biosens Bioelectron 64:625–632
7. Maier HR, Dandy GC (1996) The use of artificial neural networks for the prediction of water quality parameters. Water Resour Res 32(4):1013–1022
8. Chang FJ, Tsai YH, Chen PA, Coynel A, Vachaud G (2015) Modeling water quality in an urban river using hydrological factors-data Driven approaches. J Environ Manag 151:87–96. ISSN 0301-4797
9. Mohammed H, Hameed IA, Seidu R (2017) Adaptive neuro-fuzzy inference system for predicting norovirus in drinking water supply. In: International conference on informatics, health & technology (ICIHT)
10. Wu D, Mohammed H, Wang H, Seidu R (2018) Smart data analysis for water quality in catchment area monitoring. In: IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData), pp 900–908
11. Bozorg-Haddad O, Soleimani S, Loaiciga H (2017) Modeling water-quality parameters using genetic algorithm-least squares support vector regression and genetic programming. J Environ Eng 143:04017021
12. Mahmoudi N, Orouji H, Fallah-Mehdipour E (2016) Integration of shuffled frog leap algorithm and support vector regression for prediction of water quality parameters. Water Resour Manag 30(7):2195–2211
13. Najfzadeh M, Ghaemi A, Emamgholizadeh S (2019) Prediction of water quality parameters using evolutionary computing- based formulations. Int J Environ Sci Technol 16:6377–6396

14. Wang Y, Zhou J, Chen K, Wang Y, Liu L (2017) Water quality prediction method based on LSTM neural network. In: International conference on intelligent systems and knowledge engineering (ISKE)
15. Liu J, Yu C, Hu Z, Zhao Y, Bai Y, Xie M, Luo J (2020) Accurate prediction scheme of water quality in smart mariculture with deep Bi-S-SRU learning network. IEEE Access
16. Dulhare UN, Ahmad K, Ahmad KAB (2020) Machine learning and big data: concepts, algorithms, tools and applications. Wiley
17. Liu P, Wang J, Sangaiah AK, Xie Y, Yin X (2019) Analysis and prediction of water quality using LSTM deep neural networks in IoT environment. Sustainability 11:2058
18. Wu D, Wang H, Mohammed H, Seidu R (2020) Quality risk analysis for sustainable smart water supply using data perception. IEEE Trans Sustain Comput

# A Tuned Stacking Algorithm to Detect Skin Cancer Using Multi-classification Algorithms

**Rajashekar Deva and G. Narsimha**

## 1    Introduction

The power of ensemble mechanisms lies in their strengthening the unity among the poorly resulted algorithms. It strongly believes that combining weak models can generate more reliable results than traditional approaches. Any machine learning model is better explained by two extreme degrees of freedom parameters, namely "Bias" and "Variance". The model with both values as high or one of them as high is considered to be weak or base model. So, to reduce the degree of freedom, one of the better solutions is to replace the traditional with ensemble approaches. During the early days of ensemble, researchers used to combine homogenous algorithms, but the present technology is mainly focusing on the heterogeneous environment. Based on the concept of combining the algorithms, the classification of ensemble models is represented in Fig. 1.

Bagging is a process of combining homogenous algorithms, in which it learns the features from individual algorithms separately and in parallel. The major focus of the bagging algorithms is to reduce the variance of the model and it also tries to reduce the error rate by taking the mean value of the obtained results. Boosting is also a process of combining the homogenous algorithms but it considers the algorithms in sequence, tries to reduce the bias and, at the same time, also keeps the variance value as low as possible. Stacking provides a chance to integrate heterogeneous algorithms in which parts of the problem grab more attention than covering the entire problem

R. Deva (✉)
Department of CSE, Methodist College of Engineering & Technology, Hyderabad, Telangana, India
e-mail: rajshekardeva@gmail.com

G. Narsimha
JNTUH, Sulthanpur, Hyderabad, Telangana, India
e-mail: narsimha06@gmail.com
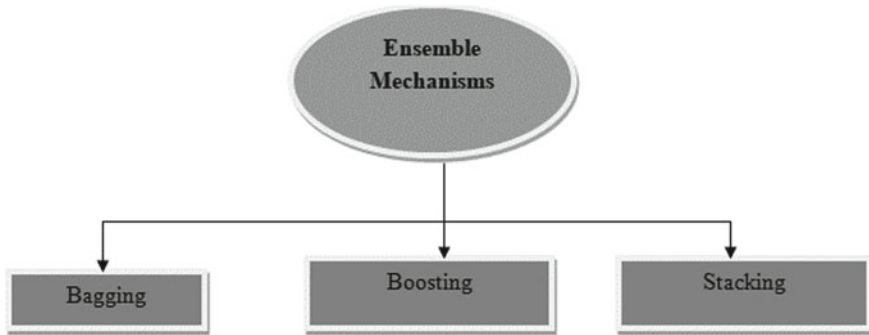
**Fig. 1** Classification of ensemble mechanisms

space. The main advantage of the stacking algorithm is it initially, cross-validates the data during the phase of training alone and each part is validated separately.

## 2 Literature Survey

According to the authors, the classification of the skin disease ESD is generally a complex issue in the field of skin sciences, as the disease is classified into six different categories. The objective of this paper is to develop a model that could distinguish these varieties with higher performance by the model using the techniques in ANN. The experimentation was conducted with the JustNN strategy with a simple yet innovative technique that follows a unique approach in executing the process. This approach is generally the first and foremost development in ANN, where the flow of data will be only in one direction [1]. AS developed, the input flows in the forward direction and the output follows that means there will be no iterations or loops. The developed model is said to be educated and verified with a pre-processed information set and had achieved a greater performance of 98.36%.

According to the author, ESD is a complex skin disease in both identifying and coping with disease. It is a critical problem that causes redness on the skin and in severe conditions the patients might lose their skin and making it more sensitive and tender. As the author stated, this disease is generally caused due to the factors like genes and environmental conditions. This disease is generally identified with six properties which are used as an indicator for this disease infection [2]. The researchers here have developed an automated machine that identifies similar symptom patients to their respective categories. The motive of this experiment is to help physicians in identifying the patients and diagnose them faster accordingly. This paper is said to propose an ideal combination of machine training algorithms, namely the Derm2Vec framework. This framework has the algorithms of autoencoders and DNN algorithms along with convolutional DNN for differentiating purpose. The authors have

performed a comparative model to check the developed system's performance where the proposed Derm2Vec showed greater results with 91.92% of perfection.

From the author of this research work, ESD, a skin problem was a complex disease and diagnosing it was also difficult due to its similar structural properties. Thus, it repeatedly leads to conflicting outcomes when done manually. Furthermore, a diagnosis was made based on visual characteristics inculcated that are related to the doctor's competence. The ontology structure was thus thought to provide an explicit depiction of the links between the condition, the indicators, and the clinical history for the prognosis of ESD by data analysis techniques [3]. However, in order to have a clear ontology, classification accuracy must be high. The appropriate collection of ideal qualities needed to forecast ESD must be identified. This paper proposed an approach in finding out the non-dependent attributes by performing a gain-based selection in diagnosing ESD. The experiment was conducted over the 19 selected features from the information using the AdaBoost algorithm, that uses the J48 classifier as its standard classifier and yields the best performance value of 95.1% The conducted experiment carried out both with and without selecting the features as a comparative study with RF, RT, and REPtree models.

This paper proposed a new framework for identifying the patients with ESD using GA with a combination of other two algorithms, namely BN and FS. The framework was designed as a GA module based on FS algorithms associated with parallel to a BN classifier opted. This GA–BN framework was developed so that, the GA makes an iterative search over the 6 skin diseases containing 34 variant features with an objective in choosing the most appropriate features that define each skin disease. This process in parallel develops performance of the BN module using a multi-cross-verification process. The subgroups of the remaining data could also identify the remaining skin issues other than ESD through a BN-responsible associated data. Thus, by deriving a higher performance of the model with 93.20% for identifying the people with ESD. The designed model was again tested with other models like SVM, MLP, SL, and FT for knowing its capacity. The GA design was also compared with other algorithms which were BF and SF [4]. The authors claim that their developed model has only showed 3 shortcomings out of 366 occasions.

The examination of erythema disease is quite complex for identifying as a skin disease due to its similar complications with the other skin diseases as said by the author. It can be seen by the redness on the inner skin layers making various skin complications, inflammations, etc. The developed model in this research is based on classifying the skin disease with the SVM algorithm using the linear kernel. The main motive of this research is to identify the ESD disease over the set of patients by drawing the outcomes to the impacting properties over the collected information. The developed methodology as stated by the author is a strong algorithm that could identify both the consecutive and discontinued types of data using the opted kernel. The proposed methodology was also said to be made a comparative approach to find out the possible outcomes over other kernels where the opted method has outperformed for the given problem [5]. By performing the cross-verification method, determining the optimal value for the parameters was done. The authors have opted J48 and C4.5 classifiers, where J48 derived higher accuracy of 92.1875%.

The motive of this research paper is to propose an automatic detection machine, or a diagnosis system based on the various symptoms to improve the best results when compared to manual diagnosis. From the author, identifying any disease at its earlier stage is more crucial and difficult so that the individuals could get their respective stage medication through a physician. This research work proposed an automatic ESD classification over the collected six various distant ESD-related symptoms as features and the person with the more impacting properties of ESD was supposed to be determined. For this purpose, the researchers performed an ML framework for differentiating purposes, namely SVM, for retrieving an optimal outcome. The innovative swarm algorithm and an EHO optimization techniques were opted to find the optimal parameters for the SVM. The proposed system is said to be performed along with the standard ML algorithms like ANN, ELM, and EHO–SVM to verify its preciseness where it gained 94.07% The authors stated that their proposed system underwent an overfitting problem, but it could gain robustness [6]. Table 1 illustrates the working phenomenon of previous models developed by various authors.

## 3   Proposed Methodology

In the given dataset, a few records contain missing values, so the proposed system first cleans the data. The second step of the proposed system is to identify the important features, the feature engineering step selects the below features represented in Table 2 as important attributes to predict the stage of cancer.

Out of 34 attributes, the proposed system has identified 13 attributes as important. Now, the proposed tries to identify the best parameters for the heterogeneous algorithms namely KNN and Decision tree using the hyperparameters. The tuning process of the algorithms is illustrated as follows.

### 3.1   Decision Trees

These trees can be utilized for both classification and regression process. The learning process of the decision tree depends on the rules formed from the if–then horn clause notation, which plays an vital role in decision-making process either by considering the entropy or by considering the Gini Index. In general, most of the systems try to split the tree by making the highest information gain attribute as the root node. But it is sometimes good to identify the low impurity nodes, which is known as "Entropy". The depth of the tree is considered as one of the tuning parameter, which is considered as number of attributes-1 but most of the training algorithms suggest that max depth will be between 2 and 8 or a multiple of 10. Another important parameter to be tuned is "Feature Importance", which computes the amount of entropy value to be decreased. The proposed system has identified four parameters to tune and got the best estimator values as shown in Table 3 by performing 5-folded cross-validation.

**Table 1** Comparison of existing models

| Authors | Algorithms used | Merits | Demerits/future work |
|---|---|---|---|
| Baraa Iyad | JustNN, backpropagation feed forward NN | Even though the developed model is a basic framework, it showed a greater performance and could be used for other skin diseases | Deeper algorithms in ANN could be used over larger data considering more features |
| Sayan | ANN, DNN, Derm2Vec, XG Boost, SVC, RF, NB, and KNN | The proposed methodology derived greater results than the standard algorithms | For higher dimensional data, the results may vary |
| Sivasubramanian | GBRNR, AdaBoost, Rf, RT, and REPtree | The experiment was conducted both with and without feature election | Semantic web mining can be implemented as an extension of this research work |
| Özçift | GA-FS, BN, MLP, SVM, FT, and SL | The combinatorial framework derived best for all the six different skin diseases | A deeper study with GA can be implemented |
| A. Basu | SVM linear | The method could classify both sequential and discontinuous data | Other ML and DL techniques could be performed as an extension When irregular data is fed, the developed model could not classify the data much |
| Tuba | ANN, ELM, EHO-SVM, swarm intelligent algorithm, and EHO | The proposed system was robust | Even the system was robust, there is an issue called, overfitting of data In future, FS was suggested |

## 3.2 Nearest Neighbors

The proposed algorithm has implemented grid search to tune the estimators, suppose the KNN has two estimators, namely weight and number of neighbors then all the possible combinations are cross-checked as shown in Fig. 2 (Table 4).

In KNN, there are five possible distances that can be acted as the estimators for the metric parameter, in general, researchers tends to use Euclidean distance, but the proposed system suggested manhattan distance, which covers both linear and non-linear distance calculation between two data points in the grid architecture by covering block by block transition. These types of distances are preferred by the high dimensional data sets. Using these two tuned algorithms, a stacking ensemble algorithm is proposed as shown below.

**Table 2** Important attributes of ESD

| S.no | Feature selected |
|------|------------------|
| 4 | Itching |
| 5 | Koebner_phenomenon |
| 13 | Pnl_infiltrate |
| 14 | Fibrosis_of_the_papillary_dermis |
| 19 | Clubbing_of_the_rete_ridges |
| 20 | Elongation_of_the_rete_ridges |
| 21 | Thinning_of_the_suprapapillary_epidermis |
| 26 | Vacuolisation_and_damage_of_basal_layer |
| 27 | Spongiosis |
| 28 | Saw-tooth_appearance_of_retes |
| 30 | Perifollicular_parakeratosis |
| 33 | Age |
| 34 | Class |

**Table 3** Best estimators of decision trees

| S.no | Estimator | Value |
|------|-----------|-------|
| 1 | Criterion | Entropy |
| 2 | Feature importance | Auto |
| 3 | Depth | 20 |
| 4 | Sample leaf | 5 |
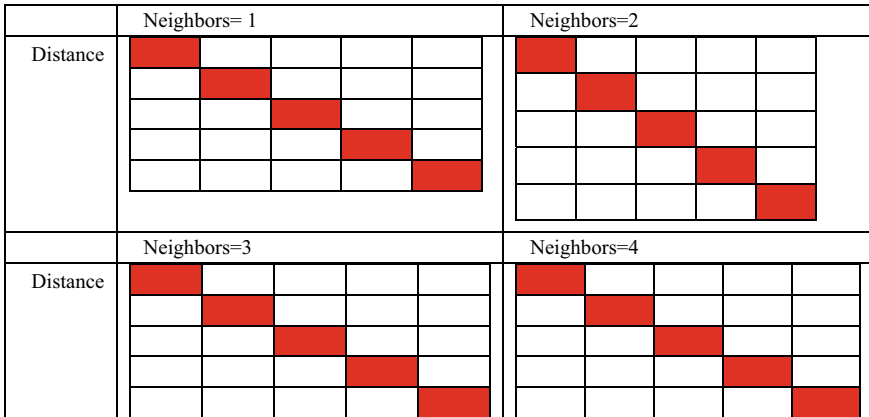


**Fig. 2** Validation across each neighboring Unit

**Table 4** Best estimators of KNN

| S.no | Estimator | Value |
|------|-----------|-------|
| 1 | Metric | **Manhattan** |
| 2 | Weights | **Distance** |
| 3 | Neighbors | **11** |

## *3.3 Pseudocode for Tuned Stacking Algorithm*

```
Input: Skin_Cancer_Data←Dataset with best features
Output: Print the performance metrics
Begin
1. Define base level as empty list
2. Add two tuned algorithms to the base level list
3. Define meta level estimator using logistic regression
4. Define stacking classifier using base and meta classifiers
5. Print the scores
End
```

## 4   Results and Discussion

See (Table 5 and Fig. 3).

Table 6 compares the existing models, KNN, Decision Trees, and proposed stacked ensemble models to check the efficiency of the proposed system.

**Table 5** Results of standalone and stacking algorithms

| S.no | Algorithm | Accuracy |
|------|-----------|----------|
| 1 | KNN | 88.7 |
| 2 | DT | 94.7 |
| 3 | Stacking ensemble | 95.8 |

**Fig. 3** Stacking of KNN and DT
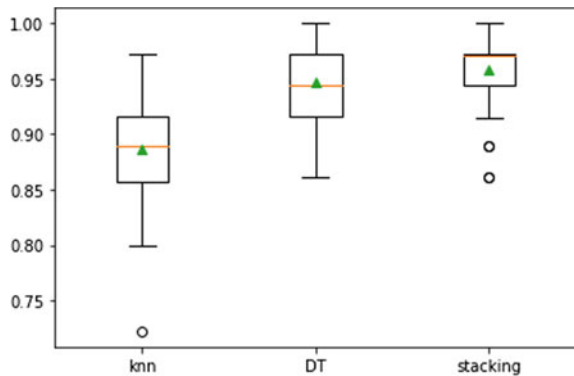
**Table 6** Accuracy analysis

| S.no | Algorithm | Accuracy |
|------|-----------|----------|
| 1 | Integrated tool | 95 |
| 2 | Derm2Vec | 91.2 |
| 3 | REPtree | 95.1 |
| 4 | ML + Genetic Algorithm | 93.20 |
| 5 | SVM | 92.18 |
| 6 | Swarm combined EHO | 94.07 |
| 7 | Decision trees | 88.7 |
| 8 | KNN | 94.7 |
| 9 | Stacked ensemble | 95.8 |

In Fig. 4, the stacked ensemble algorithm has achieved better results than the previous models. On X-axis, the system has represented studied models along with Decision Trees and KNN. It also included a stacked ensemble model. On Y-axis, obtained accuracy values are represented. The confusion matrix obtained for the proposed system is represented in Table 7.

The recall and precision are computed for each class and a weighted average is considered as the final output. The below section explains the computation of a single class as an example with the help of equations:

$$\text{Precision of ith class} = \frac{a[i, i]}{ith\ Column\ Count} \tag{1}$$

$$\text{Precision of 1st class} = \frac{27}{27} = 1.00$$
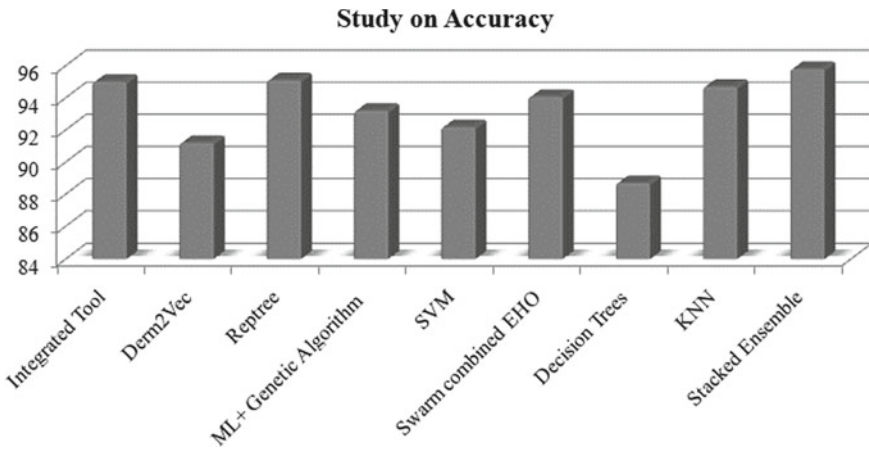


**Fig. 4** Analysis on accuracy results obtained

**Table 7** Accuracy of multi classification

|  | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | Class 6 | Total |
|---|---|---|---|---|---|---|---|
| Class 1 | 27 | 0 | 0 | 0 | 0 | 0 | 27 |
| Class 2 | 0 | 8 | 0 | 3 | 0 | 0 | 11 |
| Class 3 | 0 | 0 | 11 | 0 | 0 | 0 | 11 |
| Class 4 | 0 | 1 | 0 | 8 | 0 | 0 | 9 |
| Class 5 | 0 | 0 | 0 | 0 | 10 | 1 | 11 |
| Class 6 | 0 | 0 | 0 | 2 | 0 | 3 | 5 |
| Total | 27 | 9 | 11 | 13 | 10 | 4 | 74 |

$$\text{Precision of 2nd class} = \frac{8}{11} = 0.72$$

$$\text{Precision of 3rd class} = \frac{11}{11} = 1.00$$

$$\text{Precision of 4th class} = \frac{8}{9} = 0.88$$

$$\text{Precision of 5th class} = \frac{10}{11} = 0.90$$

$$\text{Precision of 6th class} = \frac{3}{5} = 0.60$$

$$\text{Weighted Precision} = \sum_{i=0}^{n} percentage\ of\ ith\ class\ row\ count\ *\ ith\ class\ precision \tag{2}$$

$$= \frac{27}{74} * 1 + \frac{11}{74} * 0.72 + \frac{11}{74} * 1 + \frac{9}{74} * 0.88 + \frac{11}{74} * 0.90 + \frac{5}{74} * 0.60$$
$$= 0.36 * 1 + 0.14 * 0.72 + 0.14 * 1 + 0.12 * 0.88 + 0.14 * 0.9 + 0.006 * 0.60$$
$$= 0.36 + 0.1 + 0.14 + 0.1 + 0.12 + 0.0036$$
$$= 0.82$$

$$\text{Recall of ith class} = \frac{a[i, i]}{ith\ Row\ Count} \tag{3}$$

$$\text{Recall of 1st class} = \frac{27}{27} = 1.00$$

$$\text{Recall of 2nd class} = \frac{8}{9} = 0.88$$

$$\text{Recall of 3rd class} = \frac{11}{11} = 1.00$$

$$\text{Recall of 4th class} = \frac{8}{13} = 0.61$$

$$\text{Recall of 5th class} = \frac{10}{10} = 1.00$$

$$\text{Recall of 6th class} = \frac{3}{4} = 0.75$$

$$\text{Weighted Recall} = \sum_{i=0}^{n} percentage\ of\ ith\ class\ column\ count\ *\ ith\ class\ recall \quad (4)$$

$$= \frac{27}{74} * 1 + \frac{9}{74} * 0.88 + \frac{11}{74} * 1 + \frac{13}{74} * 0.61 + \frac{10}{74} * 1 + \frac{4}{74} * 0.75$$

$$= 0.36 * 1 + 0.12 * 0.88 + 0.14 * 1 + 0.17 * 0.61 + 0.13 * 1 + 0.005 * 0.75$$

$$= 0.36 + 0.1 + 0.14 + 0.10 + 0.13 + 0.00375$$

$$= 0.833$$

## 5  Conclusion

In this paper, we present stacking, a unique ensemble learning approach for the diagnosis of Erythemato Squamous disease (ESD) that hasn't been published in the literature to our knowledge. We also discovered that there are haven't been many recent studies in the literature on the through use of ensemble algorithms in classification. As a result, we conclude that stacking, our suggested ensemble strategy, is an excellent tool for diagnosing ESD. The major motto of the stacking algorithms is to find the best value by combining the results of different algorithms. In this mechanism, the selection of algorithms is done purely by analyzing the data. This paper believes that the suggested hybrid machine learning model stacking can be used in various areas of medicine, such as heart disease detection, cancer prediction, diabetes prediction, and so on, with certain adjustments. In the future, we intend to develop neural networks for further improvement.

# References

1. Al-Kahlout BI, Naeem MM, Shepherd MJ (2021) ANN for the classification of eryhemato-squamous disease. Int J Acad Appl Res (IJAAR) 5(4):47–55. http://dspace.alazhar.edu.ps/xmlui/handle/123456789/2707
2. Putatunda S (2020) A hybrid deep learning approach for diagnosis of the erythemato-squamous disease. In: 2020 IEEE international conference on electronics, computing and communication technologies (CONECCT). (2020, July). https://doi.org/10.1109/conecct50063.2020.9198447
3. Sivasubramanian S, Jacob SG (2020) An automated ontology learning for benchmarking classifier models through gain-based relative-non-redundant feature selection: a case-study with erythemato-squamous disease. Int J Bus Intell Data Min 16(3):261. https://doi.org/10.1504/ijbidm.2020.106132
4. Özçift A, Gülten A (2018) Genetic algorithm wrapped Bayesian network feature selection applied to differential diagnosis of erythemato-squamous diseases. Digit Signal Process 23(1):230–237. https://doi.org/10.1016/j.dsp.2012.07.008
5. Basu A, Roy SS, Abraham A (2015) A novel diagnostic approach based on support vector machine with linear kernel for classifying the erythemato-squamous disease. In: 2015 international conference on computing communication control and automation, pp 343–347. https://doi.org/10.1109/ICCUBEA.2015.72
6. Tuba E, Ribic I, Capor-Hrosik R, Tuba M (2017) Support vector machine optimized by elephant herding algorithm for erythemato-squamous diseases detection. Proc Comput Sci 122:916–923. https://doi.org/10.1016/j.procs.2017.11.455

# Military Hand Signal Classification Using Deep Learning

**Nunna Mohit Sai Aravind** , **S. Hariharan** , **and Ayesha Shaik**

## 1   Introduction

Military hand signals are used as a means of non-verbal communication, usually to describe key commands to be used in an operation. These signals can be static or gesture-based [1]. Signals such as numbers, ok sign, and thumbs up and down are static signals as the command can be recognized by looking at the hand of the user alone. Other dynamic gestures may be used such as rotation of arms in upward and downward directions to signal different commands. A mixture of both may be used to communicate the severity/urgency of the message.

Militaries use different encodings and hand signals for short-distance communication. For our project, we have used a subset of the hand signals used by the USA army.

There are various situations in military operations where remote monitoring may be required. Hostage situations, rescue operations, disaster control, etc., use UAVs to monitor and pass commands to the field agents. UAVs mounted with cameras can detect signals from medium-range distances. Translation of these commands is usually done by a physical operator. However, this process of translation can be automated by various approaches, including hardware-assisted, software-based methods. Hardware-assisted methods involve the operator using specialized gloves with gesture detection sensors and transmitter systems for communication. Purely software solutions will involve scanning an image or video input and making predictions for the expected signals based on the same. This project aims to help in recognizing the hand signals used by the army using a trained deep learning model (convolutional neural network) as a software solution.

The project demonstrates the translation of a subset of static hand signals used by the NATO military manual [1].

N. Mohit Sai Aravind · S. Hariharan · A. Shaik (✉)
Vellore Institute of Technology, Chennai, India
e-mail: anoorcse@gmail.com

95

## 2   Literature Survey

There are a plethora of sign language translation and recognition models in the field. Works on SL recognition have used various languages like the American SL [2, 3], Korean SL [4], Chinese SL [5], and Japanese SL [6]. C. Wang, W. Gao, and J. Ma explain the hidden Markov model-based recognition system for Chinese SL which has a large-scale set of more than 5000 signs as its vocabulary [7]. Christian Vogler and Dimitris Metaxas develop scalable hand recognition systems which solve challenges such as handling simultaneous events, like signals which involve simultaneous change in hand movement and shape [3]. They train hidden Markov models to recognize the phonemes, instead of whole signs, and it can help in recognition of large-scale vocabularies. J. S. Kim and W. Jang et al. present a model which uses pair of data gloves as a sensing device for detecting motions of hands and fingers. They are able to recognize Korean SL (KSL) and provide a translation for the same in text format in Korean Language [4].

Convolutional neural networks (based on [8]) are deep learning algorithms that can learn features by assigning weights and biases to image-based inputs. These models are very successful at image recognition [9, 10]. For image manipulation, we have applied Gaussian blurring as it can make the image a bit more robust and it makes the model stabilized to noise. There are some projects that prepare data using Xbox Kinect Software to capture images. Kinect helps us in creating models with better accuracy for ASL as they are able to learn depth-based features [2].

In Pigou's paper [11], he recognize 20 Italian gestures using convolutional neural networks. Their model achieves an accuracy of 91.7%. For the task of gesture spotting in the ChaLearn competition, the model gives a Jaccard Index value of 0.789. In our model, we have used batch normalization as explained in [12] how batch normalization makes the optimization landscape smoother allowing for faster training.

The human hands and arm signals are recognized using a computer system according to the paper published by Lampton and his colleagues [13] which consists of two video cameras and a system to recognize gestures by analyzing the position and movement of the hands. The recognition rate, however at best is 87% for an individual and the minimum accuracy was 57%. They mainly make use of the Cybernet Gesture Recognition Software (GRS) for tracking the movements. They have used the sensors and mapped the hand and arm gestures with a virtual environment so that soldiers can train using these simulations, whereas, in our model, we design a CNN which can be used with drones for reconnaissance and similar purposes. Different ways of augmenting data such as using transformations, Generative Adversarial Networks (GANs), and other approaches are explained in this paper [14]. To increase the images in our dataset, we have mainly used transformations like rotation, reflection, and scaling.

# 3 Methodology

## 3.1 Data

We combine some of the signs from the Sign Language MNIST Dataset [2] in Kaggle with another custom dataset. From the Kaggle dataset, the signs in Fig. 1 are used.

After these signs are chosen, a custom dataset is built by capturing images of these signs using OpenCV–mouse bindings and is stored in their respective sign folders.

The captured images are converted from RGB to grayscale and blurred using Gaussian blur as shown in Fig. 2. Gaussian blur is extensively used in image processing applications, mainly to handle noise and reduce/standardize the details in the images. Mathematically, we apply the convolution of the image with the Gaussian function.

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-(x)^2/2\sigma^2} \tag{1}$$

Equation 1 represents a typical Gaussian function G(x) for normal distributions in one dimension. Parameter σ is the standard deviation of the distribution, and x
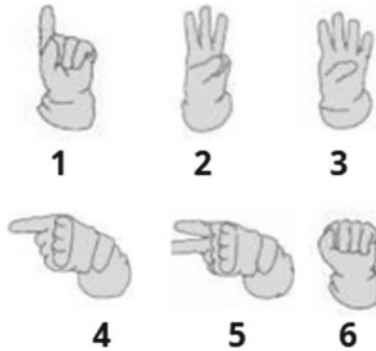


**Fig. 1** These are the signs used for this project and all of them are signals used by the army
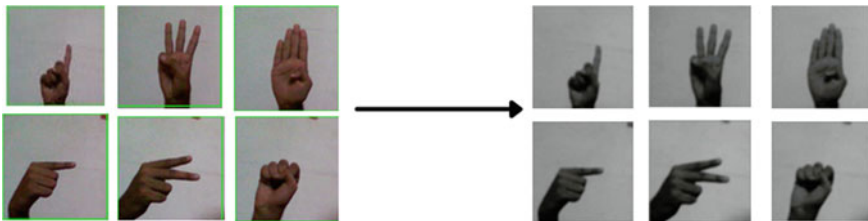


**Fig. 2** Normal image to Gaussian blurred grayscale image

represents the value of the variable, which is the distance of the point from origin along the horizontal axis.

Then, all these images are resized into the shape of 28 × 28 images so that model will be able to train faster on these images. A total of 6,558 images are generated using a script and combined with the Kaggle dataset to form 13,129 images and their respective labels are hot encoded.

## *3.2 Proposed Architecture*

We use three Conv2D layers for feature extraction, followed by Flatten, then followed by the dense layers, which help in classification, and can handle one-dimensional data. Each of the Conv2D layers has a kernel size of (3, 3) and uses the ReLu [10, 15] activation. These Conv2D layers are followed by max-pooling which uses max-pooling of the 2D dimensions with a kernel size of 2, 2 for better accuracy. Convolution 2D layers work best for images [11]. The output from these convolutional layers is flattened and passed to the dense layers which help in classifying the image.

The summary of the model is shown in Fig. 3. Convolution and max-pooling layers are stacked with batch normalization layers which apply a transformation such that the output and standard deviation are close to 0 and 1, respectively. It enhances model performance and reduces training time [12]. The dropout technique is used to prevent overfitting by reducing the capacity of the neural network. Dropout techniques achieve this by randomly ignoring certain neurons while training.

## *3.3 Generalization and Training*

We have split the data into training, cross-validation, and test set with a ratio of 80:10:10. Methods such as dropout [16] and data augmentation [14] have been applied during training to control overfitting. The data augmentation is performed in real time and consists of zooming up to 10%, shifting images horizontally and vertically by 10%, and rotations up to 20 degrees in both the directions. The model is compiled using an Adam optimizer with a learning rate of 0.001 and categorical cross-entropy is the loss function that has been used. Training is done for 50 epochs and the hyperparameters are tuned in such a way that it performs well on the development set. The entire training is done in GPU-based environment on Google Colab using the Python library TensorFlow.

```
Model: "sequential_1"
_____
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_3 (Conv2D)            (None, 26, 26, 64)        640
_____
max_pooling2d_3 (MaxPooling2 (None, 13, 13, 64)        0
_____
batch_normalization_4 (Batch (None, 13, 13, 64)        256
_____
conv2d_4 (Conv2D)            (None, 11, 11, 64)        36928
_____
max_pooling2d_4 (MaxPooling2 (None, 5, 5, 64)          0
_____
batch_normalization_5 (Batch (None, 5, 5, 64)          256
_____
conv2d_5 (Conv2D)            (None, 3, 3, 64)          36928
_____
max_pooling2d_5 (MaxPooling2 (None, 1, 1, 64)          0
_____
batch_normalization_6 (Batch (None, 1, 1, 64)          256
_____
flatten_1 (Flatten)          (None, 64)                0
_____
dense_2 (Dense)              (None, 128)               8320
_____
dropout_1 (Dropout)          (None, 128)               0
_____
batch_normalization_7 (Batch (None, 128)               512
_____
dense_3 (Dense)              (None, 6)                 774
=================================================================
Total params: 84,870
Trainable params: 84,230
Non-trainable params: 640
_____
```

**Fig. 3** Model summary

## 4 Results

The model on training with Keras achieves a training accuracy of 95.12% and a validation accuracy of 98.32% after training with 50 epochs. On evaluating the model with test dataset, it gives a test accuracy of 97.94%. The predictions for test images along with the confidence of prediction can be displayed as follows.

Some of the samples on the test set with the predicted labels (results of the model) and true labels are shown in Fig. 4. The dropout and image augmentation prevents the model from overfitting.
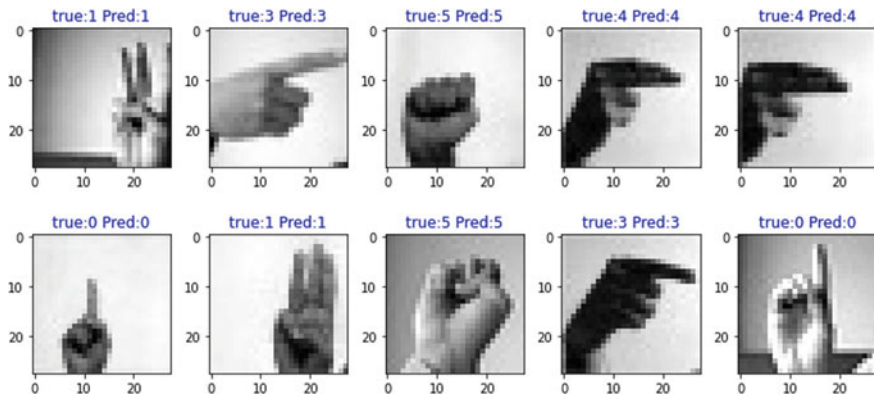
**Fig. 4** Test images with predicted and real labels

The graphs of model loss and model accuracy with each epoch are plotted in Fig. 5. From the graph, we can see that the model does not overfit the data during the training.

Metrics such as precision, recall, and F1-score for the task of classifying the hand signals are shown in Fig. 6.

Precision is the ratio between the actual positive results and the total predicted positive results for a label.

$$\text{Precision} = \frac{True\ Positives}{True\ Positive + False\ Positive} \tag{2}$$

Recall is the ratio between the actual positive results and the total actual positive results for a label.
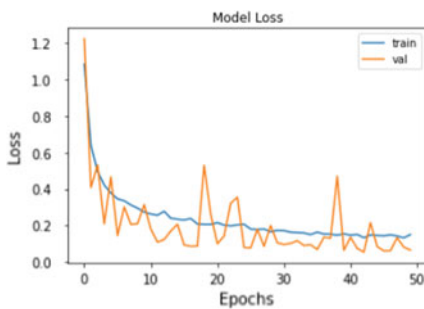


(a) Model loss vs epochs                    (b) Model accuracy vs Epochs
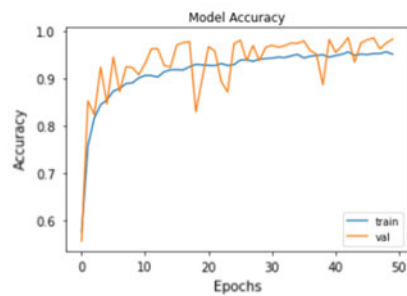
**Fig. 5** Plots of model loss and accuracy while training the model

|  | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 0.99 | 1.00 | 228 |
| 1 | 0.99 | 0.95 | 0.97 | 238 |
| 2 | 0.91 | 0.99 | 0.95 | 200 |
| 3 | 0.99 | 0.96 | 0.97 | 217 |
| 4 | 1.00 | 0.99 | 0.99 | 188 |
| 5 | 0.99 | 1.00 | 0.99 | 241 |
| micro avg | 0.98 | 0.98 | 0.98 | 1312 |
| macro avg | 0.98 | 0.98 | 0.98 | 1312 |
| weighted avg | 0.98 | 0.98 | 0.98 | 1312 |
| samples avg | 0.98 | 0.98 | 0.98 | 1312 |

**Fig. 6** Precision and recall table

$$Recall = \frac{True\ Positives}{True\ Positive + False\ Negative} \tag{3}$$

F1-score is the harmonic mean of precision and recall.

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

Support is the number of actual occurrences of the label.

The classification results of the model give a high precision and recall score ranging from 0.91 to 1.0.

## 5 Conclusion and Future Work

Hand signals are extensively used for non-verbal communication in militaries. This work shows how trained models can be used to recognize different signs of army signals not included in the training set. This project classifies a subset of the static hand signals used in the NATO army manual. Thus, we can predict static military hand signals with high accuracy using convolutional neural networks. Future work may involve extending this solution to include dynamic gesture-based predictions using hidden Markov models. Hand signal detection can be extended to other sign datasets like fire rescue and search operations. Background subtraction can be added to further optimize the model. Improved speed and durability to noise for complex gestures can be fine-tuned.

# References

1. Training Circular 3–21.60 Visual Signals (PDF) (5 May 2020)
2. Sign Language MNIST For Hand Recognition Tasks (2019)
3. Vogler C, Metaxas D (2004) Handshapes and movements: multiple-channel American sign language recognition. In Springer Lecture notes in Artificial Intelligence, vol 2915, pp 247–258 (Jan 2004)
4. Kim JS, Jang W, Bien Z (1996) A dynamic gesture recognition system for the Korean sign language KSL. IEEE Trans Syst Man Cybern 26(2):354–359
5. Gao W, Fang G, Zhao D, Chen Y (2004) Transition movement models for large vocabulary continuous sign language recognition (csl). In: Sixth IEEE international conference on automatic face and gesture recognition, pp 553–558
6. Sagawa H, Takeuchi M (2000) A method for recognizing a sequence of sign language words represented in a Japanese sign language sentence. In: Proceedings of the fourth IEEE international conference on automatic face and gesture recognition. Grenoble, France, pp 434–439. (Mar 2000)
7. Wang C, Gao W, Ma J (2002) A real-time large vocabulary recognition system for Chinese Sign Language. In: Wachsmuth I, Sowa T, (eds) Lecture Notes in Artificial Intelligence, vol 2298. Springer, pp. 86–95
8. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11)
9. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf 1–9
10. Schmidhuber J, Meier U, Ciresan D (2012) Multi-column deep neural networks for image classification. In: 2012 IEEE conference on computer vision and pattern recognition. IEEE Computer Society
11. Pigou L et al (2014) Sign language recognition using convolutional neural networks. In: European conference on computer vision. Springer, Cham
12. Santurkar S et al (2018) How does batch normalization help optimization? In: Proceedings of the 32nd international conference on neural information processing systems
13. Lampton DR et al (2002) Gesture recognition system for hand and arm signals. Army Research Inst for the Behavioral and Social Sciences Alexandria VA
14. Mikołajczyk A, Grochowski M (2018) Data augmentation for improving deep learning in image classification problem. In: 2018 international interdisciplinary PhD workshop (IIPhDW). IEEE
15. Nair V, Hinton GE (2010) Rectified linear units improve restricted Boltzmann ma- chines. In: Proceedings of the 27th international conference on machine learning (ICML 2010), pp 807–814
16. Srivastava N et al (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958

# A Novel Blockchain Framework for Enhancing Online Transactions

**Sk. Khaja Shareef** [ORCID]**, R. Sridevi, V. Rama Raju, and K. S. Sadasiva Rao**

## 1 Introduction

In digital applications, one of the secure trustworthy decentralized systems is defined as blockchain [1]. Moreover, it has included governments, private companies, and educational sectors [2], which have common or individual goals [3]. Once, the transaction is verified then the users have made the blocks that have included a pack of a transaction and then upgraded to the final block [4]. Moreover, the operated blockchain is linked with the hash value block to hide the data from unauthenticated users [5]. Recently, several blockchain communities' networks have wanted to organize machine learning (ML) [6] techniques to attain the data analysis or computation statistics results [7]. In addition, medical professionals need to offer specific treatments for specific patients, who are designing the predictive representation for disease classification [8].

In that scenario, securing the medical report is very crucial because of the attack's vulnerability [9]. However, it is flexible to broadcast the data to any user [10]. The technology, blockchain is often utilized to distribute currencies online, so it has been

Sk. K. Shareef (✉)
Computer Science and Engineering, JNTUH, Hyderabad, India
e-mail: khaja.sk08@gmail.com

R. Sridevi
Computer Science and Engineering, JNTUH College of Engineering Hyderabad, Hyderabad, India
e-mail: sridevirangu@jntuh.ac.in

V. R. Raju
Computer Science and Engineering, CMR College of Engineering & Technology Hyderabad, Hyderabad, India

K. S. S. Rao
Department of MCA, Chaitanya Bharathi Institute of Technology, Hyderabad, India
e-mail: sadasiva_mca@cbit.ac.in

utilized in many online banking applications [11]. Also, the paradigm blockchain is effectively applicable in both circumstances that are the Internet of things (IoT) and the cloud [12]. On the other hand, the blockchain strategy is proficiently used in wireless communication appliances to secure the user's details from third parties [13]. Here, the signal is encrypted by the blockchain model before the signal is transmitted [14].

The main reason for developing the security models in the blockchain is the computation of multi-party entities in the distributed environment is insecure in some cases because of system failure [15]. Moreover, this kind of problem is happened because of unauthenticated persons [16]. In addition, several ML and DL models already exist to secure the files in the blockchain environment [17]. But it couldn't succeed because of complex data and attacks harmfulness. The structure of this article is summarized as follows: the recent works related to this research are detailed in Sect. 2. The system model and problem statement are detailed in Sect. 3. In Sect. 4, the proposed method is described. The result and discussion along with the case study and comparison are explained in Sect. 5. At last, the conclusion of this research is demonstrated in Sect. 6.

## 2    Related Works

To enhance the blockchain security model, Kim et al. [20] have designed the Ml model based on the stochastic gradient approach. In the experiment, it has verified that the designed replica has sufficient capacity to fight against adversarial attacks. Finally, the presented model has gained the finest accuracy than the previous models. But it takes more time to complete the process because of the large transaction system.

In recent, the intrusion diagnosis system in blockchain became the key factor in the blockchain appliance to enhance the privacy measure. So, Alkadi et al. [21] have projected an ensemble model to recognize harmful malicious events to ensure info privacy. Also, this framework can function as decision support that can support the users to migrate the files securely. However, it is complex in design.

To prevent data forgery in the vehicular cloud environment, Lee et al. [22] and Balaram et al. [33] presented smart architecture with the crypto model to rectify the attack vulnerability issues in a decentralized network. Hence, this scheme can help the users to share the information without any difficulties. However, this smart architecture is not easy to redesign again, once the flaw occurs.

Serrano [23] has utilized the blockchain strategy in 5G wireless communication to secure communication among the user ends. Finally, the strength of the designed blockchain in the 5G wireless technology is proved by evaluating data broadcasting rate and throughput ratio. Hence, it has gained a wide range of data broadcasting rates than other models. But compared to other conventional models, it has consumed more time.

To create an efficient intrusion detection paradigm, Moustafa et al. [24] developed a detailed review based on an intrusion prediction framework in a blockchain environment.

## 3 Existing Model and Problem Statement

In the modern era, every transaction is linked with the help of cryptographic blockchain methodology [9, 25]. Here, the blocks are initiated in sequential arrangement for permitting their use to track the digital online transaction [29, 30]. Moreover, each block consists of a set of private keys, public keys, and cryptographic keys [25]. The combination of above-mentioned three keys is termed a digital signature [31]. Consequently, the online data transaction with correspondent hash functions, signatures, and keys. The paradigm of cloud computing became the large source of data to broadcast the information to the users in online applications [17, 32]. At the same time, the rise of assaults is tremendously maximized in the cloud mechanism, so the model known as blockchain has been designed diversely and effectively [18, 30].

However, the attacks can influence the blockchain and capture the half of info from the original transaction [19]. Thus, it has affected the integrity measure of the blockchain. The existing models are somewhat good but fail to offer continuous security because of the large data size. Hence, all these issues are motivated this research towards securing the blockchain by preventing assaults.

## 4 Proposed Methodology

A secure homomorphic Serpent Blockchain (HSB) algorithm is proposed to preserve privacy and enhance security activities. Initially, the amounts of data bytes are trained to the system and stored in the particular database. At first, find the hash value using a homomorphic encryption algorithm in trained data and that value is considered as hash one. Consequently, the collected data bytes are encrypted using Serpent cryptographic algorithm, and that encrypted data is taken as hash two. Finally, hash one and hash two are verified successfully, which is demonstrated in Fig. 1.

### 4.1 Process of HSB Algorithm

The proposed HSB algorithm is a combination of homomorphic and Serpent cryptography algorithms. Moreover, this algorithm is used for securing data through online transaction manner. Based on blockchain technology, online transaction process will happen.

**Hash-1 calculation using Homomorphic**. The homomorphic concept includes several functions like primary transformation, round function, etc. Here, the homomorphic concept is developed to deliver the data effectively in the online application. Moreover, the homomorphic encryption algorithm is a powerful strategy when compared to other encryption techniques, which are utilized to achieve higher security during the transaction process. The proposed strategy is performed based on
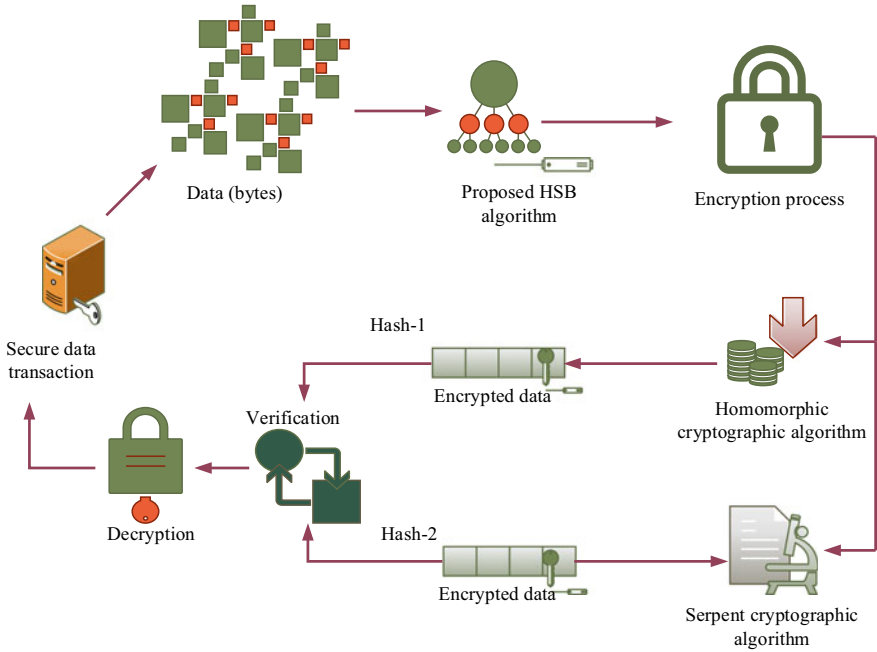
**Fig. 1** Proposed framework

a homomorphic cryptography hash algorithm. Moreover, the calculation of hash function ($H_i$) is mentioned in Eqs. (1) and (2)

$$H_i = H^*\left(v_i^*\right) \tag{1}$$

$$H_i = \sum_{i=1}^{k} v_i^* w_i^* \tag{2}$$

where $H^*$ is represented as a raw hash value, $v_i^*$ is denoted as the raw data, and $w_i^*$ is the weightage function of each raw data. Moreover, the homomorphic algorithms for two data that is $O_1^*$, $O_2^*$ and that weight functions $w_1^*$, $w_2^*$ hold that all hash function and verify whether data is hacked or not using Eq. (3)

$$H_i\left(O^*\right) = H_i\left(O_1^*\right)^{W_1^*} H_i\left(O_2^*\right)^{W_2^*}. \tag{3}$$

The proposed algorithm has provided the finest security to transfer large data in the online application.

**Data encryption**. In the encryption process, 64-bit plaintext is taken and 64-bit ciphertext is generated through the corresponding plaintext. Consequently, the input

plain text is configured by the primary transformation, which is provided transformed output. In these transformed functions, the correspondent key is used in decryption and encryption is similar and it's assumed as a symmetric key. Here, the 64-bit key length is generated from 128 user-defined key lengths. Moreover, the serpent encryption algorithm is used as an encryption process. Consequently, the generation of the key is using Eq. (4)

$$C_i = E^*(k)\left[p_i^* \oplus C_{i-1}\right] * FT \,|\, P_u^*(k) \tag{4}$$

where $C_i$ is denoted as ciphertext, $E^*(k)$ is represented as the encrypted key of the correspondent plain text $P_i^*$, and FT is denoted final transformation. Additionally, subkeys are also generated linear information for accessing the online transaction details. Then, 64 bits and pre-keys are identified using Eq. (5)

$$p_i^* = P_{i-8}^* \oplus \left(P_{i-7}^* <<< 1\right) \oplus \left(P_{i-6}^* <<< 2\right)........\left(P_{i-1}^* <<< 4\right) \tag{5}$$

where $i$ is denoted as 0–61, $P_{i-8}^*........P_{i-1}^*$ is the 128 user-defined keys which is split into four forms of the 32-bit plain text to develop 128. Shift rotation is denoted as <<<. Moreover, this encrypted data is taken as the second hash function. The transformation function is expressed in the below Eq. (6):

$$T_f\left(P_i^*\right) = \left(b_{1,0}, ........b_{1,k-1}, ........b_{1,0}, ........b_{l,k-1}\right) \Rightarrow C_i^* \tag{6}$$

where $b_{m,n}$ is the nth bit in the binary representation of $b_m$ from lower level to higher level bit. Consequently, this process is iterating still each subkey allotted to the particular plain text. Here, hash 2 is estimated using a serpent encryption algorithm and attained hash is stored in the cloud database. Therefore, the proposed algorithm is developed in terms of hash algorithm set it will split the data as well as provide high security during the transaction process.

The destination node decrypts the data using the private key. In addition, the online transaction system generates the public key based on the below Eq. (7)

$$P_u^*(k) = \left|P_1^* P_2^*\right|; \qquad P_1^*,\ P_2^* \Rightarrow P_i^* \tag{7}$$

where $P_u^*(k)$ is represented as the public key, $P_i^*$ is denoted as plain text. After that, find create the secret key of all plain text using the below Eq. (8):

$$S_e^*(k) = P_u^*(k)\left(P_i^*\right) \tag{8}$$

After the key generation process, the data is encrypted using a correspondent encryption algorithm. Here, the encrypted data consist of two ciphertexts, which are calculated mathematically using Eqs. (9) and (10),

$$C_1 = (Q * H) + S_e^*(k) \tag{9}$$

$$C_2 = \left(O^* + \left[Q * P_u^*(k)\right]\right) + S_e^*(k) \tag{10}$$

where $Q$ is denoted as randomly choosing the number in a particular range [1, 0], $O^*$ is represented as original data and $H$ is denoted hash function. In addition, the encrypted data is changing the IP address during the transaction process.

*Algorithm. HSB model*
*Input: plain text (data bytes)*

Start        $\Rightarrow$ *Hash-1 calculation*

$H_i \Rightarrow v_i^\bullet(O_i^\bullet)$

    *for (i=0;i>n;i++)*
*End for*

$H_i \Rightarrow H^*(w_i^\bullet)$                *// Hash function-1*

*Data encryption*                *// compute second hash function// serpent encryption algorithm*

*64-bit plaintext* ($P_i^\bullet$)

*XOR operation with 4*
*rounds*                $P_1^*, P_2^* \Rightarrow P_i^*$

    *for* $(k = 0; k \geq 3; k++)$
*Transformation function*

$T_f(P_i^*) \Rightarrow b_{m,n}$                *Binary representation*

    *Encrypted 64-bit cipher*                *// Hash function-2*
    *text*
*Verification*

$H_i(O^*) \Rightarrow$ *(hash1=hash2)*                *// If it is not matched the data is hacked*
                                          *// if it is matched the data is secured*

*Decryption process*
*Inverse operation*                *//encrypted cipher text*

Stop        *Output: Finest solution*

While decrypting the data, the proposed algorithm should be satisfied. From the satisfaction, if hash 1 is the same as hash 2, then the proposed condition is verified the data is cannot hack the external user or third parties. If hash 1 is not matched with hash 2, the data is hacked by the external user or third parties during the transaction process. Furthermore, the developed HSB replica workflow is demonstrated in the Algorithm 1.

# 5 Result and Discussion

In this section, the efficiency of the developed HSB algorithm in the online transaction is executed and designed with suitable parameters. Performance estimation.

Performance of the developed model is compared with other existing techniques like Modification of Elliptic curve cryptography (MECC) strategy [26], Cloud-based RSA scheme [27], and cryptography hash (CH) algorithm [28] in terms of encryption time, decryption time, accuracy, resource usage, and confidential range.

**Accuracy**. The achieved accuracy rate of the proposed HSB algorithm is compared with other existing replicas such as CH, MECC, and cloud RSA. The proposed HSB replica has achieved 99% accuracy for 16 bytes of data size, and it has a higher accuracy measure compared with conventional replicas in Fig. 2.

**Encryption time**. Comparison of encryption time with other existing techniques is illustrated. Here, the CH technique has achieved 21 ms encryption time for using 16 bytes of data and the MECC technique has gained 196.6 ms. Furthermore, the Cloud RSA method has attained 95.56 ms encryption time for using 16 bytes of data.

Also, the developed HSB mechanism has obtained a 38 ms encryption time for using 16 bytes of data. While increasing the data size simultaneously, the encryption time will increase, and this is elaborated in Fig. 3.

**Decryption time**. The proposed HSB algorithm has achieved 34.92 ms decryption time, while compared to other methods, the proposed algorithm has taken less time. This comparison shows that the proposed strategy has attained the finest result and is described in Fig. 4.

**Resource usage**. The developed algorithm results are compared with other prevailing techniques like CH, MECC, and cloud RSA, which is shown in Fig. 5.

The proposed HSB algorithm has utilized 96.23% of resources for 16 bytes of data.
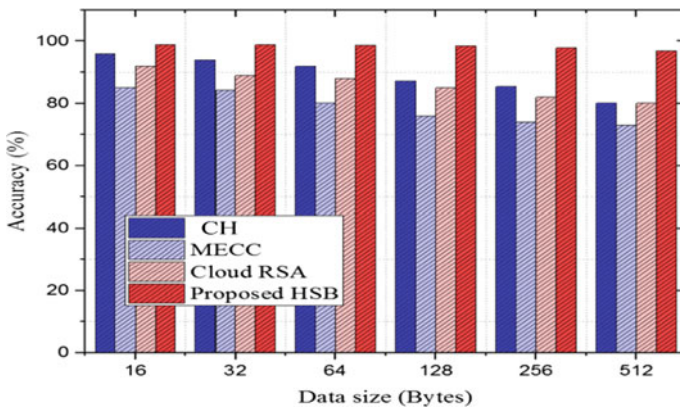


**Fig. 2** Performance comparison accuracy measure
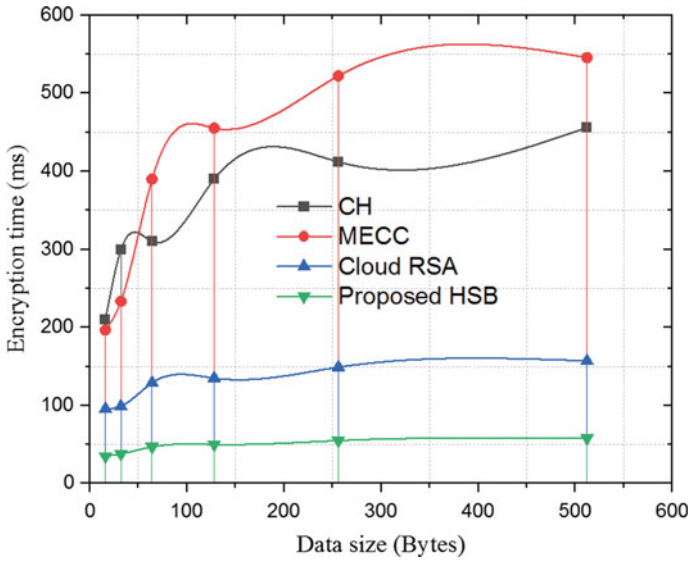
**Fig. 3** Validation encryption time
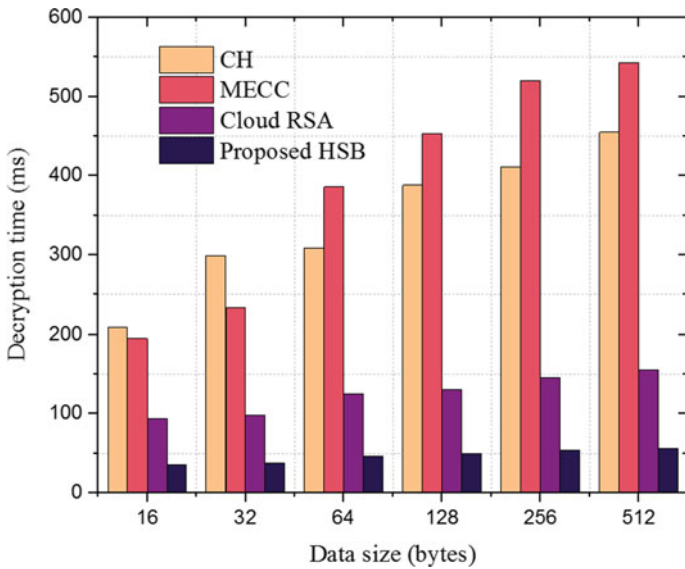


**Fig. 4** Comparison of decryption time
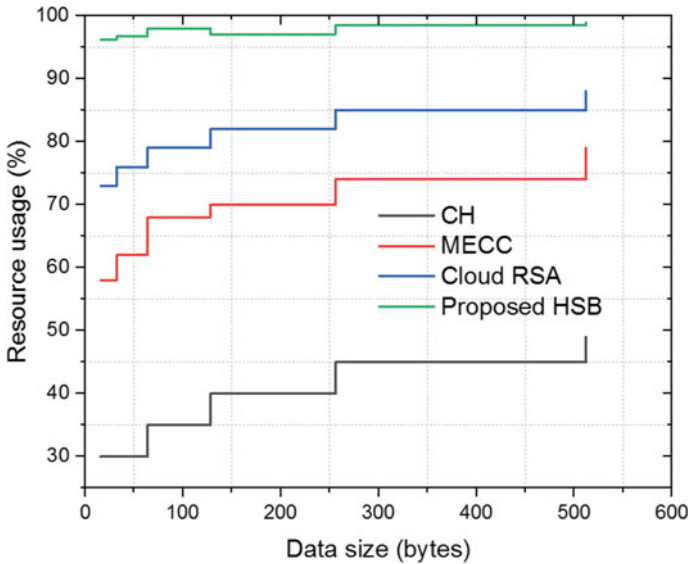
**Fig. 5** Performance of resource usage

**Confidential rate**. The comparison shows that the CH, MECC, and cloud RSA method has obtained a very low confidential rate over other methods. However, the proposed HSB algorithm in the online transaction system attained a very high confidential rate of 98% for 16 bytes of data. While compared with the other existing methods, the confidential rate is considerably increased, which is demonstrated in Fig. 6.

## 5.1 Discussion

The Overall discussion of this research shows a large data is securely delivered from source to destination in an online application. Moreover, blockchain techniques provide high security and high data protection during the transmission process. Here, the proposed algorithm is utilized to transfer the large data validly. From the overall comparison, the proposed HSB algorithm demonstrated the finest performance by achieving less encryption time and higher accuracy.

## 6 Conclusion

Nowadays, blockchain-based communication technology is significantly increasing to share information from one place to another place. Digital cloud-based platforms are facing so many harmful vulnerable activities. Therefore, enhancing security is an
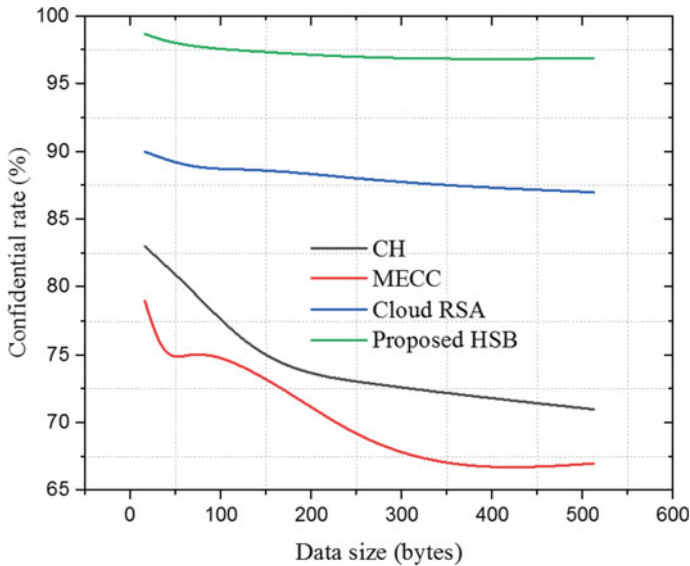
**Fig. 6** Performance of a confidential range

essential paradigm to protect the data from external users. Hence, in this research, a novel HSB algorithm is developed to encrypt and decrypt the large data size. Here, the HSB algorithm performs both encryption and decryption processes while the data transaction time. Moreover, the security is evaluated based on the equivalent of the hash function. In addition, the execution outcomes are evaluated in terms of confidential rate, encryption time, decryption time, resource usage, and accuracy. Significantly, the developed HSB algorithm has utilized 35 ms to encrypt 16 bytes of data. Also, the encryption and decryption time is reduced to 60 ms and 62 ms compared with existing security methods. Therefore, the proposed method is very secured for data sharing, especially in online application.

# References

1. Monrat AA, Schelén O, Andersson K (2019) A survey of blockchain from the perspectives of applications, challenges, and opportunities. IEEE Access 7:117134–117151
2. Rajeswari TSR, Shareef SK, Khan S, Venkatesh N, Ali A, Monika Devi VS (2021) Generating and validating certificates using blockchain. In: 2021 6th international conference on communication and electronics systems (ICCES), pp 1048–1052. https://doi.org/10.1109/ICC ES51350.2021.9489105
3. Syed TA et al (2019) A comparative analysis of blockchain architecture and its applications: problems and recommendations. IEEE Access 7:176838–176869
4. Yang J et al (2020) Blockchain-based sharing and tamper-proof framework of big data networking. IEEE Netw 34(4):62–67

5. Roy DG et al (2019) QoS-aware secure transaction framework for internet of things using blockchain mechanism. J Netw Comput Appl 144:59–78
6. Liu Y et al (2019) LightChain: a lightweight blockchain system for industrial internet of things. IEEE Trans Ind Inf 15(6):3571–3581
7. Florian M et al (2019) Erasing data from blockchain nodes. In: 2019 IEEE European symposium on security and privacy workshops (EuroS&PW). IEEE
8. Xu C et al (2018) Making big data open in edges: a resource-efficient blockchain-based approach. IEEE Trans Parallel Distrib Syst 30(4):870–882
9. Iqbal N et al (2021) A novel blockchain-based integrity and reliable veterinary clinic information management system using predictive analytics for provisioning of quality health services. IEEE Access 9:8069–8098
10. Singh S et al (2020) Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city. Sustain Cities Soc 63:102364
11. Bhardwaj A et al (2020) Penetration testing framework for smart contract blockchain. Peer-to-Peer Netw Appl 1–16
12. Krishnan SSR, et al (2020) A blockchain-based credibility scoring framework for electronic medical records. In: 2020 IEEE globecom workshops (GC Wkshps). IEEE
13. Wang X et al (2019) Blockchain-based smart contract for energy demand management. Energy Procedia 158:2719–2724
14. Pee SJ et al (2019) Blockchain based smart energy trading platform using smart contract. In: 2019 international conference on artificial intelligence in information and communication (ICAIIC). IEEE
15. Wang H et al (2020) Blockchain-based fair payment smart contract for public cloud storage auditing. Inf Sci 519:348–362
16. Ye C et al (2018) Analysis of security in blockchain: case study in 51%-attack detecting. In: 2018 5th international conference on dependable systems and their applications (DSA). IEEE
17. Dey S (2018) Securing majority-attack in blockchain using machine learning and algorithmic game theory: a proof of work. 2018 10th computer science and electronic engineering (CEEC). IEEE
18. Wei P, Yuan Q, Zheng Y (2018) Security of the blockchain against long delay attack. In: International conference on the theory and application of cryptology and information security. Springer, Cham
19. Gupta R et al (2020) Blockchain-based security attack resilience schemes for autonomous vehicles in industry 4.0: a systematic review. Comput Electr Eng 86:106717
20. Kim H et al (2019) Efficient privacy-preserving machine learning for the blockchain network. IEEE Access 7:136481–136495
21. Alkadi O et al (2020) A deep blockchain framework-enabled collaborative intrusion detection for protecting IoT and cloud networks. IEEE Internet Things J 8(12):9463–9472
22. Lee Y et al (2020) A blockchain-based smart home gateway architecture for preventing data forgery. Hum-Cent Comput Inf Sci 10(1):1–14
23. Serrano W (2021) The blockchain random neural network for cyber secure IoT and 5G infrastructure in smart cities. J Netw Comput Appl 175:102909
24. Alkadi O, Moustafa N, Turnbull B (2020) A review of intrusion detection and blockchain applications in the cloud: approaches, challenges and solutions. IEEE Access 8:104893–104917
25. Shuaib M, Alam S, Daud SM (2020) Improving the authenticity of real estate land transaction data using blockchain-based security scheme. In: International conference on advances in cyber security. Springer, Singapore
26. Rani DR, Geethakumari G (2020) Secure data transmission and detection of anti-forensic attacks in cloud environment using MECC and DLMNN. Comput Commun 150:799–810
27. El Makkaoui K, Beni-Hssane A, Ezzati A (2019) Speedy Cloud-RSA homomorphic scheme for preserving data confidentiality in cloud computing. J Ambient Intell Hum Comput 10(12):4629–4640
28. Velmurugadass P et al (2021) Enhancing Blockchain security in cloud computing with IoT environment using ECIES and cryptography hash algorithm. Mater Today Proc 37:2653–2659

29. Sodhro AH et al (2020) Towards blockchain-enabled security technique for industrial internet of things based decentralized applications. J Grid Comput 18(4):615–628
30. Singh P et al (2021) Cross-domain secure data sharing using blockchain for industrial IoT. J Parallel Distrib Comput
31. Dunphy P, Petitcolas FAP (2018) A first look at identity management schemes on the blockchain. IEEE Secur Priv 16(4):20–29
32. Sodhro AH et al (2020) Towards blockchain-enabled security technique for the industrial internet of things based decentralized applications. J Grid Comput 18(4):615–628
33. Balaram A, Pushpa S (2018) Sybil attack resistant location privacy in VANET. Int J Inf Commun Technol 13(4):386–405. Inder Science

# Towards Improving the Packet Delivery Ratio in Mobile Ad Hoc Networks Using DSDV Protocol and Cuckoo Search Algorithm

**T. Devi** [ORCID], **R. Ganesan** [ORCID], **N. Deepa** [ORCID], **and K. Jaisharma** [ORCID]

## 1  Introduction

Mobile nodes form a collection that is wireless in nature and are involved in the creation of a temporary network called MANET (Mobile ad hoc network) [1]. The absence of centralized administration is one of the important benefits of MANET. Identification of paths is the major issue in MANET as the infrastructure is not centralized, topology seems to be dynamic, energy constraint, node heterogeneity, and bandwidth which is limited. Recognizing the path from source to destination for forwarding packets is termed routing. Due to the changing topology and bandwidth limitations, routing is a challenging issue in MANET [2]. It also depends on the consumption of energy of the nodes. The protocols can be used for the identification of shorter distances from source to destination and, at the same time, they should be adaptive.

### 1.1  *Problems of MANET*

Suppose a network consists of two nodes, the communication among them does not require any routing protocols or decisions for routing need not be done as they are close to each other. When the count of nodes involved in communication is more, routing protocols can be utilized to make decisions at the time of choosing the

T. Devi (✉) · N. Deepa · K. Jaisharma
Department of Computer Science and Engineering, Saveetha School of Engineering,
SaveethaInstitute of Medical and Technical Sciences, Chennai, India
e-mail: devi.janu@gmail.com

R. Ganesan
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
e-mail: ganesan.r@vit.ac.in

route between source and destination. This approach is important as battery power is utilized for operation of the mobile nodes [3]. Data transfer along with a delay that is minimum is necessary to reduce the wastage of power. Compression from the side of the protocol is also noted as an important factor as the bandwidth is wasted. There is a need for cryptographic algorithms to help in protecting data from the attackers. In order to prevent packet drop, QoS (Quality of Service) is also an important factor that is necessary in a MANET.

## 1.2   Factors in MANET

When the packets are sent to several nodes at a particular time it is called multicasting. Considering all nodes in a network broadcasting is performed for all nodes. One of the important factors is multicasting as the time taken for data transfer is very less in the case of multiple nodes involved. While the packet is traveling to nodes in a path, it does not go to the same node again and again before reaching the destination. The routing protocols should be designed so that the path is free from loops which results in avoiding bandwidth wastage as well as consumption of CPU [4, 5]. While breakage of routes occurs when disasters happen, transfer of data should happen without any interruption with the help of other available routes. The protocol designed should help in the creation of multiple routes at the time when it is necessary.

Dependency on the node which is centralized should not be allowed as the operation of protocol should be distributed. Route discovery between source and destination can be done only when there is a need for data transfer [6]. The protocols are reactive or proactive depending on their nature of working as the route discovery occurs in the latter case without considering the necessity. Links which are unidirectional in nature can be formed along with the bi-directional links also. These links can help in improving the performance of the protocol. Devices such as laptops and PDAs have battery power that is limited and they utilize the mode of stand-by for saving the power [7]. The protocols considered for routing should also consider the sleep mode.

Suppose data transfer needs to take place between two nodes chosen as node A and node C and this latter node is not in the range of node A. Services of the other B node can be used by this A for data transfer as there is an overlap in the range. The problem of routing in ad hoc networks needs special attention and the reasons behind this are as follows: (i) the propagation is not uniform while transmitting in wireless medium and (ii) hosts associated with them may cause movement at any instant of time. The problem of routing in MANET needs to be addressed as this may occur due to changes in the topology which causes the movement of nodes. It may also be caused by interference of radio signals as well as partitions in the network. Maintenance of information related to routing for every node is taken care of by the proactive protocols and the discovery of new routes at the time of need is done in reactive protocols and, finally, information related to the geographical location is
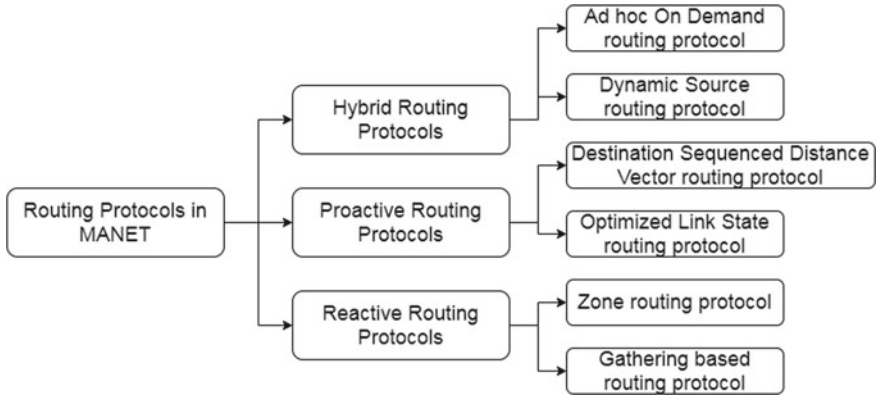
**Fig. 1** Routing protocols in MANET

also used for routing information [8]. The main classification of routing protocols is shown in Fig. 1.

1. Proactive protocol
2. Reactive Protocol
3. Hybrid Protocol.

In the case of proactive protocols, for the maintenance of links among the nodes, broadcasting of control packets takes place in the available network [9]. Table construction takes place in every node. For a specific destination chosen, the next hop is calculated in the table and the availability of information at the phase of transfer of data is a major benefit of this method. At the same time, when there is a change in the topology, tracking all paths becomes difficult. The main reason is some nodes may be present or they may also be mobile. DSDV termed Destination Sequenced Distance Vector as well as OLSR termed Optimized Link State Routing are the routing protocols of this type.

In the case of on-demand routing protocols, formation of path is done only when necessary. Route discovery is launched when there is a need for a path to reach the destination. Although the method is effective, it suffers from limitations such as the unavailability of routing information when it is necessary. AODV termed Ad Hoc On-Demand Distance Vector as well as DSR called Dynamic Source Routing are examples of this type of protocol.

Combining both proactive and reactive protocols, a hybrid approach is obtained. Nearby neighborhoods are examined using a proactive approach and as the paths are available. Identification of routes in the case of nodes that are not neighbors is performed by reactive protocols. ZRP termed Zone routing protocol is an example of hybrid approach.

Polynomial-time algorithms cannot be defined for optimization problems as exact solutions are not available. Biologically inspired algorithms can be employed for

solving optimization problems. Some of the significant algorithms are genetic algorithms, PSO termed Particle Swarm Optimization as well as Cuckoo Search. The principles behind the evolution occurring naturally are being used in solving optimization problems and have been used extensively in present research works. The reason behind this is due to less range in transmission for several networks. All nodes need to participate in performing this operation in a decentralized manner. The proposed system utilizes a novel routing protocol using cuckoo search algorithm.

The objective of the proposed work is as follows:

(a) Enhancement of DSDV protocol using the packets while requesting the routes in an ad hoc network to improve the PDR values.
(b) Cuckoo search algorithm to avoid route congestion and to find the best path for communication and data transfer at a higher speed.

The paper organization is as follows. Section 2 presents the literature works, Sect. 3 explains the proposed system, Sect. 4 presents the results obtained from implementation, and Sect. 5 discusses the conclusion of the proposed work.

## 2   Related Work

### 2.1   DSDV

Among the table-driven routing protocols, DSDV is an important protocol used in MANET [10]. In the case of protocols of distance vector, source, destination, and the distance among the nodes are also maintained. The packet travels from one node to another using the next hop in the packet information thereby finding the shortest path. The next hop is chosen in this way for the successive path and this makes the packets arrive at the destination by choosing the path that appears to be the shortest [11]. Estimation of distance needs to be known as the monitoring of the links that are outgoing is performed by every node. Broadcasting of the distance that seems to be shorter is also done periodically to all the neighbors of the nodes [12]. Periodical broadcasting of this information regarding the distance consists of entry for including the distance information including the advertising node also.

Sequence numbers are utilized by the DSDV protocol whose update is done with the help of destination. This avoids the problem of looping which is due to the information of routing available in a stale state. Every node in this protocol has a separate table for storing the information related to routing whose update is done periodically and the process of advertisement takes place to their neighbor nodes. In the case of a routing table, the sequence number which is known lastly is stored as an entry. Periodical transmission of updates is done by every node while there is the availability of information which is found to be new. Broadcasting of data consists of the sequence number generated newly with several information regarding the route such as the address of the destination node, hop count necessary for reaching

the destination as well as the received information with its corresponding sequence number.

Since DSDV is a proactive protocol, there exists a route from every node to destination and the exchange of hello packets takes place for the purpose of advertising. Upon receiving these packets, the nodes who are neighbors to these nodes add the address of sender in the table kept for storing the routing information [13]. This table is then shared to its neighbor thereby making every node identify its neighbors and establishing the paths among them.

Synchronization in the case of time is not performed and also the association regarding the updated information is also not done in this case. Traditionally, nodes that are in the accessible range of other nodes and their related information are updated in the packets along with the hop count necessary in reaching these nodes. In order to forward the decisions made, the sequence numbers that are utilized recently are noted and that particular path with this feature is preferred. Among these paths, the path with less hop counts is utilized. The addresses of the network layer are used by layer 3 for computation of the address of the destination as well as to find the next hop. MAC addresses are used in layer 2 of operation [14].

## 2.2 Cuckoo Search Algorithm

The sound of cuckoo birds is beautiful and they differ from other birds based on their strategies in reproduction. Some species of cuckoos lay eggs in other birds' nests or they even remove eggs from the nest of the same species. The probability of eggs available will be increased because of their actions. Several species lay eggs on the nests of other birds and this is termed as brood parasitism which is further classified as intra-specific, co-operative and nest takeover types. While the host bird finds the egg does not belong to its category it starts throwing these eggs or else they leave the nest and end up in building a new one [15].

The rules followed in cuckoo search as follows,

(a) One egg is being laid by every cuckoo at a specific time and is being dumped in a nest chosen randomly.
(b) The nest with eggs of high quality are chosen to be carried to subsequent generations.
(c) Count of the host nest is being fixed along with the probability of the egg being discovered. Host birds leave the nest or can throw away the egg and result in building a new nest.

Performance of protocols such as DSDV as well as AODV is analyzed in environments such as wireless sensor networks. With various node configurations, various parameters for analyzing such as PDR values and energy consumed by the nodes [16]. Various researchers have carried out research on the protocols in ad hoc networks. The protocols vary from one another as they use various methods and metrics for selection of routes among the source and the destination. Count of the nodes along

with the time taken for pausing are taken as parameters for the analysis process. Comparison of DSDV with other protocols such as DSR and AODV is performed. Additionally the proposed work included parameters such as throughput, count of packets received and hop count [17]. The proposed system performed simulation using 30 nodes with the help of OPNET. With the size of the queue also comparison results were found among the routing protocols [18].

The researchers studied the protocols under the following scenarios with the level of mobility as low, medium and high. In the scenarios where TCP as well as UDP is used, protocols such as DSDV show better performance in comparison with AODV. DSDV provides the best performance while using the packets in the traffic of type TCP. As the existing systems suffer from several limitations, the proposed work makes use of CSA for finding the path from source to destination.

## 3 Proposed Work

The proposed system works on improving the DSDV protocol used in routing along with the cuckoo search algorithm which is employed in identification of path that is best among the nodes in the path. This approach works on two stages where in the first stage is topology identification followed by the second stage which comprises management of routing.

Algorithm

begin

(a) Topology identification to find the source and destination

(b) Management of routing using CS algorithm

end

### 3.1 Topology Identification

The first phase begins with the node considered as the source and finds the route to reach the particular destination. Unavailability of the route initially is resolved using the proposed approach. Broadcasting is performed using route request packet. Cuckoo search is used for building the initial population and the information regarding the nodes on the intermediate routes are added to this packet. Finally the set of solutions is built. The nodes available in the route from source to destination are identified and the id of the nodes in between them are stored in a table representing the solution. The packet information is received only by those nodes who are related to both the source and destination. This packet upon reaching the destination, the packet header is stored for creation of populations for solutions.

Graph can be used for representing the MANET where the V consists of a set of stations and E consists of communication links available in the path from source to destination. Upon representing these topologies as graphs, graph theories can be used in problem solving. AT the destination node, the population is constructed now. Adjacency matrix is created with the data based on the links of communications available among the nodes that are intermediary. This information is available in the packet header also. This matrix is used for representing the links available among the network nodes which have obtained the corresponding packets.

## 3.2 Management of Routing

Cuckoo search is applied to the population formed in the initial phase for the purpose of optimization of the search process while finding the route among the nodes. For optimizing, the fitness function is used in detection of distance among the source and destination. Fitness value is calculated based on the count of hops available already. Again with this value, calculations for new solutions are done with the help of Levy flight. Generation of new solutions takes place using Levy flights in a cuckoo search algorithm. Maintaining the nodes including the source as well as destination is done by using the changes done to the nodes that are intermediate. In the case of solutions that are invalid, the algorithm for repairing is discussed.

The three sections of the algorithm are as follows, (a) Determination of presence of link among the nodes. Upon the presence of links, the nodes are not disturbed. In absence of the link, the second section is utilized. (b) Connection between the node and the preceding node is checked. Existence of connection with the first nodes results in removal of all the other intermediate nodes. In the absence of such nodes, a third section is utilized. (c) Entry of the node in the neighborhood of the node chosen at first whose connection is present using the node present second.

## 4 Results and Discussion

NS-2 called Network Simulator is utilized for simulating the environment and can be executed on Windows as well as Unix. The simulation of the model is done for DSDV at the level of the packets leading to the result analysis of the simulation. To analyse the performance of the protocols used in routing, node configuration and the PDR value as well as dropped packets is taken as an important measure. The overall throughput of the network depends on the PDR value as it also helps in estimating the rate of loss of the packets.

PDR stands for Packet Delivery Ratio and the value for PDR is greater for the proposed system using CSA than the existing system using AODV protocol. The count of packets that reach the destination is used in computing the PDR value.

Figure 2 shows the comparison of the PDR value for both the proposed and existing systems.

The proposed system works effectively as the amount of packets dropped is much less compared to the existing system. Due to the structure of DSDV, the count of packets lost is always lesser to other systems in this environment. The same node configuration is used here as those used in computing the PDR values as shown in Fig. 3. The proposed system works on improving the DSDV protocol used in routing along with the cuckoo search algorithm which is employed.
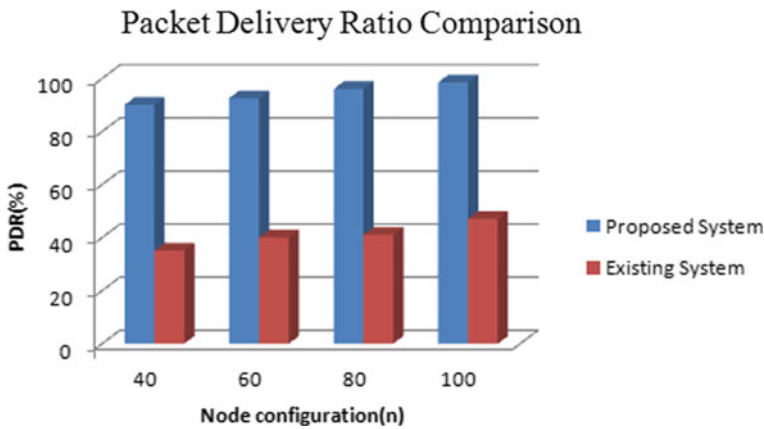


**Fig. 2** Comparison of Packet delivery ratio for proposed and existing system
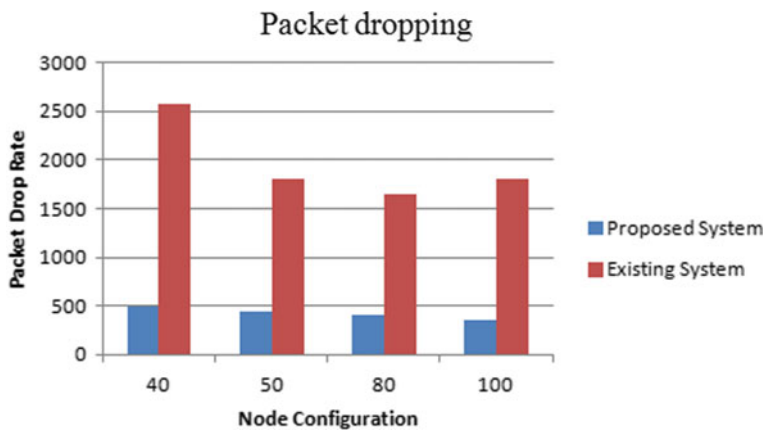


**Fig. 3** Dropped packets rate in the proposed and existing system

## 5 Conclusion

The proposed system utilizes a novel proactive protocol which works based on the metaheuristic algorithm of cuckoo search. The model works in two stages where the first stage deals with the identification of topology followed by the second stage where the management of routing between the source and destination takes place with the CSA. The evaluation of the model is done by implementing it in NS-2 and a comparison of results is performed. Analysis of the system is done using the parameters such as PDR values and the level of packets dropped during the data transfer. This approach has produced effective results thereby making it suitable for networks such VANET called Vehicular Ad Hoc Network as well as FANET called Flying Ad Hoc Network.

## References

1. Dash M, Balabantaray M (2014) Routing problem: MANET and ant colony algorithm. Int J Res Comput Commun Technol 3(9):954–960
2. Perkins C, Watson T (1994) Highly dynamic destination sequenced distance vector routing (DSDV) for mobile computers. In: ACM SIGCOMM conference on communications architectures
3. Jacquet P, Muhlethaler P, Clausen T, Laouiti A, Qayyum A, Viennot L (2001) Optimized link state routing protocol for ad hoc networks. In: IEEE international multi topic conference (INMIC), pp 62–68
4. Perkins C, Royer E (1999) Ad-hoc on-demand distance vector routing. In: Proceedings of WMCSA, pp 90–100
5. Johnson DB, Maltz DA (1996) Dynamic source routing in ad hoc wireless networks. In: Imielinski T (ed) The Kluwer international series in engineering and computer science, series vol 353 of Mobile Computing. Springer, pp 153–181
6. Hass Z, Pearlman R (2002) Zone routing protocol for ad hoc networks. IETF MANET, draft-ietf-manet-zone-zrp-04.txt
7. Di Caro G, Ducatelle F, Gambardella LM (2004) AntHocNet: an ant-based hybrid routing algorithm for mobile ad hoc networks. In: Proceedings of parallel problem solving from nature (PPSNVIII), series vol 3242 of LNCS. Springer, pp 461–470
8. Yang X, Deb S (2009) Cuckoo search via Lévy flights. In: Nature & biologically inspired computing NaBIC'2009, Coimbatore, pp 210–214
9. Chandra SJ, Sivarama PR (2015) Trust predicated routing framework with optimized cluster head selection using cuckoo search algorithm for MANET. IEIE Trans Smart Proc Comput 4(2):115–125
10. Kaur J, Kaur GR (2014) Performance analysis of AODV and DYMO routing protocols in MANETs using cuckoo search optimization. Int J Adv Res Comput Sci Manag Stud 2(8):236–247
11. Kurniawan A, Kristalina P,Hadi MZS (2020) Performance analysis of routing protocols AODV, OLSR and DSDV on MANET using NS3. In: 2020 international electronics symposium (IES), pp 199–206. https://doi.org/10.1109/IES50839.2020.9231690
12. Al-khatib AA, Hassan R (2018) Performance evaluation of AODV, DSDV, and DSR routing protocols in MANET using NS-2 simulator. In: Saeed F, Gazem N, Patnaik S, Saed Balaid A, Mohammed F (eds) Recent trends in information and communication technology. IRICT 2017. Lecture Notes on Data Engineering and Communications Technologies, vol 5. Springer, Cham. https://doi.org/10.1007/978-3-319-59427-9_30

13. Rozy NF, Ramadhiansya R, Sunarya PA, Rahardja U (2019) Performance comparison routing protocol AODV, DSDV, and AOMDV with video streaming in manet. In: 2019 7th international conference on cyber and IT service management (CITSM), pp 1–6. https://doi.org/10.1109/CITSM47753.2019.8965386

14. Setijadi E, Purnama IKE, Pumomo MH (2018) Performance comparative of AODV, AOMDV and DSDV routing protocols in MANET using NS2. In: 2018 international seminar on application for technology of information and communication, pp 286–289. https://doi.org/10.1109/ISEMANTIC.2018.8549794

15. Sureshkumar A, Ellappan V, Manivel K (2017) A comparison analysis of DSDV and AODV routing protocols in mobile AD HOC networks. In: 2017 conference on emerging devices and smart systems (ICEDSS), pp 234–237. https://doi.org/10.1109/ICEDSS.2017.8073696

16. Devi T, Deepa N, Lakshmi Swathi K, Harthik M (2019) A novel protocol to provide authentication and privacy in WSN. Int J Innov Technol Explor Eng 8(9 Special Issue 4):264–266

17. Alghamdi SA (2021) Cuckoo energy-efficient load-balancing on-demand multipath routing protocol. Arab J Sci Eng

18. Gopalan SH (2021) ZHRP-DCSEI, a novel hybrid routing protocol for mobile ad-hoc networks to optimize energy using dynamic cuckoo search algorithm. Wireless Pers Commun 118:3289–3301

# COVIDEffiNet: Pulmonary Diseases and COVID-19 Detection from Chest Radiographs Using EfficientNet Deep Learning Model

**Madhurima Magesh** [ID]**, Hitarth S. Menon** [ID]**, and S. Geetha** [ID]

## 1 Introduction

Background: With the current Coronavirus disease pandemic hitting countries all over the world, what was observed was that there was an undeniable strain on hospital resources and medical staff, especially in countries like India, where the population and population density are extremely high. In times of pandemics like these, in order to maintain a lead and contain the spread of the virus, one of the main objectives would be to know the current situation by keeping a robust and updated record of the number of cases and continuous diagnosing of patients. This is, however, difficult for the medical staff in any country, no matter how modern, especially a highly populated country. Not only is India impacted by this virus but also, for the past several years, there have been many deaths due to other lung diseases like tuberculosis and pneumonia. Pneumonia has claimed 1,27,000 lives in India and it is only second to Nigeria on that list. Tuberculosis claims 2,20,000 deaths every year in India, and is among the top ten leading causes of death according to Rahman et al. in [1]. The Coronavirus disease has claimed 3,41,000 deaths in India, and even in developed nations like the USA, the loss of life toll stands at 5,96,000. The doctor-to-patient ratio in India is (1:1456) which is lower than the WHO-recommended (1:1000).

As of this point, there are two main methods to diagnose COVID-19 infection: Real-Time Reverse Transcription–Polymerase Chain Reaction (RT-PCR) and Rapid Antibody Test (RAT) according to Agrawal et al. [2]. When the infection is spread to a large part of the population there might be a shortage of RT-PCR test kits. Its results are also prone to be erroneous as mentioned in the paper by Jaiswal et al. [3] and time-consuming, this indicates the need for a faster and more efficient solution as suggested by Singh et al. [4]. Lung X-rays are relatively easier to access and

M. Magesh · H. S. Menon · S. Geetha (✉)
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu 600127, India
e-mail: geetha.s@vit.ac.in

can serve as a better alternative for diagnosis as suggested by Chowdhury et al. in [5]. Artificial Intelligence can potentially be extremely useful in this scenario as suggested by Kermany et al. [6]. Identifying and classifying the abnormalities found in these scans of the lungs of patients was considered a tedious and time-consuming task even before the pandemic. Now, with an acute shortage of doctors around the globe, being occupied with data of various patients it is more necessary than ever to have an alternative diagnosis mechanism, to help the healthcare workers in making quick decisions, but also ones which are accurate and precise to an extent of not requiring the approval and supervision of a medical expert.

Hence, the notion of a relatively new mechanism in the vast field of Computer-Aided Diagnosis (CAD), which makes it possible to determine all these major kinds of lung diseases that a person can be infected with and do so with a very reasonable accuracy only seemed relevant and appropriate for pandemic situations like these. It was important along with determining all these diseases to maintain an accuracy as close to 100% as possible, to make sure the solution is feasible, and practically applicable to the medical resources around the world.

This paper is proposed in order to provide a solution which may enhance the speed, efficiency, and accuracy of diagnosis of COVID-19 and other fatal pulmonary diseases. To achieve this the paper proposes an approach and a new model in the field of CAD which uses the EfficientNet model to operate on Lung X-rays and classifies between different categories namely: COVID-19, pneumonia, tuberculosis, and normal. The model is trained on images of chest X-rays of these categories with the aim of maximizing the performance which can be quantified using various metrics which will be discussed in later sections of this paper. At the end of this paper, we will also compare the performance of our proposed model with that of other papers published in similar fields as ours. Figure 1 shows sample images from each category.

In this paper we will discuss the following topics: In Sect. 2 related works, we will discuss the various other papers and the approaches suggested by them, including the dataset sources, models, methodologies, and approaches used. In Sect. 3, we will elaborate on the model and approach that we are going to propose as a solution, where we will elaborate on the model of EfficientNet and transfer learning. In the next section of experimentation, we will be going through the process of our work, the dataset, and the result we obtained. We will also define the metrics based on which we will analyze and quantify our performance and then compare our results with the other papers and their approaches and models.
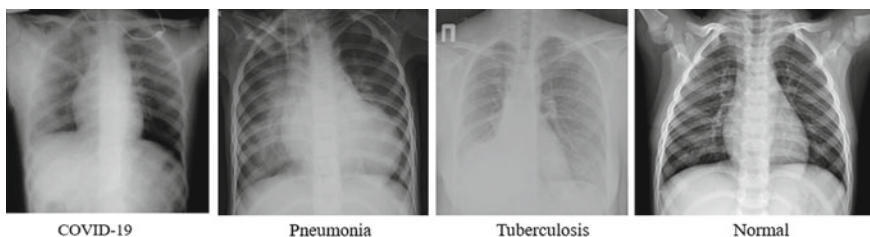


**Fig. 1** Sample image for each category

## 2   Related Works

There have been numerous attempts made in the past to diagnose lung diseases based on CT scans and radiography scans. Hwang et al. [7] proposed a Convolution Neural Network (CNN) model which operates on a dataset of 246 pairs of lung CT scans, each pair containing two CT scans of the same patient's lungs, initially and as a follow-up after a period of 2 years. It aims to differentiate between the three major idiopathic interstitial pneumonias. It also performed retrieval and the metric used to determine the similarity was on a scale of 1–5, with 5 being almost identical and 1 being a different disease. The retrieval was performed with a maximum accuracy of 93%, with an average score of 4.17 out of 5.

Das et al. [8] proposed to implement ensemble learning techniques with CNNs on a dataset consisting of 538 images of COVID-positive patients and 468 images of COVID-negative people. It tries to determine whether a given patient is or isn't infected with the coronavirus. It uses the following CNN models—DenseNet201, Resnet50V2, and Inceptionv3, which are independently trained and combined to give an accuracy of 91.62%.

The paper by Hwa et al. [9] presents an enhancement on the Canny edge detection system. The proposed approach is Contrast-Enhanced Canny Edge Detection or CEED Cany. CEED Canny outperforms the Canny edge detection system. Here, with the help of edge detection and CNN networks, classifications are made into the following categories COVID-19, normal, and pneumonia.

The paper by Mamalakis et al. [10] also tackles the task of classifying chest X-ray images into COVID, normal, pneumonia, and tuberculosis. It proposes a hybrid model consisting of DensNet-121, ResNet-50, and VGG-16, it is called DenResCov-19. The training and validation are done on four datasets, one with pneumonia-positive and -negative samples, another with categories as normal, COVID-19-positive, and pneumonia, and the other two have images belonging to four categories as follows: COVID-19, pneumonia, tuberculosis, and healthy. The precision, recall, and F1-Score for classifying the chest radiographs into one of the four categories was found to be 82.9, 69.7, and 75.8%, respectively.

A similar approach was seen in the work by Vijaykumar et al. [11] which compared CNN models like ResNet50, VGG19, Inception V3, and also traditional CBIR methods like Global Color Histogram, Texture: Gray-level co-occurrence matrix (GLCM), correlation, homogeneity, energy, dissimilarity, and Angular Second Moment (ASM), Shape: Hu-invariant moments) and concluded that ResNet model outperformed every other model with an accuracy of 89.37%.

Zhong et al. [12] use the deep learning model of ResNet-50 as the base model to build the algorithm. It then uses K-Nearest Neighbors (KNN) to extract the ten nearest neighbors, which then makes a weighted majority call for the classification. The classification is made between COVID-19 and non-COVID pneumonia. The accuracies for non-COVID pneumonia and COVID-19 were found to be 94.3 and 92.5%, respectively.

El Asnaoui [13] attempts to compare the performance of different combinations and fusions of the models InceptionResNet_V2, ResNet50, and MobileNet_V2 on the following dataset: one consisting of 5856 images with three classes (1493 viral pneumonia, 2780 bacterial pneumonia, and 1583 normal), while the second one is containing 231 COVID-19 chest X-ray images. These two datasets were joined to perform training on the model and give the output as a classification into the following labels: bacteria, COVID-19, normal, and viral. The dataset consists of 6087 images. Ultimately, it was found that a fusion of all three models combined outperformed the individual models and other combinations of models taken two at a time. The final fusion of three models gave an F1 score of 94.84%.

Kumar et al. [14] use pre-trained models by transfer learning. Two datasets are used here, one from Kaggle, containing X-ray images of pneumonia, tuberculosis, and COVID-positive and COVID-negative patients. The other dataset also consists of similar images. These samples are divided into four subsets depending on the type of image as CXR_Healthy, CXR_Tuberculosis, CXR_Pneumonia, and CXR_COVID. The models used here are EfficientNet, GoogLeNet, and Xception. Ensemble learning was carried out.

From the various related works, we have studied, we see that insufficient dataset is a limitation faced by them. This would in turn lead to overfitting and thus affect the performance of their models in real-life situations. Another limitation we observed was that most models that were used had a large number of parameters. Training them would only increase the computational cost. In our search for a more efficient model which would decrease cost, we came across the EfficientNet model. The smallest EfficientNet Model has 5,330,564 parameters and yet outperforms the ResNet50 model which has 23,534,592 parameters. This means we get better performance and successfully reduce the computational complexity.

## 3 Proposed Methodology

Deep learning is a sophisticated artificial intelligence technique that has grown massively in popularity in recent years as a result of improved processing power and access to more data. Many of the tools we use every day, including virtual personal assistants (Siri, Alexa), personalized recommendations in streaming services like Netflix and e-commerce platforms like Amazon and face recognition security in mobile phones, are powered by this technology. In deep learning, deep neural networks are deployed that try to imitate the functioning of a human brain. Neural networks are made up of layers stacked on layers of variables that change according to the features and of the data they're trained on and may perform tasks like image classification and converting speech to text.

Artificial Neural Networks (ANNs), Convolution Neural Networks (CNNs), and Recurrent Neural Networks (RNNs) are the three most important categories of neural networks that make up the base for most of the pre-trained models used to solve problems. CNNs are most commonly used for image processing. They comprise

kernels/filters that extract features of the image, especially the spatial features. We have used CNN to help solve the problem at hand.

## 3.1 Model

It is a known fact that modeling and scaling can be done on any CNN architecture. But at the same time, the choice of the initial model to scale is found to make a difference in the final output. So, the baseline architecture chosen was EfficientNet-B0. The EfficientNet model was introduced by Google's Research Brain Team comparison of Tan et al. [15]. They came up with a scaling strategy that would scale the depth, width, and resolution of the network and thereby attain higher performance. As mentioned in Sect. 2, they are able to perform better than CNNs like ResNet50 in spite of having lesser parameters which would eventually reduce cost.

It was trained with a multi-objective neural architecture search that optimizes both accuracy and FLOPS. We use Keras to load the model pre-trained on ImageNet which is a large dataset of around 1000 images for each synonym set. Its architecture is shown in Fig. 2.

| Stage i | Operator $F_i$, | Resolution $H_i \times W_i$ | #Channels $C_i$ | #Layers $L_i$ |
|---------|-----------------|------------------------------|------------------|----------------|
| I | Conv3x3 | 224 x 224 | 32 | I |
| 2 | MBConv1, k3x3 | 112 x 112 | 16 | I |
| 3 | MBConv6, k3x3 | 112 x 112 | 24 | 2 |
| 4 | MBConv6, k5x5 | 56 x 56 | 40 | 2 |
| 5 | MBConv6, k3x3 | 28 x 28 | 80 | 3 |
| 6 | MBConv6, k5x5 | 14 x 14 | 112 | 3 |
| 7 | MBConv6, k5x5 | 14 x 14 | 192 | 4 |
| 8 | MBConv6, k3x3 | 7 x 7 | 320 | I |
| 9 | Convix1 & Pooling & FC | 7 x 7 | 1280 | I |

**Fig. 2**  Architecture of EfficientNet-B0

## 3.2 Transfer Learning

Transfer learning is a machine learning method where a model developed for a task is reused as the starting point for a model on a second task. It is a popular approach in deep learning where pre-trained models are used as the starting point on computer vision and natural language processing tasks given the vast compute and time resources required to develop neural network models on these problems and from the huge jumps in skill that they provide on related problems. We use a pre-trained model which is a neural network model which has been trained on a dataset with the same dimensions as the problem at hand, and preferably also has a similar problem statement or a similar domain. Using a pre-trained model is a way to implement transfer learning. Alternate ways of doing this would involve designing and training a model for different datasets which are large in number. This would require a lot of training time and much larger dataset of images. Hence, it is practical to use a model that has been pre-trained and use its learning.

This is a way of implementing transfer learning, one might argue as to why not design a model from scratch instead of implementing transfer learning. Let us say there was an attempt made to design a neural network from scratch, it would result in two things; first, the training time would increase by a large amount, and second, and more importantly, we would not achieve the same accuracy and result with the same amount of data available. Since the model would be training from scratch, it would require much more images from each category in order to obtain a performance as close to the one that is achieved by using transfer learning. Making use of transfer learning essentially means that we are utilizing the learning of another model with a slightly different and much larger training dataset. In a scenario where getting more training data would mean more patients infected, it is advisable to work with the data available and ensure maximum learning.

We have performed transfer learning for our project. We take the EfficientNetB0 model of Keras as the base model to which other layers are added. In our project, we have added five dense layers, one dropout layer, and one flatten layer. Given below in Fig. 3, is the summary of the overall model that is being used (see Figs. 4 and 5). The architecture is shown in Fig. 4. We use a stochastic gradient descent optimizer with a learning rate of 0.001 and momentum of 0.9. Categorical cross-entropy loss and accuracy metric are used for training. We have used ReduceLROnPlateau, a callback that monitors the metric and if no improvement is seen for three epochs, the learning rate is reduced by a factor of 0.01.

The original train data is then split into two sets, train data and validation data. This is then fitted over an image generator which generates batches of tensor image data with real-time data augmentation. The images are then passed over for training. We have trained for 40 epochs in total, which took a large amount of time. The model once trained was saved in the drive to avoid having to go through the training process repeatedly.

```
Layer (type)                     Output Shape          Param #
===============================================================
efficientnet-b0 (Functional)     (None, 1280)          4049564

flatten (Flatten)                (None, 1280)          0

dense (Dense)                    (None, 1024)          1311744

dense_1 (Dense)                  (None, 512)           524800

dense_2 (Dense)                  (None, 256)           131328

dropout (Dropout)                (None, 256)           0

dense_3 (Dense)                  (None, 128)           32896

dense_4 (Dense)                  (None, 4)             516
===============================================================
```

**Fig. 3** Model summary
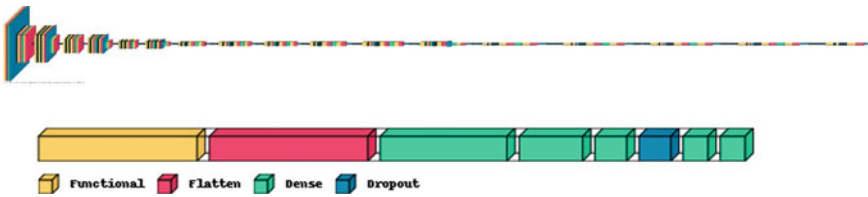


**Fig. 4** Architecture of the proposed model. The functional layer refers to the base model (EfficientNetB0)

## 4 Experimentation

### 4.1 Dataset

The database chosen is a combination of three different datasets obtained from Kaggle. The first one is the tuberculosis (TB) Chest X-ray Database collected by Rahman et al. [1] of 961 MB, containing two subfolders, namely TB-positive and TB-negative, which contain 3500 images each. The second dataset is chest X-Ray images (Pneumonia) which was assembled by Kermany et al. [6] of lung X-rays of pneumonia-positive and -negative patients 2 GB in size, it contains three files: test, train, and value each having two files each NORMAL and PNEUMONIA. Under the PNEUMONIA file, there are images of patients with bacterial pneumonia and viral pneumonia. The third dataset is the COVID-19 Radiography Database used
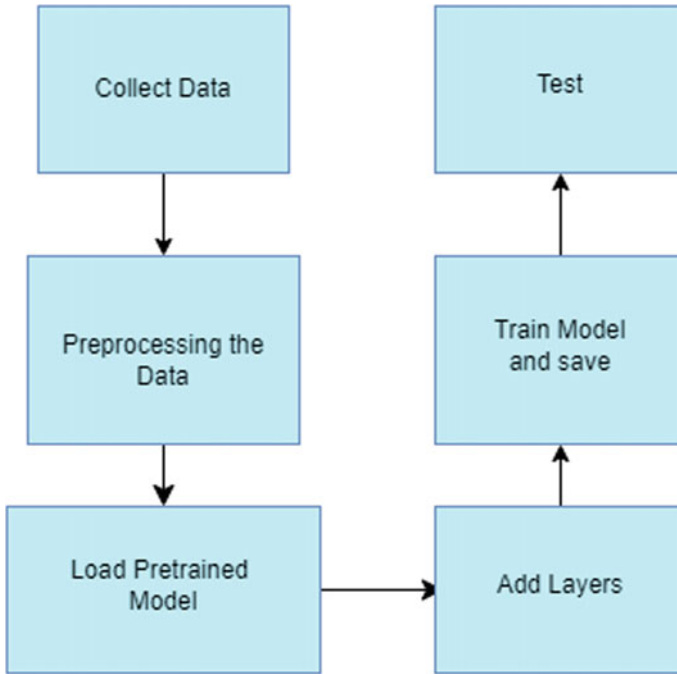
**Fig. 5** Flowchart for COVIDEffiNet

and introduced in the studies by Chowdhury et al. [5] and Rahman et al. [16]. Its size is 745 MB and contains four sub-folders: COVID, lung opacity, normal, and viral pneumonia. Samples are taken from each dataset and stored in a new folder on Google Drive. This drive folder contains four sub-folders for each of the following categories: COVID, pneumonia, tuberculosis, and normal lung images.

The processing is done on Google Colab with Python as the language of implementation and the data is accessed from the drive. The images are stored as matrices along with their respective labels. Then LabEncoder and OneHotEncoder are used to give the labels a proper format, since, initially, they are simply strings. This would put the output label in the form of four columns, one for each category. (Out of the four columns, one would have the value 1.0 and the rest would have the value 0.0). After that, we would need the output labels to be in a single row, for doing that we would loop through all the images and assign the label as the column number which holds the value 1.0. Before this is done, we take a count of images in each of the categories in order to know which number stands for which category.

The overall dataset consists of 3,616 images of COVID-positive, 4273 pneumonia-positive, 3500 tuberculosis-positive, and 4232 normal lung scans. These 15,621 images were divided into train and test data. This gives us 10,934 training images and 4687 test images. For reducing computation complexity, the model was trained on randomly selected 8000 images with balanced classes. It was then tested on 2000 images.

## *4.2 Result*

On testing with the test dataset, we found the classification accuracy to be 93%. Below, in Fig. 6 is the confusion matrix. The labels numbered 0, 1, 2, and 3 referring to COVID, normal, pneumonia, and tuberculosis classes, respectively. When observed closely, we see that the maximum misclassification is by COVID-positive images. Ninety-one times a person infected with the coronavirus has been misclassified as normal and healthy. While looking at the overall size of 2000 images this is a small number. False negatives could be due to various factors like taking the scans at a very early stage before the lung is affected, or too late when the lung starts to clear up, low viral loads, and restricted to the upper respiratory tracts.

The quantitative performance of the model can be analyzed from the sensitivity, specificity, precision, and F1-score metrics which are given in the table.

In order to analyze the performance of our model we consider different parameters that indicate different aspects of the performance which will also help us to quantitatively compare our performances to other papers. The parameters are as follows:

- Precision can be thought of as the percentage of predictions that are correct from all the predictions made for that particular class. It essentially tells you of all the predictions made and what percentage of them were right.
- Sensitivity or recall is the percentage of correct classifications made for a class out of all the instances of that class. It gives you the percentage of the correct



**Fig. 6** Confusion matrix for test data prediction
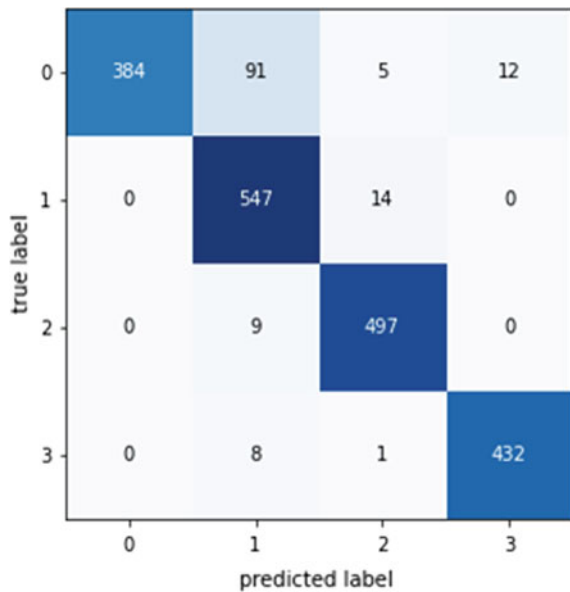
| Actual Class | True Positive (TP) | False Negative (FN) |
|---|---|---|
| | False Positive (FP) | True Negative (TN) |

**Fig. 7** Generalized confusion matrix

instances recalled from the class. That is out of all the instances of the class, how many are correctly remembered by the model?

- Specificity of a class is basically how many instances that do not belong to that class are correctly discarded.
- Accuracy is the total number of correct predictions made out of the total number of predictions.
- F1-score can be thought of as the weighted average of precision and recall. It takes both false positives and false negatives into account; it is more useful than accuracy.

The classification predictions made by the model can primarily be categorized under four types, which are: True Positive, False Positive, True Negative, and False Negative. Once these numbers are obtained it is easy to calculate the metrics mentioned in Fig. 7.

Given below are the formulae of the metrics mentioned above:

$$\text{Accuracy} = (TP + TN)/(TP + FP + TN + FN) \tag{1}$$

$$\text{Precision} = TP/(TP + FN) \tag{2}$$

$$\text{Recall(Sensitivity)} = TP/(TP + FN) \tag{3}$$

$$F1\text{Score} = 2 * (\text{Recall} * \text{Precision})/(\text{Recall} + \text{Precision}) \tag{4}$$

$$\text{Specificity} = TN/(TN + FP) \tag{5}$$

In Table 2, we summarize the results of the various other similar approaches suggested by different papers. Here, we have tabulated the performances of different approaches based on the metrics used in their respective papers. Most of them judged their performance based on the accuracy of their model. In the first paper by Mamalakis et al., the overall recall of the hybrid model is 69.7%. The recall of the different classes by our model (shown in Table 1) is higher than the given 69.7%. The F1-scores of all four classes are also higher than the overall F1-score for the DenResCov-19 which is 75.75%.

In Zhong et al. [12], the average accuracy is given to be 83.9%. The sensitivity for the three categories: control, non-COVID pneumonia, and COVID is 74.3, 89.3, and 85.0%, respectively; and the PPV for the above three categories are 79.4, 64.5, and

**Table 1** Class-wise sensitivity, specificity, precision, and F1-score of the proposed model

|  | Normal | COVID | Tuberculosis | Pneumonia |
|---|---|---|---|---|
| Sensitivity (Recall) (%) | 97.50 | 78.04 | 97.95 | 98.22 |
| Specificity (%) | 92.49 | 100 | 99.23 | 98.66 |
| Precision (%) | 83.51 | 100 | 97.29 | 96.13 |
| F1-score | 0.899644 | 0.876657 | 0.976188844 | 0.971637623 |

**Table 2** Performances comparison of various models with our proposed approach

| Related work | Classes | Model | Accuracy (%) |
|---|---|---|---|
| Mamalakis et al. [10] | COVID TB Pneumonia Normal | DenResCov-19 | 75.75 |
| Zhong et al. [12] | COVID Normal pneumonia TB | ResNet-50, KNN | 83.9 |
| Hwa et al. [9] | COVID-19 Normal Pneumonia | Contrast Enhanced Canny Edge Detection | 86.7 |
| COVIDEffiNet (Proposed model) | COVID-19 Tuberculosis Pneumonia Normal | EfficientNet | 93 |

95.2%. The accuracy obtained by our proposed approach is 93%. The sensitivities for non-COVID pneumonia and COVID obtained in the ResNet approach were 89.3 and 85%, respectively. Sensitivities of our model for pneumonia and COVID were 98.22 and 78.04%, respectively.

The classification of Chest X-rays done in Hwa et al. [9] into COVID-19, pneumonia, and normal categories is done using edge detection and two separate networks, VGG16 classifier and Inception V3 classifier. The overall accuracy rate is given to be 86.7%. Our proposed model exhibits superior performance to them and provides an accuracy of 93%.

Finally, to consider a paper that detects COVID-19 with a neural network trained from scratch, no transfer learning is applied. The paper by Bhoumik et al. [17] considers data for training which consists of 500 images in both categories of COVID-positive and -negative, which is divided into 70–30 ratio for training and validation. The final testing is done on 25 images. The metric used to assess the performance is the F1-score. The model obtains a perfect F1-score of 100%. The catch however is that the testing data was extremely small in size and it is not a good indication of the real-world scenario. For a dataset which contains as small as 500 images of only two categories, for training, it can be well assumed that the high F1-score can be attributed to overfitting in the dataset. Hence, the approach proposed here may have a good F1-score theoretically, but the paper itself claims it is due to the small

dataset considered. In the practical scenario, this approach might not work in order to reach the same performance, it will need a much larger dataset and training time for the neural network to give any solution that stands applicable in the real world. Hence, the proposed approach in our paper of using transfer learning and obtaining a good performance is superior to construction of a neural network from scratch.

## 5 Conclusion

We hence conclude by reporting our findings and the performance of the model. We have trained the COVIDEffiNet model on the dataset of COVID-19, pneumonia, and tuberculosis-positive and healthy (negative) chest radiography scans. On testing with 2000 images, we obtained a final classification accuracy of 93%. When comparing with other works, we see that most of them have tested on a smaller number of images as opposed to us. Many of them restrict their classification to only two or three categories. COVIDEffiNet aims to classify into four categories on the other hand. This proves our model's superior performance. The proposed system not only reduces the strain on the healthcare system but also brings down the misclassifications that could have occurred due to human error. For future works, we plan to improve the performance of the model and bring it as close as possible to 100%. We also plan to design a web application with a backend which stores the model and its learning process so that it continuously learns from input images. With a good and intuitive user interface, it would be possible to make this application open to use in the medical domains and be deployed in healthcare institutions as technologies like these may prove to be pivotal in saving lives and time, not only in a pandemic situation but also in the long-term fight for the eradication of lung diseases and ailments.

## References

1. Rahman T, Khandakar A, Kadir MA, Islam KR, Islam KF, Mazhar R, Hamid T, Islam MT, Mahbub ZB, Ayari MA, Chowdhury ME (2020) Reliable tuberculosis detection using chest X-ray with deep learning, segmentation and visualization. IEEE Access 8:191586–191601. https://doi.org/10.1109/ACCESS.2020.3031384
2. Agrawal T, Choudhary P (2021) FocusCovid: automated COVID-19 detection using deep learning with chest X-ray images. Evol Syst, 1–15.https://doi.org/10.1007/s12530-021-09385-2
3. Jaiswal A, Gianchandani N, Singh D, Kumar V, Kaur M (2021) Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. J Biomol Struct Dyn 39(15):5682–5689. https://doi.org/10.1080/07391102.2020.1788642
4. Singh RK, Pandey R, Babu RN (2021) COVIDScreen: explainable deep learning framework for differential diagnosis of COVID-19 using chest X-rays. Neural Comput Appl, 1–22. Advance online publication. https://doi.org/10.1007/s00521-020-05636-6
5. Chowdhury M, Rahman T, Khandakar A, Mazhar R, Kadir M, Mahbub Z et al (2020) Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 8:132665–132676. https://doi.org/10.1109/access.2020.3010287

6.  Kermany DS, Goldbaum M, Cai W, Valentim C, Liang H, Baxter SL, McKeown A, Yang G, Wu X, Yan F, Dong J, Prasadha MK, Pei J, Ting M, Zhu J, Li C, Hewett S, Dong J, Ziyar I, Shi A, … Zhang K (2018) Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 172(5):1122–1131.e9https://doi.org/10.1016/j.cell.2018.02.010

7.  Hwang HJ, Seo JB, Lee SM, Kim EY, Park B, Bae HJ, Kim N (2021) Content-based image retrieval of chest CT with convolutional neural network for diffuse interstitial lung disease: performance assessment in three major idiopathic interstitial pneumonias. Korean J Radiol 22(2):281–290. https://doi.org/10.3348/kjr.2020.0603

8.  Das A, Ghosh S, Thunder S, Dutta R, Agarwal S, Chakrabarti A (2021) Automatic COVID-19 detection from X-ray images using ensemble learning with convolutional neural network. Pattern Anal Appl 24(3):1111–1124. https://doi.org/10.1007/s10044-021-00970-4

9.  Hwa SKT, Bade A, Hijazi MHA (2020) Enhanced Canny edge detection for Covid-19 and pneumonia X-Ray images. In: IOP conference series: materials science and engineering, vol 979, no 1, p 12016https://doi.org/10.1088/1757-899X/979/1/012016

10. Mamalakis M, Swift AJ, Vorselaars B., Ray S, Weeks S, Ding W, Clayton RH, Mackenzie LS, Banerjee A (2021) DenResCov-19: a deep transfer learning network for robust automatic classification of COVID-19, pneumonia, and tuberculosis from X-rays. Comput Med Imag Graph Off J Comput Med Imag Soc 94:102008. Advance online publication. https://doi.org/10.1016/j.compmedimag.2021.102008

11. Bhandi V, Sumithra Devi KA (2020) COVID-19 X-ray image retrieval using deep convolutional neural networks. Am J Eng Res (AJER) 9(07):47–55

12. Zhong A, Li X, Wu D, Ren H, Kim K, Kim Y, Buch V, Neumark N, Bizzo B, Tak WY, Park SY, Lee YR, Kang MK, Park JG, Kim BS, Chung WJ, Guo N, Dayan I, Kalra MK, Li Q (2021) Deep metric learning-based image retrieval system for chest radiograph and its clinical applications in COVID-19. Med Image Anal 70:101993. https://doi.org/10.1016/j.media.2021.101993

13. El Asnaoui K (2021) Design ensemble deep learning model for pneumonia disease classification. Int J Multimed Info Retr 10:55–68. https://doi.org/10.1007/s13735-021-00204-7

14. Kumar N, Gupta M, Gupta D, Tiwari S (2021) Novel deep transfer learning model for COVID-19 patient detection using X-ray chest images. J Ambient Intell Hum Comput, 1–10. Advance online publication. https://doi.org/10.1007/s12652-021-03306-6

15. Tan M, Le QV (2019) EfficientNet: rethinking model scaling for convolutional neural networks. https://arxiv.org/abs/1905.11946

16. Rahman T, Khandakar A, Qiblawey Y, Tahir A, Kiranyaz S, Abul Kashem SB, Islam MT, Al Maadeed S, Zughaier SM, Khan MS, Chowdhury M (2021) Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. Comput Biol Med 132:104319https://doi.org/10.1016/j.compbiomed.2021.104319

17. Bhoumik S, Chatterjee S, Sarkar A, Kumar A, Joseph FJJ (2020) Covid 19 prediction from X-ray images using fully connected convolutional neural network. In: CSBio '20: Proceedings of the eleventh international conference on computational systems-biology and bioinformatics (CSBio2020). Association for Computing Machinery, New York, NY, USA, pp 106–107. https://doi.org/10.1145/3429210.3429233

# Multi-classification of Brain Tumor MRI Images Using Deep CNN Features and ML Classifiers

Check for updates

**B. Sandhiya** and **S. Kanaga Suba Raja**

## 1 Introduction

Brain tumor existence may be conjectured when an aberrational development of cells propagates in an unrestrained manner. Tumors can start from different areas of brain like cells, membranes, and glands/nerves. Brain cells can be destroyed particularly by tumors. Brain tumors can cause more pressure inside the skull that can especially damage the brain cells [1]. Basically, the hazardous forms of tumors are malignant tumors. Each year the death count is nearly 14,000. The survival rate of malignant tumor (glioma and glioblastoma) patients still remains around 15 months [2]. Brain tumors can be separated into diverse classes of phases based on their rigorousness level like phases 1–4. In that, phase-1 class of tumors is least dangerous and in general associated with extended existence. Surgical treatment can be successful for this type of grade 1 tumors. The causes of phase-1 tumors are Pilocytic astrocytoma, ganglioglioma, and gangliocytoma. The next category is phase-2 tumor that grows gradually and looks irregular through a microscope. In this, some will spread into the neighboring cells repeatedly as higher grade tumors [3]. Phase-3 tumor is the next category which is malignant and the highest category in tumor categorization is phase-4, which is the utmost malignant tumor. It has an unusual structure when we look through the microscope and it also spreads rapidly and into the neighboring cells and the resultant will be the new-fangled brain vessel structure. The phase-4 tumor cells will have dead cell zones in their middle and the example of this phase-4 type is glioblastoma multiform [4].

The different types of gliomas can be described from the phrase glioma as high-phase (HP) tumors, and low-phase (LP) tumors. Chemotherapy, radiotherapy, and

B. Sandhiya (✉)
Department of Information Technology, Easwari Engineering College, Chennai, India
e-mail: sandhiya5705@gmail.com

S. Kanaga Suba Raja
Easwari Engineering College, Chennai, India

surgery are the available treatments for treating gliomas [5]. Radiographic images can assist health professionals to accurately diagnose and treat tumors. The brain tumor diagnosis involves three distinct techniques, like tumor discovery, tumor segmentation, and tumor categorization [6].

The BT features like location, size, intensity, structure, and kind of BT phase can be diagnosed with the help of diverse types of imaging modalities like X-ray, PET Scan (Positron emission tomography), MRI (MR Imaging), MRA (MR angiography), MRS (MR Spectroscopy), and CT. Large magnets and radio waves are used by MRIs to examine the organs and their structures. MRI scans are widely used by healthcare professionals to diagnose a multiplicity of conditions like ligaments, cysts, tumors, etc. Magnetic resonance imaging is highly helpful in detecting diverse conditions in brain like tumors, hemorrhages, cysts, bulges, abnormalities, infections, and provocative conditions/problems related to blood vessels. So, MRIs are very helpful for investigating the brain and spinal cord [7, 8]. Figure 1 shows the process flow of brain tumor diagnosis with its detailed phases.

Generally, for brain tumor diagnosis the clinical image has to be processed and pattern recognition can be done using various ML and DL techniques. For brain tumor diagnosis, magnetic resonance image will be taken as an input for extracting the tumor portion. To increase the image quality, the input image can be pre-processed to expropriate noise and augment some image features in the image. Once the images are upgraded by the pre-processing styles and methodologies, the upcoming phase is tumor segmentation from the given input image. Then the input image is partitioned into multiple sections to locate tumor and its boundaries. In spite to section and spot the BT (brain tumor), the most important characteristics of the input datasets should be fetched properly to locate the tumor accurately. After the feature extraction process, the redundant and non-effective features should be reduced or discarded to decrease the huge input dataset of features by feature selection. Once the necessary attributes are chosen the tumor classification may be completed effectively. The
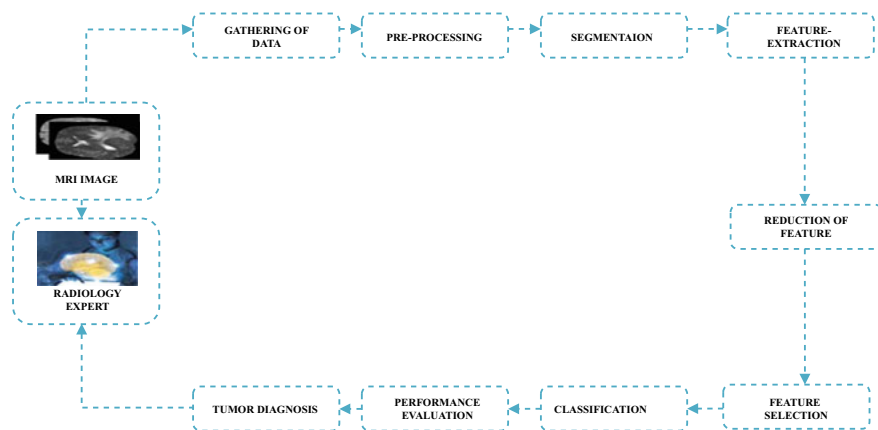


**Fig. 1** Process flow of brain tumor diagnosis
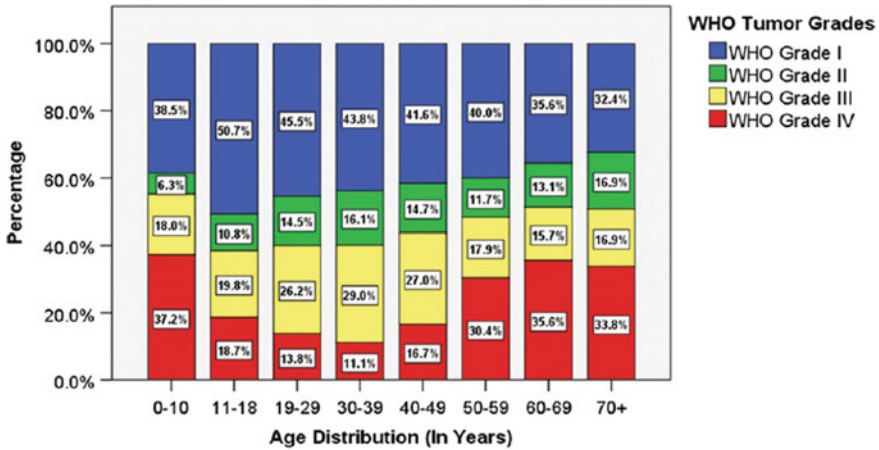
**Fig. 2** WHO's tumor grades (I, II, III, and IV) distribution based on the age category

classification process is nothing but categorizing and labeling the groups of vectors in the image based on the requirement. In our study, the BTs can be distributed as benign/malignant, also the labels can be LGG (Low-Phase Gliomas) or HGG (High-Phase Gliomas). Finally, the obtained results can be evaluated based on accuracy. If the performance measures are greatly satisfactory, then the brain tumor detection and diagnosis process can be done with better accuracy.

Conforming to World Health Organization (WHO), 36.3 percentage of tumors are categorized as phase-1, 11.4 percentage as phase-2, 20 percentage as phase-3, and 18.9 percentage as phase-4 tumors. As per the WHO Grading System, about 13.3 percentage tumor cases are still unclassified. Figure 2 gives the detailed age-wise tumor distribution with the tumor grade percentage. From the above results, the Phase-IV tumor proportions are increased with increasing age [50].

## 2 Literature Survey

### 2.1 Machine Learning-Based Brain Tumor Diagnosis

#### 2.1.1 ML-Based Feature Extraction

ML (Machine Learning) is a concept of training our machine to do some task, by allocating sets of training data. Machine learning has some models, loss functions, and a number of approaches used to decide which would afford better results.

The traditional machine learning methods involve different steps like pre-processing, attribute extraction, attribute reduction, and attribute categorization. The accurate classification of the brain tumor attribute/feature extraction will play a

vigorous part in the traditional ML methodologies. In feature extraction, the first category is low-stage (LS) features like texture, instance, and intensity, first- and second-order statistics. The first- and second-order statistics are implemented using the least square SVM and also binary classifier is used to distribute the ordinary and anomalies of MRI brain images [9]. John et al. [10] made use of a GLCM (gray-level co-occurrence) matrix and discrete wavelet transformation (DWT) for brain tumor detection and classification.

Ullah et al. [11] used FF-ANN (Feed Forward Artificial Neural Network) to recognize the normalities/abnormalities from MR images of the brain. The second category of attribute extraction is high-stage (HS) features like FV, SIFT, and BoW. Various experimenters have signified the BoW for medical image grade assessment and attribute recovery like bone tissue viscosity categorization in mammograms [12]. Retrieval and classification of organs are done by the X-Ray images [13], and content-predicated recovery of brain tumors [14]. Cheng et at. [15] implemented a model for brain tumor retrieval using the feature vector.

The limitations of traditional ML techniques in the attribute extraction phase can be the handcrafted feature extraction process and the relatively high probability of human errors in the handcrafted feature extraction. Either low-stage attributes or high-stage attributes can be concentrated on, for brain tumor classification. The development of a fully automatic feature extraction technique is really in great demand to merge both the high-level features and the low-level features. These confines can be overcome using some deep learning techniques.

### 2.1.2 Machine Learning-Based Brain Tumor Classification

Various automatic segmentations of brain tumor works are done in the existing works/study. Several researchers have implemented various methodologies to diagnose and perform tumor region segmentation [16–18]. After the successful segmentation of the brain tumors, they should be categorized into different grades by classification. A binary classifier is used for tumor classification in many works to differentiate benign or malignant tumors [11, 19, 20].

Ullah et al. [11] developed a model using HE (histogram equalization), DWT, and FF-ANN for brain tumor classification. Kharrat et al. [19] classified the brain tumors, using the GN (genetic algorithm) and support vector machine. Papageorgiou et al. [20], differentiated low- and high-grade gliomas using fuzzy cognitive maps and attained an accuracy of 93.22% and 9026%, respectively.

Many research works have been done to detect and classify the brain tumors using MRI images by several machine learning classifiers, like neural network classifer [11, 28, 32], Naïve Bayes classifier [29], AdaBoost classifier [33], k-NN classifier [28], RF classifier [28, 34], SVM classifier [35], and ELM classifier [36]. In [52], the author proposes a model using SVM for classification.

Though many exploratory studies have been done in this BT diagnosis domain, still the effective ML classifier-based BT categorization using MRI images is still a future enhancement work.

Here, the ML- and DL-based feature or attribute extraction techniques and brain tumor classification methods of the previous studies are illustrated. Also, for the effective evaluation of the brain tumor diagnosis, the robust and optimal features are greatly required. The existing works feature selection and evaluation processes with ML classifiers and deep features are given in Table 1.

**Table 1** Summary of BT Attribute or Feature (LS and HS) extraction methods, brain tumor classification methods, and various ML classifiers

| Traditional ml-based brain tumor diagnosis | | | | |
|---|---|---|---|---|
| S.no | Feature extraction | | | |
| | LS-features | | HS-features | |
| | Model/paper | Method | Model/paper | Method |
| 1 | Selvaraj et al. [9] | Least Square Support Vector Machine and Binary Classifier | Bosch et al. [12] | BoW |
| 2 | John et al. [10] | Gray-Level Co-Occurrence Matrix and Discrete Wavelet-Transformation | Avni et al. [13] | BoW |
| 3 | Ullah et al. [11] | Feed-Forward ANN (Artificial Neural Network) | Yang et al. [14] | SIFT |
| | | | Cheng et al. [15] | Feature Vector |
| S.no | Tumor classification | | | |
| | Model/paper | Method/ML-classifier | | |
| 1 | Ullah et al. [11] | Histogram Equalization, DWT, and Feed Forward-ANN | | |
| 2 | Kharrat et al. [19] | Genetic Algorithm and Support Vector Machine | | |
| 3 | Papageorgiouet al. [20] | On Fuzzy Cognitive Maps | | |
| 4 | Kaplan et al. [28] | Neural network classifier, k-NN classifier, RF classifier | | |
| 5 | Kaur et al. [29] | Naïve Bayes classifier | | |
| 6 | Arunachalam [32] | Neural network classifier | | |
| 7 | Ullah et al. [11] | Neural network classifier | | |
| 8 | Minz [33] | AdaBoost classifier | | |
| 9 | Anitha [34] | AdaBoost classifier | | |
| 10 | Rajan [35] | SVM classifier | | |
| 11 | Gumaei [36] | ELM classifier | | |

## 2.2 Deep Learning Technologies for Brain Tumor Diagnosis

### 2.2.1 Deep Learning-Based Brain Tumor Feature Extraction

BT discovery is the process of exactly locating the tumor area. For detecting the brain tumors, data can be fetched or obtained from a few public datasets or local datasets of MR images. The obtained MR images can be preprocessed before applying them as key to detect the tumors. Kang et al. [21] developed an automatic brain tumor segmentation CNN architecture (multi-pathway) model for gliomas, meningiomas, and pituitary tumor classification. A T1W-Contrast-Enhanced MRI public dataset was used in this model. The training phase is comparatively high in this model.

Deepak et al. [22], in this model a pre-trained GoogLeNet is used for the feature extraction. Raja et al. [27], Noise is removed by applying a non-local mean filter and BFC is used for sectioning and attribute extraction is done through dissipating transform, information-theoretic expedients, and wavelet pack-Tsallis-entropy.

### 2.2.2 DL-Based BT-Classification

DL method and its features have been widely used for brain MRI classification [24, 31]. Deepak et al. [22], in this model, a deep-CNN is used to categorize the brain tumors. Çinar et al. [23] CNN models like GoogLeNet, IV3 (Inception-V3), DenseNet-201, AlexNet, and RN50 (ResNet-50) are used for brain tumor classification. Saxena et al. [24] developed IV3, RN50, and VGG-16 models with TL (transfer learning) methodologies for BT classification. Raja et al. [27] performed brain tumor classification by deep autoencoder with the Bayesian fuzzy-clustering method. A fusion design of DAE is used to classify the brain tumors. High computational time is the drawback of this design.

ML classifiers performances are associated with the input feature map. For a successful MRI-based BT-classification model well-defined and informative feature extraction is very important. In order to fetch the detailed features, nowadays PT-DCNN (pre-trained Deep-CNN) systems are applied for MRI-predicated BT categorization, like ResNet-50 [30, 37], ResNet-101 [38], DenseNet-121 [30, 39], VGG-16 [30, 37], VGG-19 [30, 40], AlexNet [41], Inception-V1(GoogLeNet) [22], Inception-V3 [37, 42], and MobileNet V2 [43]. R-CNN model is used for the effective classification of MRI BT images [51].

The above-stated pre-trained deep-CNN feature models have been used in the past research works for the effective evaluation of the features. An effective, robust, and adaptable feature extraction will help the researchers greatly in brain tumor diagnosis. These deep-CNN feature models are evaluated for their robust feature selection. If the useful feature of the tumor is evaluated effectively, the ultimate outcome will be an accurate one. That is the brain tumor classification and diagnosis accuracy will be improved to the highest degree.

Despite the fact that many research studies have been done in this field, still, the effective evaluation of MR image-predicated BT classification using PT-DCNN systems is still in the improvement stage. For effective feature extraction, pre-trained CNN models are very much supportive to classify brain tumors in MRI. Table 2 states the detailed existing deep learning models used for feature extraction and classification. Also, the pre-trained DCNN models for brain tumor classification are discussed. Generally, by using the pre-trained deep-CNN feature models, the important ones can be selected to give the end results accuracy. So, this work mainly focuses on these pre-trained deep feature models for effective feature selection and BT categorization.

## 3 Performance Analysis

In our work, the brain tumor classification is done using DCNN features and ML classifiers. Here, the transfer learning idea is used with a large number of PT-DCNN models for attribute extraction by the key in MRI images. The performance evaluation is done with the help of the extracted features with several ML classifiers. Based on the evaluation, the top averaged deep features will be selected. The selected features will be given too many ML classifiers to classify the brain tumor accurately.

### 3.1 Analysis of Performance in State of the Art

In our effort, the Kaggle dataset with 3064 T1-W images, namely brain tumor classification containing three categories of BTs like gliomas, meningiomas and pituitary tumors, is used to evaluate the effectiveness on ML classifiers and the pre-trained DCNN models [21]. The acquired dataset is named the BT-DS dataset and the samples of the different classes are shown in Fig. 3. The BT-DS has 4 classes, 2611 training sets, and 653 test sets that have been used for BT detection and classification.

#### 3.1.1 Accuracy Evaluation of BT-DS Dataset with ML-Classifiers and Pre-trained Deep CNN Models

The performance evaluation greatly depends on the extracted features. Here, ten different PT-DCNN feature models are compared with five diverse ML classifiers on BT-DS datasets for evaluating the accuracy and performance analysis. Based on the performance evaluation analysis results the peak averaged three deep features will be selected for brain tumor classification.

From Table 3, DenseNet-169, MobileNet-V2, and MnasNet features are the top three features that were obtained from BT-DS datasets by BT categorization. The detailed graphical representation of the PT-CNN model using ML classifiers is shown

**Table 2** Summary of DL technologies in BT attribute extraction and categorization, as also the attribute extraction methods for MRI-based BT categorization using PT-DCNN models

Deep learning technologies in brain tumor diagnosis

PT-DCNN for tumor feature extraction and classification

| Model/paper | AIM | Classification method | PT-DCNN models | Model/paper | AIM | PT-DCNN models |
|---|---|---|---|---|---|---|
| Kang et al. [21] | Brain tumor (BT) classification | Multi-CNN with Accuracy—90% | PT-CNN | Polat et al. [30] | BT-categorization using PT-DCNN models | RN50,DenseNet-121, VGG-16,19Accu—88% |
| Deepak et al. [22] | Brain tumor (BT) classification | DeepT learning accuracy—90% | GoogleNet | Khan [37] | BT-categorization using PT-DCNN models | ResNet-50, VGG-16, IV3,Accuracy—88.5% |
| Çinar et al. [23] | Brain tumor (BT) classification | CNN with accuracy—89% | ResNet-50 | Ghosal [38] | BT-categorization using PT-DCNN models | ResNet-101 with Accuracy—90.8% |
| Saxena et al. [24] | Brain tumor (BT) detection/classification | CNN with TL Accuracy—91% | Manual Features | Zhou [39] | BT-categorization using PT-DCNN models | DenseNet-121 with Accuracy—90.8% |
| Paul et al. [25] | Brain tumor classification | Fully connected CNN with Accuracy—91% | Manual Features | Saba [40] | BT-categorization using PT-DCNN models | VGG-19 with Accuracy—87% |
| Hemanth et al. [26] | MRI-based BTclassification | CNN with Accuracy—90% | Stochastic GD Algorithm | Soumik [42] | BT-categorization using PT-DCNN models | Inception V3 with Accuracy—89% |

**Table 3** PT-DCNN model Average Accuracy Evaluation with ML classifiers on BT-DS dataset for performance analysis

| PT-CNN | | | | | | |
|---|---|---|---|---|---|---|
| DCNN model's | | | | | | |
| Deep feature | Accuracy of M-learning classifier | | | | | |
| | F–C | GNB | k-NN | R-F | SVM-RBF-k | Average |
| ResNet-50 feature | 0.876 | 0.6937 | 0.8576 | 0.853 | 0.8989 | 0.83584 |
| ResNet-101 feature | 0.8867 | 0.7228 | 0.8438 | 0.8499 | 0.9081 | 0.84226 |
| DenseNet-169 feature * | 0.8959 | 0.7228 | 0.8821 | 0.8652 | 0.9204 | 0.85728 |
| VGG-16 feature | 0.876 | 0.6677 | 0.8331 | 0.83 | 0.8744 | 0.81624 |
| AlexNet feature | 0.8637 | 0.634 | 0.8714 | 0.8453 | 0.9066 | 0.8242 |
| InceptionV3-feature | 0.8653 | 0.6707 | 0.83 | 0.8132 | 0.8867 | 0.81318 |
| Res-NeXt-101 feature | 0.8851 | 0.6692 | 0.8714 | 0.8346 | 0.8989 | 0.83184 |
| ShuffleNet-V2 feature | 0.8637 | 0.7152 | 0.8637 | 0.8576 | 0.9112 | 0.84228 |
| MobileNet-V2 feature * | 0.8928 | 0.6983 | 0.8897 | 0.8423 | 0.9158 | 0.84778 |
| MnasNet feature * | 0.8851 | 0.6922 | 0.8928 | 0.8515 | 0.9127 | 0.84686 |



NO TUMOR          GLIOMAS          MENINGIOMAS          PITUITARY
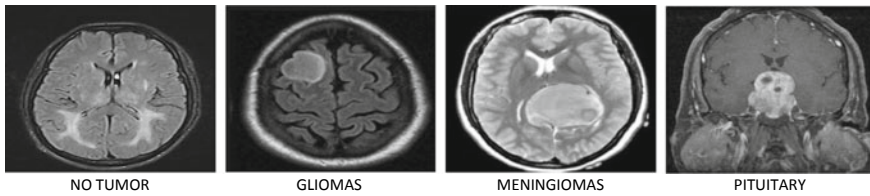
**Fig. 3** Brain MRI image sample in BT-DS dataset

in Fig. 4. Apart from other ML classifiers, SVM (RBF) kernel performance is the best/high on the BT-DS dataset with 92.5% accuracy and, in parallel, the least performed classifier is the GNB with 72.2% accuracy. On the deep features from the other PT-CNN models the DenseNet architectures perform better, and eventually the VGG and InceptionV3 perform worse on the same. As for the computation time, the K-NN classifier's computational time is very high on the test set, whereas FC, GNB, and RFs computational and implication time is less.

However, though the traditional machine learning methods can be useful with some benefits in the field of image processing, the deep learning models show a better and effective performance. The traditional ML algorithm is most familiar for its simplest mechanism. Also, the hardware and structures' computational cost and time are comparatively less for training and testing. The conventional ML algorithms are simple to know and infer.
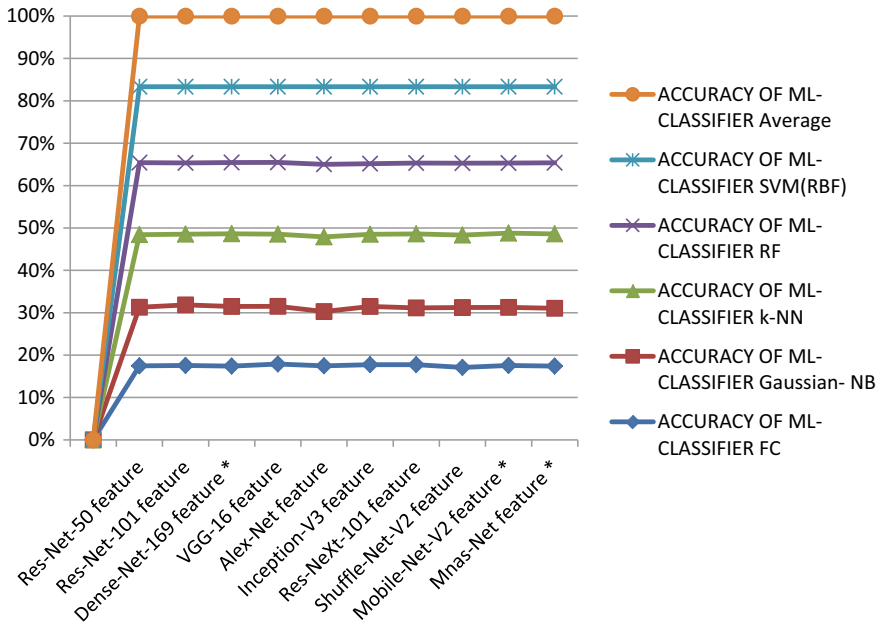
**Fig. 4**  Accuracy range of Pre-trained CNN model with ML classifiers

## 4  Inferences

- Traditional ML methods mainly depend on the handcrafted features.
- Probability of human error is comparatively high.
- In the traditional ML methods, either LS attributes or HS attributes can be concentrated on.
- Development of a completely automatic deep learning-based feature extraction technique is really in great demand to merge both the HL features and the LL features.
- Effectiveness evaluation of MR-Image-based brain tumor classification using PT-DCNN models is still a challenging task.

## 5  Conclusion

Our summary of the study is mainly on BT categorization with deep attributes by the PT-DCNN model. The extracted deep attributes are applied and evaluated with ML classifiers. The peak-selected deep attributes are given to various ML classifiers for evaluating the effective performance of the model. In our work, a wide performance evaluation is done using ten different pre-trained deep-CNN and five diverse ML classifiers on BT-DS datasets. DenseNet-169, MobileNet-V2, and MnasNet features

are very much efficient in large datasets with four different classes like no tumor, gliomas, meningiomas, and pituitary tumor. In wide researches, the SVM Classifier with RBF-k's efficiency is superior when compared with the further ML classifiers for BT categorization by MRI images. In future, a knowledge-based distillation approach can be used to minimize the dimension of the deployment model.

# References

1. Haj-Hosseini N, Milos P, Hildesjö C, Hallbeck M, Richter J, Wårdell K (2016)Fluorescence spectroscopy and optical coherence tomography for brain tumor detection. In: SPIE photonics Europe, biophotonics: photonic solutions for better health care, Brussels, Belgium, 3–7 April 2016. SPIE-International Society for Optical Engineering, pp 9887–9896
2. Liu Y, Carpenter A, Yuan H, Zhou Z, Zalutsky M, Vaidyanathan G, Yan H, Vo-Dinh T (2016)Gold nanostar as theranostic probe for brain tumor sensitive PET-optical imaging and image-guided specific photothermal therapy 4213–4213
3. Louis DN, Perry A, Reifenberger G, Von Deimling A, Figarella-Branger D, Cavenee WK, Ohgaki H, Wiestler OD, Kleihues P, Ellison DW (2016) The 2016 World Health Organization classification of tumors of the central nervous system: a summary. Acta Neuropathol 131(6):803–820
4. Choi SJ, Kim JS, Kim JH, Oh SJ, Lee JG, Kim CJ, Ra YS, Yeo JS, Ryu JS, Moon DH (2005) [18 F] 3′-deoxy-3′-fluorothymidine PET for the diagnosis and grading of brain tumors. Eur J Nuclear Med Molecul Imag 32(6):653–659
5. El-Dahshan E-S, Mohsen HM, Revett K, Salem A-B (2014) Computer-aided diagnosis of human brain tumor through MRI: a survey and a new algorithm. Expert Syst Appl 41(11):5526–5545
6. Gordillo N, Montseny E, Sobrevilla P (2013) State of the art survey on MRI brain tumor segmentation. Magn Reson Imaging 31(8):1426–1438
7. Jayadevappa D, Srinivas Kumar S, Murty DS (2011) Medical image segmentation algorithms using deformable models: a review. IETE Tech Rev 28(3):248–255 (2011)
8. Yazdani S, Yusof R, Karimian A, Pashna M, Hematian A (2015) Image segmentation methods and applications in MRI brain images. IETE Tech Rev 32(6):413–427
9. Selvaraj H, Selvi ST, Selvathi D, Gewali L (2007) Brain MRI slices classification using least squares support vector machine." Int J Intell Comput Med Sci Image Process 1(1):21–33
10. John P (2012) Brain tumor classification using wavelet and texture based neural network. Int J Sci Eng Res 3(10):1–7
11. Ullah Z, Farooq MU, Lee SH, An D (2020) A hybrid image enhancement based brain MRI images classification technique. Med Hypotheses 143:109922
12. Bosch A, Munoz X, Oliver A, Marti J (2006) Modeling and classifying breast tissue density in mammograms. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06), vol 2. IEEE, pp 1552–1558
13. Avni U, Greenspan H, Konen E, Sharon M, Goldberger J (2010) X-ray categorization and retrieval on the organ and pathology level, using patch-based visual words. IEEE Trans Med Imaging 30(3):733–746
14. Yang W, Lu Z, Yu M, Huang M, Feng Q, Chen W (2012) Content-based retrieval of focal liver lesions using bag-of-visual-words representations of single-and multiphase contrast-enhanced CT images. J Digit Imaging 25(6):708–719
15. Cheng J, Yang W, Huang M, Huang W, Jiang J, Zhou Y, Yang R et al.Retrieval of brain tumors by adaptive spatial pooling and fisher vector representation. PloS one 11(6):e0157112
16. Havaei M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y, Pal C, Jodoin P-M, Larochelle H (2017) Brain tumor segmentation with deep neural networks. Med Image Anal 35:18–31

17. Prastawa M, Bullitt E, Moon N, Van Leemput K, Gerig G (2003) Automatic brain tumor segmentation by subject specific modification of atlas priors1. Academ Radiol 10(12):1341–1348

18. Ateeq T, Majeed MN, Anwar SM, Maqsood M, Rehman ZU, Lee JW, Muhammad K, Wang S, Baik SW, Mehmood I (2018) Ensemble-classifiers-assisted detection of cerebral microbleeds in brain MRI. Comput Electr Eng 69:768–781

19. Kharrat A, Gasmi K, Messaoud MB, Benamrane N, Abid M (2010) A hybrid approach for automatic classification of brain MRI using genetic algorithm and support vector machine." Leonardo J Sci 17(1):71–82

20. Papageorgiou EI, Spyridonos PP, Glotsos DT, Stylios CD, Ravazoula P, Nikiforidis GN, Groumpos PP (2008) Brain tumor characterization using the soft computing technique of fuzzy cognitive maps." Appl Soft Comput 8(1):820–828

21. Kang J, Ullah Z, Gwak J (2021) MRI-based brain tumor classification using ensemble of deep features and machine learning classifiers. Sensors 21(6):2222

22. Deepak S, Ameer PM (2019) Brain tumor classification using deep CNN features via transfer learning. Comput Biol Med 111:103345

23. Çinar A, Yildirim M (2020) Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture. Med Hypotheses 139:109684

24. Saxena P, Maheshwari A, Tayal S, Maheshwari S (2021) Predictive modeling of brain tumor: a deep learning approach. In: Innovations in computational intelligence and computer vision. Springer, Singapore, pp 275–285

25. Paul JS, Plassard AJ, Landman BA, Fabbri D (2017) Deep learning for brain tumor classification. In: Medical imaging 2017: biomedical applications in molecular, structural, and functional imaging, vol 10137. International Society for Optics and Photonics, p 1013710

26. Hemanth DJ, Anitha J, Naaji A, Geman O, Popescu DE (2018) A modified deep convolutional neural network for abnormal brain image classification. IEEE Access 7:4275–4283

27. Raja PS (2020) Brain tumor classification using a hybrid deep autoencoder with Bayesian fuzzy clustering-based segmentation approach." Biocybern Biomed Eng 40(1):440–453

28. Kaplan K, Kaya Y, Kuncan M, Ertunç HM (2020) Brain tumor classification using modified local binary patterns (LBP) feature extraction methods. Med Hypotheses 139:109696

29. Kaur G, Oberoi A (2020) Novel approach for brain tumor detection based on Naïve Bayes classification. In: Data management, analytics and innovation. Springer, Singapore, pp 451–462

30. Polat Ö, Güngen C (2021) Classification of brain tumors from MR images using deep transfer learning. J Supercomput 1–17

31. Kleesiek J, Urban G, Hubert A, Schwarz D, Maier-Hein K, Bendszus M, Biller A (2016) Deep MRI brain extraction: a 3D convolutional neural network for skull stripping. Neuroimage 129:460–469

32. Arunachalam M, Royappan Savarimuthu S (2017) An efficient and automatic glioblastoma brain tumor detection using shift-invariant shearlet transform and neural networks. Int J Imag Syst Technol 27(3):216–226

33. Minz A, Mahobiya C (2017) MR image classification using adaboost for brain tumor type. In: 2017 IEEE 7th international advance computing conference (IACC). IEEE, pp 701–705

34. Anitha R, Siva Sundhara Raja D (2018) Development of computer-aided approach for brain tumor detection using random forest classifier. Int J Imag Syst Technol 28(1):48–53

35. Rajan PG, Sundar C (2019) Brain tumor detection and segmentation by intensity adjustment. J Med Syst 43(8):1–13

36. Gumaei A, Hassan MM, Hassan MR, Alelaiwi A, Fortino G (2019) A hybrid feature extraction method with regularized extreme learning machine for brain tumor classification. IEEE Access 7:36266–36273

37. Khan HA, Jue W, Mushtaq M, Mushtaq MU (2020) Brain tumor classification in MRI image using convolutional neural network. Math Biosci Eng 17:6203

38. Ghosal P, Nandanwar L, Kanchan S, Bhadra A, Chakraborty J, Nandi D (2019) Brain tumor classification using ResNet-101 based squeeze and excitation deep neural network. In: 2019 second international conference on advanced computational and communication paradigms (ICACCP). IEEE, pp 1–6

39. Zhou Y, Li Z, Zhu H, Chen C, Gao M, Xu K, Xu J (2018) Holistic brain tumor screening and classification based on densenet and recurrent neural network. In: International MICCAI brainlesion workshop. Springer, Cham, pp 208–217
40. Saba T, Mohamed AS, El-Affendi M, Amin J, Sharif M (2020) Brain tumor detection using fusion of hand crafted and deep learning features. Cognit Syst Res 59:221–230
41. Ezhilarasi R, Varalakshmi P (2018) Tumor detection in the brain using faster R-CNN. In: 2018 2nd international conference on I-SMAC (IoT in social, mobile, analytics and cloud)(I-SMAC) I-SMAC (IoT in social, mobile, analytics and cloud)(I-SMAC). IEEE, pp 388–392
42. Soumik MF, Hossain MA. (2020) Brain tumor classification with inception network based deep learning model using transfer learning. In: 2020 IEEE region 10 symposium (TENSYMP). IEEE, pp 1018–1021
43. Lu S-Y, Wang S-H, Zhang Y-D (2020) A classification method for brain MRI via MobileNet and feedforward network with random weights. Pattern Recogn Lett 140:252–260
44. Abd-Ellah MK, Awad AI, Khalaf AA, Hamed HF (2019) A review on brain tumor diagnosis from MRI images: practical implications, key achievements, and lessons learned. Magn Reson Imaging 61:300–318
45. Amin J, Sharif M, Yasmin M, Fernandes SL (2020) A distinctive approach in brain tumor detection and classification using MRI. Pattern Recognit Lett 139:118–127
46. Ranjbarzadeh R, Bagherian Kasgari A, Jafarzadeh Ghoushchi S, Anari S, Naseri M, Bendechache M (2021) Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images. Sci Rep 11(1):1–17
47. Sheeba L, Mitra A, Chaudhuri S, Sarkar SD, Detection of exact location of brain tumor from MRI data using big data analytics
48. Amin J, Sharif M, Yasmin M, Fernandes SL (2018) Big data analysis for brain tumor detection: deep convolutional neural networks. Futur Gener Comput Syst 87:290–297
49. Amin J, Sharif M, Yasmin M, Fernandes SL (2018) Big data analysis for brain tumor detection: deep convolutional neural networks. Fut Gener Comput Syst 87:290–297
50. Jaiswal J et al. (2016) Spectrum of primary intracranial tumors at a tertiary care neurological institute: a hospital-based brain tumor registry. Neurol India 64(3):494
51. Sandhiya B, Priyatharshini R, Ramya B, Monish S, Raja GR (2021) Reconstruction, identification and classification of brain tumor using gan and faster regional-CNN. In: 2021 3rd international conference on signal processing and communication (ICPSC). IEEE, pp 238–242
52. Kausalya K, Chitrakala S (2012) Idle object detection in video for banking ATM applications. Res J Appl Sci Eng Technol 4(24):5350–5356

# Athlete Action Recognition in Sports Video: A Survey

**K. Kausalya** and **S. Kanaga Suba Raja**

## 1 Introduction

The sports video analysis concentrates on the team's and individual performance of the player. With the help of a human vision system, the actions of the players are easily recognized and observed. But monitoring a player's action using human labor is highly expensive and also there is a chance of misrecognition of the action. So, there is a need for some automated way of recognizing the actions. The learning algorithms could build a machine that can apparently realize the action which will be needful for sports video action recognition. The learning algorithms like machine learning and deep learning are used to easily predict and recognize the data. This automatic system not only helps in improving the player's performance but also helps the coach to train the players: it is also able to analyze the game rules, like the movements of the players and team performances.

Considering the advantages of learning algorithms, this paper attempts to focus the survey on sports video action recognition. It is surveyed under two categories mainly: (i.e.) athlete tracking as well as sports video action recognition. First in athlete tracking the player's identity is to be predicted in [48]; the author differentiates the jersey color of the players using MAP detection and later separates the numbers and then with the help of a template matching the players' identities are found out. In [49] the author pro- poses a novel model that automatically identifies and tracks the players. In order to overcome issues like occlusions, players' ambiguous appearance and motion patterns outside the field, Chun-Wei lu proposed a novel framework that

K. Kausalya (✉)
Department of Information Technology, Easwari Engineering College, Chennai, India
e-mail: kausalyamurthy@gmail.com

S. Kanaga Suba Raja
Easwari Engineering College, Chennai, India

can achieve 82% and 79% for tracking algorithm and player identification. In [51] the author proposes a model to identify if a player is idle for some period of time without any action.

In sports video action recognition, the player's actions are recognized based on the trained datasets or on learning algorithms. In a few datasets like the BEAVOLL dataset [6], UCF Sports [10], Olympic [50], Hockey dataset [15], THETIS [16, 30], and HMDB [16, 31] are mainly used to detect the players' actions. In the above dataset, some of the actions like free hit, serve, dig, non-action, pass, spike, block, save, walk and run can be achieved. And learning algorithms like Convolution Neural Network (CNN) along with Recurrent Neural Network (RNN) be implemented to recognize the actions in sports videos. Each of the above phases is important and the purposes vary based on particular monitoring. All the above listed issues are to be solved that enhance the performance optimization problems using current emerging technologies. Figure 1 represented the work flow of the sports video action recognition. Where, in initial stage the input video is framed then its feature is extracted followed by detecting the objects. In Final stage, the motion tracking followed by action classification is done. In this flow, the action recognition of sports video is done generally.

The organization of the paper has seven sections: Sect. 2 of the paper represents the related works that include athlete tracking, sports video action recognition and datasets based on existing studies. Section 3 describes the analysis of the techniques.
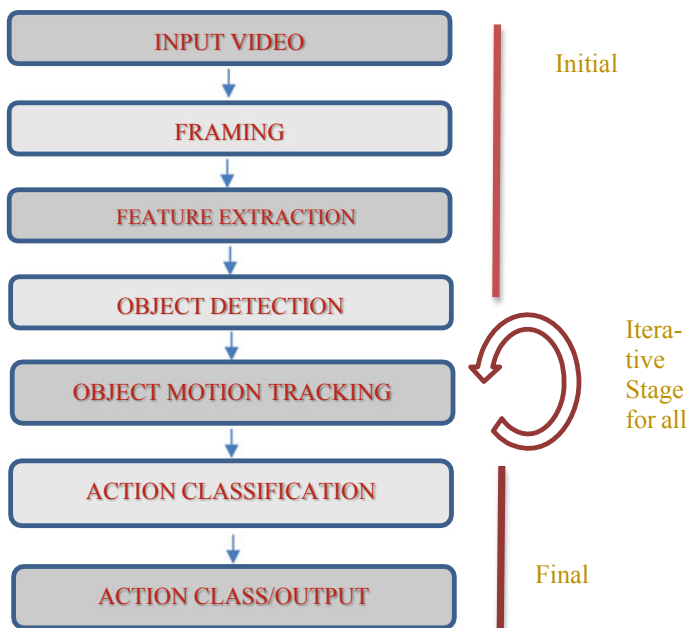


**Fig. 1** Work flow of the sports video action recognition

Section 4 determines the inference made from the papers. Section 5 discusses the real world applications. Section 6 explains about the future direction of the proposed research with Sect. 7 concludes the paper.

## 2 Literature Review

Researchers in the area of pattern recognition and computer vision had a great interest in action recognition. When it comes to sports video processing, it is considered to be one of the most interesting and challenging topics. Frequent occlusion, background clutter, out of view, fast motion and motion blur are some of the action recognition issues in sports video analysis. The problems are investigated and categorized by other works. This section includes the studies based on athlete tracking followed by the action recognition in sports videos and concludes with the list of datasets that are used for training and testing purposes in sports videos.

### 2.1 Athlete Tracking

Athlete tracking is standard practice for monitoring the player's performance, behavior and also to reduce injury risk. It would be helpful for coaches to train the players by identifying their faults and train them accordingly [6]. Athlete tracking and action recognition are considered to be the two major problems that are associated with each other. In existing studies, they have been considered separately. Longteng Kong et al., proposed a system that solves the existing problem. The new dataset BeaVoll is released for analyzing the sports video. Here, both athlete tracking and action recognition are developed and work as a joint framework. A Scaling and occlusion robust tracker (SORT) based on Compressive Tracking (CT) is proposed for the tracking module. Long-term Recurrent Region-guided Convolutional Network takes the tracking results as input for action recognition.

Features of different sizes are extracted by the assigned SPP-net along with the Long and short-term memory network that is been adapted to model action dynamics. Figure 2 shows the screenshots of the BeaVoll datasets tracking results that go through heavy occlusions, scale variations, and severe deformations. In [13] the author Mingwei Sheng et al., proposed a method for tracking the motion of the athlete–Distractor-aware SiamRPN (DaSiamRPN) network which is also used to track the result of the heavy rely..? Positions of the objects. And classification of the key frames is done using the Haar feature-based cascade. Finally, by the combination of HOG and a linear SVM the athlete of the sports video is detected. Athlete tracking is most important in recent studies to solve many more problems in sports and also to protect the future of sports.
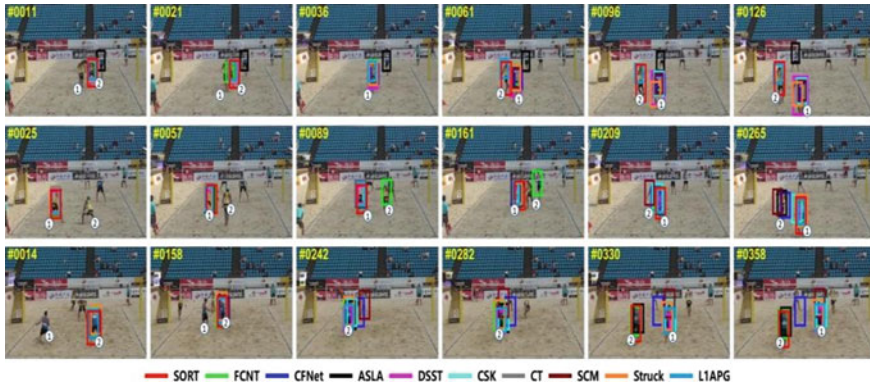
**Fig. 2** BeaVoll dataset which goes through heavy occlusions, large scaling, and deformations [6]

In many sports like soccer, it would be difficult to identify players having the same clothes of the same colors for indexing and retrieving the applications. So, the proposed study describes the player identity in sports video with suitable algorithms. The most common problem in the existing methods is not considering player identity specifically. In [5] taking that into consideration the author proposed three parts of the framework that consists of the DeepPlayer model which extracts the features of the number and partial feature embedding. Secondly, 3D localization with the ID of the players uses an Individual Probability Occupancy Map (IPOM) model, and lastly, a proposed player ID links the nodes in the flow graph based upon the K-Shortest Path with an ID can be said to be as (KSP-ID) model. The architecture of the DeepPlayer model is shown in Fig. 3. The distinguished identity helps in improving the performance tracking. In [7] Long- teng Kong et al., proposed a novel approach to solve the hierarchical deep association of the same identity detection issues in sports videos. The encoded powerful deep features are used to employ the detection association. And, the new deep architecture of Siamese Tracklet Affinity Networks (STAN) presented for the tracklet association helps to model the long-term dependencies between the athletes. Finally, the minimum-cost network flow algorithm is used to solve the hierarchical association. Many coaches think that players' identity tracking would help to track both technical performance and health related performance.

## 2.2 Sports Video Action Recognition

This section describes the analysis of sports videos with various learning techniques that overcomes the issues like frequent occlusion, background clutter, out of view, motion blur and Inter-class similarity. According to the author [1] many methodologies predicting video sequences either omit or could not use the temporal information

**Fig. 3** The architecture of the DeepPlayer model [5]

for action recognition and also to reduce the high computational cost of human action recognition. To overcome this issue the author proposed a preprocessing phase with the combination of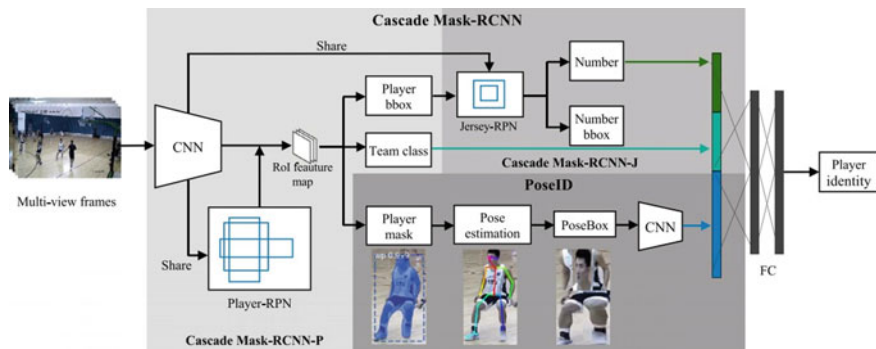 background subtraction, HOG, and Skeletal models that are used to analyze and illustrate the suitable frames. Later, the grouping of the CNN [23] and LSTM [21] networks was implemented for the feature selection process. At last human activities are labeled using the Softmax KNN classification method. The above methods are evaluated based on the UCF101 dataset [39]. In [2] the author proposed a work to overcome the satisfied accuracy problems in real-time online data stream processing. The author implements an efficient and optimized process from the surveillance environment of visual sensors based on CNN. Initially, the frame-level features are extracted using the pre-trained CNN model. Then the system is followed by the optimized deep autoencoder (DAE) method to study the temporal changes of actions. Finally, the SVM classifies the human action and the fine-tuning procedure is added to the testing phase, using accumulated data. It is believed that the planned method is appropriate for action recognition in surveillance data streams [52]. The Model Proposes a method for feature extraction using RCNN [3]. In this work, the author proposes a method for extracting the features and analysis of video content. Gaussian mixture model along with Kalman filter used to track the motion and other features are extracted by means of RNN [24] with the gated recurrent unit. This novel approach takes advantage of analyzing and extracting all features every time and in every frame of the video. Here [4], the several inclusive pre-trained models undergo hybrid approaches that can be achieved by Meta heuristic and genetic algorithms. This algorithm's major work is to merge the features obtained. Later, merging the features of the hybrid model is first compared with each model and then with all the scenarios. Then, the state of the art studies is compared to experiments with the performance optimization results. From the above study it is clear that CNN and LSTM play a major role in human action recognition from surveillance videos.

In [8], the author proposed the motion expression ability to improve, based on the spatiotemporal features that were first designed for the important key frame algorithm of motion videos.

Next, only the player is tracked and the interference fields in the video are omitted as referred to in Fig. 4. Finally, the behaviors of the players are classified. The classification of the video keyframe feature sequences is done with the help of CNN and RNN frameworks. Next, to overcome the challenges like low efficiency and high error rate in basketball action recognition, [9] the author proposed a work going on basketball shooting action that depends on feature extraction in addition to machine learning. This method elaborates the study of image feature extraction and Gaussian hidden variables that are used to recognize the basketball shooting gesture. This paper [10] uses the visual attention mechanism as well as two-stream attention based LSTM network. Furthermore, taking into consideration the correlation among two deep feature streams, a deep feature correlation layer is planned to regulate the deep learning system parameter. In [11] the main aim of the author is to build an automated solution to evaluate the handball player's motions. Here, the new dataset named RGB-D dataset is used. The player actions are filmed to learn who performed similar actions, and depth data and skeletons are sensed using the Kinect V2 sensors. The main actions of the players are examined using skeleton data simulation and they use dynamic time warping techniques to balance with the action among two players. This enables the coach to identify the player's performance and train the players accordingly.

In [12] taking action recognition in basketball as a challenge, the authors Zhiguo Pan and Chao Li proposed a novel model to recognize human action in basketball. The dataset of the basketball sports video is taken from YouTube and manually labeled. The feature from the motion regions is calculated by the Gaussian mixture model (GMM) followed by the gradient histogram being implemented to represent the shape descriptor. Finally, the motion and shape descriptor are combined and using the KNN algorithm the action recognition of basketball is determined [14]. In this model the author used a LSTM network to symbolize an individual person's action dynamic in sequence and another model aggregates the person-level details. Both the collective activity dataset and the new volleyball dataset are evaluated. In
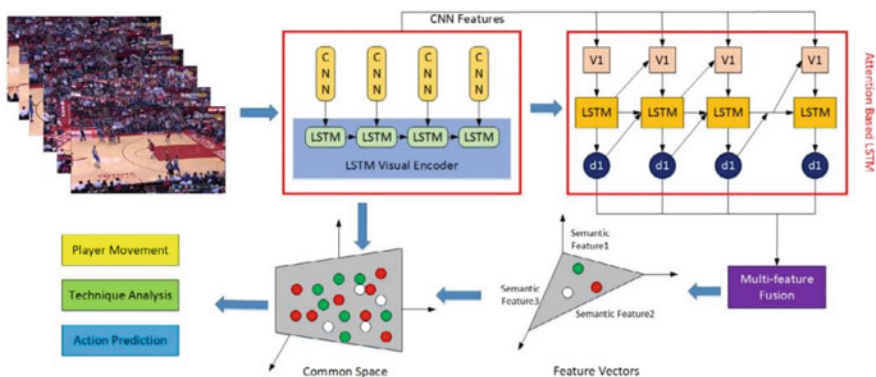


**Fig. 4** The encoder-decoder based video motion recognition framework [8]

[15] the author uses his own hockey dataset to recognize hockey activity. Free hit, goal, penalty corner and long corner are the four main activities that are included in the hockey dataset. The LSTM [22] model is used to extract the spatial feature and deep learning model VGG-16 a pre-trained model is implemented for hockey activity recognition. In paper [16], Silvia Vinyes Mora and William J. Knottenbelt, proposed a method for action recognition in a tennis sports video deep learning approach. Here, the three layered long and short-term memory (LSTM) model is presented to work on the classified fine-grained tennis actions. There are not many studies based on badminton sports video analysis; so in [17] Nur Azmina Rahmad et al., decided to propose a method based on badminton sports. Here, the broadcast video of badminton is taken and Faster Region Neural Network is implemented to follow the positions of the players.

The author in [18] introduced a new dataset for a soccer game that consists of 222 broadcast videos. All the 222 videos cover the major annotations of soccer events, which are shoot annotation, event annotation and story annotation. Deep learning techniques such as Long-term short memory model and VGG model are implemented to extract the feature and detect the event of the soccer game. In [19] Gun Junjun, the dataset contains footnotes of detailed actions in a video. The data are classified using the adaptive multi-label classification methods. The main aim of the proposed work is to track the common interest and links with the group of individuals in the model. For this purpose, an FPGA network is used that identifies and extracts the features. The Ref. [20] proposed a work with the combination of convolutional neural network along with recurrent neural network works in the classification part. Then, the extracted features of CNN are combined with the RNN temporal information. Finally, the VGG- 16 model is applied with transfer learning to achieve performance accuracy classification. The above study explains various techniques used to recognize and track the in- formation in sports videos. Majorly in [8, 10, 14–16, 18, 20] references, CNN [22] and RNN [24] frameworks and LSTM [21, 25] methods are used for the recognition and tracking process.

## 2.3 Dataset

This section explains the datasets used in existing studies and also some of the popular datasets that are used in sports videos. A detailed list is shown in Table 1. These datasets differ in camera motion, number of actions, blur background, etc. and are used for the evaluation of the assorted algorithm.

APIDIS dataset [5, 27], and STU dataset [5] are based on basketball sport in Fig. 5. It is known as a publicly available dataset; here, the data are recorded using 7 cameras with 25 fps in 800*600 resolutions. It includes 1500 frames with 16 unlabeled periods of data additionally. STU dataset is a novel dataset composed from Shantou University. The videos are recorded with 8 cameras at 24 fps in 1280*720 resolutions. This dataset contains 11 periods where only 2 periods are evaluated in

**Table 1** List of available datasets used for sports video analysis

| References | Datasets | Year | Source | Sports |
|---|---|---|---|---|
| [5] | STU | 2020 | Stantou university | Basket ball |
| [6] | BEAVOLL | 2019 | General administration tradition of sports in china | Beach volley ball |
| [7] | VolleyTrack | 2020 | YouTube | Volley ball |
| [10] | UCF11 | 2020 | Open Source | All |
| [10] | UCFSports | 2020 | Open Source | All |
| [10] | jHMDB | 2020 | Open Source | All |
| [14] | Volleyball | 2016 | YouTube | Volley ball |
| [15] | Hockey dataset | 2020 | YouTube | Hockey |
| [27] | APIDIS | 2009 | Open Source | Basket ball |
| [28] | NCAA basketball | 2009 | YouTube | Basket ball |
| [29] | H3DD | 2019 | Open Source | Hockey |
| [30] | Thetis | 2013 | Open Source | Tennis |
| [31] | HMDB | 2011 | Open Source | Tennis |
| [32] | Soccer-8 k | 2019 | YouTube | Soccer |
| [33] | Soccer dataset | 2009 | Surveillance | Soccer |
| [34] | Soccer dataset | 2019 | Surveillance | Soccer |
| [35] | SoccerNet | 2018 | Open Source | Soccer |
| [36] | RobocupSimData | 2017 | Open Source | RoboCupSoccer |
| [37] | SoccerNet-V2 | 2021 | Open Source | Soccer |
| [38] | GolfDB | 2019 | Open Source | Golf |

the proposed work. Both APIDIS and STU datasets are mainly used for the player's identity in [5].

BEAVOLL dataset [6] is a new benchmark dataset (in Fig. 6) and is based on beach volleyball games which are larger and very difficult datasets; it provides action labels for determining players' tracking and action recognition methods. It consists of 30 video clips that last from 80–120 per second. The videos are captured with 1440*1080 resolutions. To confirm action recognition, the dataset was created based on the 9 typical actions, like serve, dig, non-action, pass, spike, block, save, walk and run.

NCAA Basketball [28], VolleyTrack [7]: NCAA Basketball datasets are created based on the YouTube videos for team action recognition. The sequence consists of 1179 frames with 30fps in 640*480 resolutions. Since it is small and not enough to train the network, the APIDIS dataset [27] is combined with the NCAA basketball dataset for testing and training in the proposed model. VolleyTrack is a newly collected dataset that contains 18 video clips collected from YouTube. Each video lasts for about 8–12 s that include 5406 frames at 30fps with 1920*2080 resolutions. Both the datasets are shown in Fig. 7.

**Fig. 5** Sample APIDIS dataset [7]



**Fig. 6** Sample BEAVOLL dataset [6]

UCF11, UCFSports, jHMDB [10] UCF11 dataset contain 1600 videos with 11 different actions. Due to large variations of illumination, it is known as a challenging dataset [26]. UCFSports dataset is collected from broadcast television channels. It has 150 sequences with 720*480 resolutions are included in this dataset. jHMDB is a larger dataset that contains 923 videos with 21 various actions.

H3DD [29] dataset is an open source. The dataset contains handball activities in 3d video sequences. For every individual player RGB frame, depth frame and skeleton frame are noticed. According to the resolution size, the frames are categorized; the RGB data has the frame size of 1920*1980, the depth data has the frame size of 514*424 and a.txt file has skeleton data with 3D coordinates for 25 joints.

Hockey dataset [15] is a new dataset that was gathered from YouTube that contains the videos of Hockey World Cup 2018 and International Hockey Federation. The resolution of the video is 1280*720 size. The hockey dataset includes actions like

**Fig. 7** Sample NCAA and VolleyTrack Datasets [7]

free hit, long corner and penalty corner of video frames, which are collected from 12 broadcasted hockey matches.

THETIS [16, 30], HMDB [16, 31]: THETIS dataset is created with 1980 tennis videos based on 12 actions. Each action is performed several times with 31 amateurs and 24 experienced players. This dataset is considered to be a longer dataset. HMDB dataset is used for action recognition. This dataset contains 6849 videos with various 51 actions that include facial actions along with body movements.

Soccer-8 k Dataset [32, 33] there is no standard dataset available for soccer games. Here, with the help of an event detector and based on the full HD videos of Laliga and a few Champions League the soccer -8 k dataset is created, and shown in Fig. 8. All the above datasets are used in the various applications of sports video analysis for both athlete tracking as well as action recognition; each dataset helps in a unique way to recognize actions in the sports video analysis.
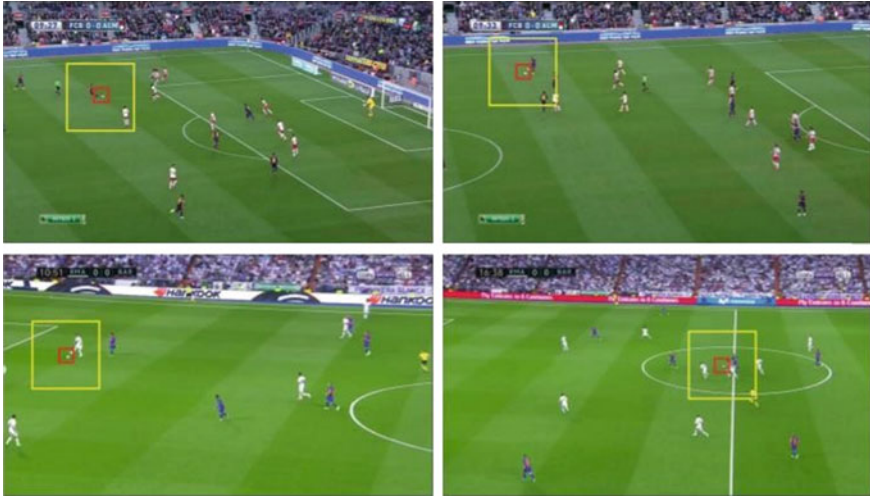
**Fig. 8** Sample Soccer-8 k Dataset with correct event detection [32]

## 2.4 Advantages and Disadvantages of Action Recognition in Sports Video

From the above literature reviews the merits and the demerits of sports video action recognition are listed below,

**Advantages**

- Visualizing the function and understanding the representation is simple and easy for analysis
- The skeleton features from the data are more important.
- The model of the features that are used for training the data is openly known.
- Through the machine learning algorithms, the features are robotically learned.
- The deep learning algorithms are more adoptable for solving complex tasks.

**Disadvantages**

- Due to excessive dimensions, they may cause computational intensive.
- Human Detection in occlusion.
- Difficulty in updating the background information at regular intervals.
- Accurate identification of human motion is not recognized.
- Fixed-angle camera.
- Model needs to be trained for recognizing sub-types of action.
- Long-term dependencies between the frames.

# 3   Comparative Analysis

The below table comparatively summarizes about the deep learning algorithms used for action recognition in sports videos.

# 4   Result and Discussion

Analysis of sports performance is the investigation of real sports video performance. In [8] the movement recognition and movement prediction with its movement type results are presented in Table 2 as well as Table 3. The accuracy of a valid set and testing set of movement type pass/foul is lower for both recognition and prediction results. The proposed work recognition and prediction accuracy is compared with the existing method recognition and prediction accuracy of the datasets and shown in Fig. 9. The proposed work uses a larger dataset that gives high accuracy and identifies the players efficiently (Table 4).

In [10] the confusion matrix is used to make clear differences between the different datasets used in the proposed work. The above Fig. 10a and b shows the confusion matrix of the UCF11 dataset and UCF Sports datasets. The confusion matrix helps to identify the classes with the highest accuracy and lowest accuracy. In Fig. 10a class s-juggling has the lowest accuracy of 90.3% and class divine has the highest

**Table 2**  Comparative table of deep learning method

| References | Year | Algorithm | Dataset | Accuracy (%) |
|---|---|---|---|---|
| Kong et al. [6] | 2019 | Long-term Recurrent Region-guided Convolutional | BeaVoll | 73.2 |
| Chen and Wang [8] | 2020 | CNN and RNN | NBA | 76.5 |
| Dai et al. [10] | 2020 | LSTM | UCF11 UCF Sports jHMDB | 96.9 98.6 76.3 |
| Rangasamy et al. [15] | 2020 | Pre-trained VGG-16 model | Hockey | 98 |
| Vinyes Mora and Knottenbelt [16] | 2017 | Three layered LSTM | THETIS | 88.16 |

**Table 3**  Movement recognition result

| Movement recognition result on the basketball player motion recognition dataset | | | | |
|---|---|---|---|---|
| Movement type | Shoot (%) | Pass (%) | Dribble (%) | Foul (%) |
| Accuracy/valid set | 84 | 75 | 90 | 80 |
| Accuracy/testing set | 82 | 70 | 85 | 74 |
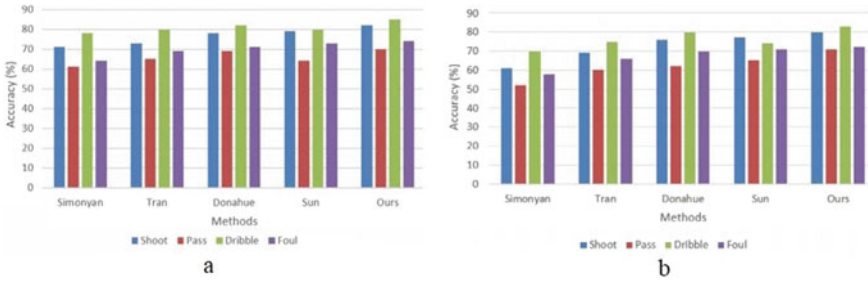
**Fig. 9**  **a** Represents the recognition accuracy for the test set and **b** Represents the prediction accuracy for the test set [8]

**Table 4**  Movement prediction result

| Movement prediction result on the basketball player motion recognition dataset | | | | |
|---|---|---|---|---|
| Movement type | Shoot (%) | Pass (%) | Dribble (%) | Foul (%) |
| Accuracy/valid set | 83 | 73 | 87 | 77 |
| Accuracy/testing set | 80 | 71 | 83 | 72 |

ac- curacy of 99.8%. In Fig. 10b the accuracy is distributed uniformly and also all the accuracy classes have higher than 95% of accuracy, except the swing side class action. Figure 11 shows the confusion matrix of the proposed work of the jHMDB dataset. Here, some actions have higher accuracy and some have lower accuracy comparatively.

In [11] to calculate the potential level of players the set of performances had been made between the experts and beginners. To analyze their performance the author used the skeleton method of 3D motion. In [13] the proposed work uses testing datasets that include two video sequences where the video sequence I with 510 frames is the men's 100 m race video at Rio 2016 and the video sequence II with 380 frames is the London 2012 Olympic Games. Both the video sequences are



(a) Confusion matrix of our proposal for UCF11 dataset     (b) Confusion matrix of our proposal for UCF Sports dataset
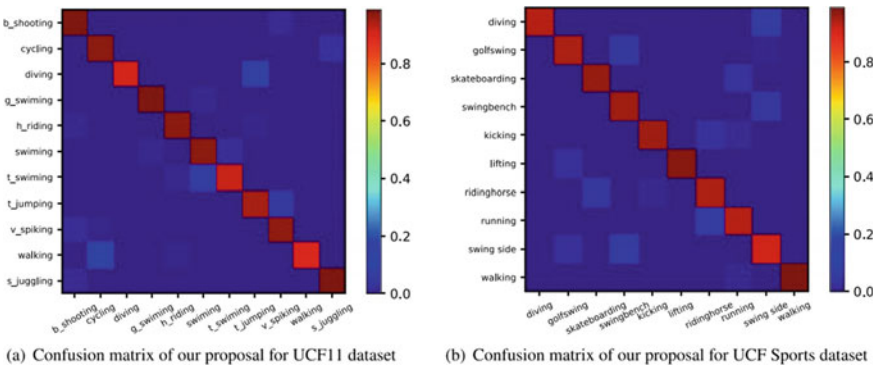
**Fig. 10**  Confusion matrix of UCF11 and UCF Sports datasets [10]
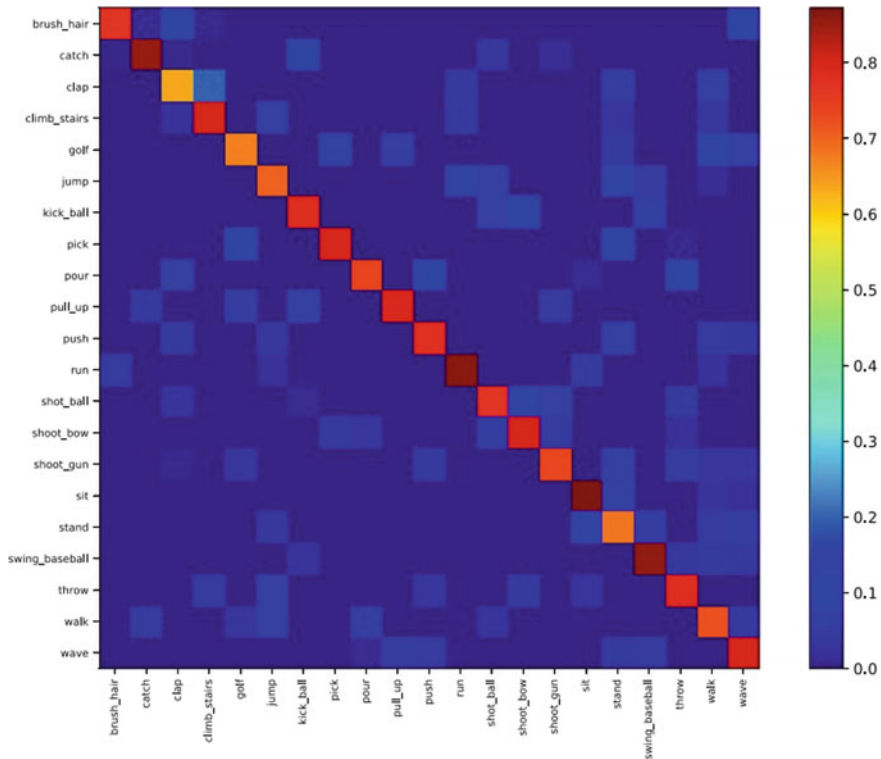
**Fig. 11** Confusion matrix of jHMDB dataset [10]

combined together as 890 frames and proceeds for implementation. From the above section, it is decided that the pre-trained model gives better accuracy compared with other algorithms.

## 5 Inference Made

- An additional reference frame is included for occluded regions in a frame for accurate human detection. Occlusion can be reduced by the selection of camera positions.
- Background Subtraction of the sports video are compared and analyzed in dynamic scenes.
- Creating bounding boxes and identifying everything within each box is to improve the accuracy and precision of object detection and tracking in a video.
- Models need to be trained for recognizing subtypes of action.
- Multi-camera tracking systems were essential for tracking all the players.
- Lack of Short tracklets (frames).

## 6 Suggestion Proposed

The association of data analytics and sensors in sports video analysis provides needful information for players. Sensors are used to generate data that is essential in the field of the player's health, detects the player's posture, and fan engagement, by using sensor data transforming it with the real streaming analysis and then joining the data with artificial intelligence. This creates an emerging application developed in various fields. In future, predictive intelligence will help the sports team to improve the game strategies by coaching the players of the game based on athlete tracking. It is also used for extracting and analyzing players and team statistics. Using novel deep learning algorithms the system could help the teams and coaches improve the accuracy. And by training, the system will track the action of the players and their mistakes. Also based on the tracking details, the players are coached and monitored for better performance in future. The emerging techniques of deep learning can also be focused on predicting or analyzing the game results based on the audience's facial reaction. Later, based on the novel computer vision application the shots of the ball are predicted, as to whether it is in/out of boundary lines. In many existing works, the pre-recorded data streaming video is used; instead, the live video streaming data can be used for sports video action recognition. The training duration of the system could be reduced for performing with better accuracy.

## 7 Conclusion

This paper focuses on the athlete action recognition of sports videos; so it highlights some of the common issues in action recognition of sports, like athlete tracking and player identity detection. And also reviews its commercial applications that include video abstraction, performance analysis and augmented reality. A detailed survey of the emerging system of sports video action recognition and athlete tracking is highlighted. Along with it the availability of benchmark datasets is studied, that is very significant. Here, some datasets are globally available and their results are evaluated. But some evaluations are based on their own dataset which is not available globally.

And also the analysis of the sports video action recognition is experimented followed by the inference made. From the above studies it is concluded that the pre-trained model gives better accuracy in action recognition of the sports videos. An apparent future direction for sports video athlete action recognition is therefore imposed on different algorithms.

# References

1. Serpush F, Rezaei M (2021) Complex human action recognition using a hierarchical feature reduction and deep learning-based method. SN Computer Science 2(2):1–15
2. Ullah A, Muhammad K, Haq IU, Baik SW (2019) Action recognitionusing optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments. Fut Gener Comput Syst 96:386–397.
3. Jaouedi N, Boujnah N, Bouhlel MS (2020)A new hybrid deep learning model for human action recognition. J King Saud Univ-Comput Infor m Scie 32(4):447–453
4. Yilmaz AA, Guzel MS, Bostanci E, Askerzade I (2020) Anovel action recognition framework based on deep-learning and genetic algorithms. IEEE Access 8:100631–100644
5. Zhang R, Wu L, Yang Y, Wu W, Chen Y, Xu M (2020) Multi-cameramulti-player tracking with deep player identification in sports video. Pat tern Recognit 102:107260
6. Kong L, Huang D, Qin J, Wang Y (2019) A joint framework for athlete tracking and action recognition in sports videos. IEEE Trans Circuits Syst Video Technol 30(2):532–548
7. Kong L, Huang Di, Wang Y (2020) Long-term action dependence-based hierarchical deep association for multi-athlete tracking in sports videos. IEEE Trans Image Process 29:7957–7969
8. Chen L, Wang W (2020) Analysis of technical features in basketball video based on deep learning algorithm. Signal Process: Image Commun 83:115786
9. Ji R (2020) Research on basketball shooting action based on image feature extraction and machine learning. IEEE Access 8:138743–138751
10. Dai C, Liu X, Lai J (2020) Human action recognition using two-stream at- tention based LSTM networks. Appl Soft Comput 86:105820
11. Elaoud A, Barhoumi W, Zagrouba E, Agrebi B (2020) Skeleton-based comparison of throwing motion for handball players. J Ambient Intell Humaniz Comput 11(1):419–431
12. Pan Z, Li C (2020) Robust basketball sports recognition by leveraging motion blockestimation. Signal Process: Image Commun 83:115784
13. Sheng M, Wang W, Qin H, Wan L, Li J, Wan W (2020) A novel changing athlete body real-time visual tracking algorithm based on distractor-aware SiamRPN and HOG-SVM. Electronics 9(2):378
14. Ibrahim MS, Muralidharan S, Deng Z, Vahdat A, Mori G (2016) A hierarchical deeptemporal model for group activity recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1971–1980
15. Rangasamy K, As'ari MA, Rahmad NA, Ghazali NF (2020) Hockey activity recognition using pre-trained deep learning model. ICT Express 6(3):170–174
16. Vinyes Mora S, Knottenbelt WJ (2017)Deep learning for domain-specific action recognition in tennis. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 114–122
17. Rahmad NA, Sufri NA, Muzamil NH, As'ari MA (2019) Badminton player detection using faster region convolutional neural network. IndonesianJ Electr Eng Comput Sci 14(3):1330-1335
18. Yu J, Lei A, Hu Y (2019) Soccer video event detection based on deep learning. In: Internationalconference on multimedia modeling. Springer, Cham, pp 377–389
19. Junjun G (2021) Basketball action recognition based on FPGA and particle image. Microprocess Microsyst 80:103334
20. Russo MA, Kurnianggoro L, Jo KH (2019) Classification of sports videos with combination of deep learning models and transfer learning. In: International conference on electrical,computer and communication engineering (ECCE). IEEE, pp 1–5
21. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
22. Baccouche M, Mamalet F, Wolf C, Garcia C, Baskurt A (2010) Action classification in soccer videos with long short-term memory recurrent neural networks. In: International con ference on artificial neural networks. Springer, Berlin, pp 154–159
23. Yamashita R, Nishio M, Do RK, Togashi K (2018) Convolutional neural networks: anoverview and application in radiology. Insights Imaging 9(4):611–629

24. Tarwani KM, Edem S (2017) Survey on recurrent neural network in natural language processing. Int J Eng Trends Technol (IJETT) 48(6)
25. Lindemann B, Müller T, Vietz H, Jazdi N, Weyrich M (2021) A survey on long short-term memory networks for time series prediction. Procedia CIRP 99:650–655
26. Liu J, Luo J, Shah M (2009) Recognizingrealistic actions from videos "in the wild". In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 1996–2003
27. De Vleeschouwer C, Delannay D (2009) Basketball dataset from the European project APIDIS
28. Dataset, https://www.kaggle.com/ncaa/ncaa-basketball
29. Dataset, https://github.com/Elaoud/H3DD-dataset
30. Gourgari S, Goudelis G, Karpouzis K, Kollias S (2013) Thetis: three dimensional tennisshots a human action dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 676–681
31. Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) HMDB: a largevideo database for human motion recognition. In: 2011 international conference on computer vision. IEEE, pp 2556–2563
32. Ganesh Y, Sri Teja A, Munnangi SK, Murthy GR (2019) A novel framework for finegrained action recognition in soccer. In: International work-conference on artificial neural networks. Springer, Cham, pp 137–150
33. D'Orazio T, Leo M, Mosca N, Spagnolo P, Mazzeo PL (2009) A semi-automatic system forground truth generation of soccer video sequences. In: 6th IEEE international conference on advanced video and signal surveillance. Genoa, Italy September, 2–4
34. Kukleva A, Khan MA, Farazi H, Behnke S (2019) Utilizingtemporal information in deep convolutional network for efficient soccer ball detection and tracking. In: Robot World Cup. Springer, Cham, pp 112–125
35. Giancola S, Amine M, Dghaily T, Ghanem B (2018) Soccernet: a scalable dataset for action spotting in soccer videos. In: Proceedings of the IEEE conference on computer visionand pattern recognition workshops, pp 1711–1721
36. Dataset: https://paperswithcode.com/dataset/robocupsimdata
37. Deliege A, Cioppa A, Giancola S, Seikavandi MJ, Dueholm JV, Nasrollahi K, Ghanem B, Moeslund TB (2021) Soccernet-v2: a dataset and benchmarks for holistic understanding of broadcast soccer videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4508–4519
38. Dataset,https://github.com/wmcnally/GolfDB
39. Soomro K, Zamir AR, Shah M (2012) UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402
40. Rui Y, Gupta A, Acero A (2000) Automatically extracting highlights for TV baseball programs. In: Proceedings of the eighth ACM international conference on Multimedia, pp 105-–115
41. Kawashiima T, Yoshino K, Aok Y (1994) Qualitative image analysis of group behavior. In: Proceedings computer vision and pattern recognition
42. Taki T, Hasegawa JI, Fukumura T (1996) Development of motion analysis system for quantitative evaluation of teamwork in soccer games. In: Proceedings of 3rd IEEE international conference on image processing, vol 3. IEEE, pp 815–818
43. Yu X, Leong HW, Xu C, Tian Q (2004) A robust and accumulator-free ellipse Houghtransform. In: Proceedings of the 12th annual ACM international conference on multimedia, pp 256–259
44. Yu X, Yan X, Hay TS, Leong HW (2004)3D reconstruction and enrichment of broadcast soccer video. In: Proceedings of the 12th annual ACM international conference on multi- media, pp 260–263
45. Bebie T, Bieri H (2000)A video-based 3D-reconstruction of soccer games. Eurographics vol 19(3)
46. Pingali G, Opalach A, Jean Y, Carlbom I (2001) Visualization of sports using motion trajectories: providing insights into performance, style, and strategy. In: Proceedings visualization,2001. VIS'01. IEEE pp. 75–544
47. Rui Y, Gupta A, Acero A (2000) Automatically extracting highlights for TV baseball programs. In: Proceedings of the eighth ACM international conference on multimedia, pp 105-115

48. Ahammed Z (2018) Basketball player identification by jersey and number recognition (Doctoral dissertation, Brac University)
49. Lu CW, Lin CY, Hsu CY, Weng MF, Kang LW, Liao HY (2013) Identificationand tracking of players in sport videos. In: Proceedings of the fifth international conference on internet multimedia computing and service, pp 113–116
50. Niebles JC, Chen CW, Fei-Fei L (2010) Modeling temporal structure of decomposablemotion segments for activity classification. In: European conference on computer vision. Springer, Berlin, pp 392–405
51. Kausalya K, Chitrakala S (2012) Idle object detection in video for banking ATM applications. Res J Appl Sci Eng Technol 4(24):5350–5356
52. Sandhiya B, Priyatharshini R, Ramya B, Monish S, Raja GR (2021)Reconstruction, identification and classification of brain tumor using gan and faster regional-CNN. In 2021 3rd international conference on signal processing and communication (ICPSC). IEEE.

# Evaluation of Sustainable Drainage Systems in the Mekong Delta: A Case Study of Vinh Long City, Vietnam

**Trong Nhan Huynh** and **Eric C. W. Lou**

## 1 Introduction

The Mekong Delta in Vietnam is severely affected by climate change rise of sea levels, inundation, saltwater intrusion, and the increase of extreme weather events. The urban areas in this region are greatly affected due to the high concentration of population and the presence of many vulnerable areas. Inundation is a prominent problem and occurs in most of the major cities around the Delta. In addition to being affected by subsidence and inland dike systems, urbanization is also a main cause of changes in the natural flow of rainwater, increasing peak runoff on the existing urban drainage system, and causing flooding. Previous studies have identified urban land use planning is not linked with drainage planning, leading to the lack of control of urban surface as the main cause of inundation [1, 2]. The availability of open access remote sensing data such as Landsat 8, MODIS and Sentinel-2 play a key role in this research. Sentinel-2 is a European wide-swath, high-resolution multi-spectral imaging mission that contributes to earth observation applications [3]. The research question—through the analysis of urban land cover from open-access satellite data, how to evaluate the sustainable drainage efficiency of the city, and how to apply this assessment in urban development planning that is suitable to Mekong Delta cities context. As we move towards the new industrial revolution, Industry 4.0 and digitalization of the landscape will provide the opportunity for us to build a digital representation of our physical landscape. Through digitalization (and verifying the digital data), various simulations could be conducted to imitate actual sustainable drainage system or systems over time and the evolution towards smart cities—for decision makers and researchers alike to make informed decisions for our future [15].

T. N. Huynh (✉)
MienTay Construction University, Vinh Long 89000, Vietnam
e-mail: huynhnhancien@mtu.edu.vn

E. C. W. Lou
Manchester Metropolitan University, Manchester, UK

## 2 Urban Flooding in Mekong Delta Cities

### 2.1 Mekong Delta Urban System

The Mekong Delta, one of the largest deltas in the world, has a dense network of rivers, canals and canals and features of settlement along rivers and canals. The Delta is densely populated, and has the largest potential for agriculture, fishing and aquaculture in the country. Agriculture contributes 34.6% of the national GDP of Vietnam, including: rice accounts for more than 50% of the area and output, contributing 90% of the country's export rice output; Aquaculture and seafood provide over 42% of catches and nearly 67% of aquaculture output, and seafood export accounts for 60% of the country's output [4]. The urban system is distributed according to the dispersion model of small and medium scale operations and the population structure is spread evenly; most are attached to rivers, canals or intersections between roads and waterways. The three-level urban system is evenly distributed in a hierarchical structure with one city being the regional and national center; 12 provincial cities and arranged equally from 60 to 80 km; the rest are district centers and commune clusters [2] (Fig. 1). In general, the existing urban system ensures the function of providing basic administrative services to the locality. However, the urban infrastructure projects have not been synchronously invested, leading to limited effectiveness. According to Royal Haskoning's assessment, most of the urban and provincial capitals of the Mekong Delta are urban embellishments, expansions, or upgrades from old town [5].

### 2.2 Mekong Delta Urban Flood Risk Context

According to the assessment by the Southern Institute of Water Resource Planning, the general feature of urban areas in the Mekong Delta is its flat terrain and low elevation compared to the water level of rivers and canals, making it difficult for water drainage [6]. When it rains, river and canals does not flood and the river water level is low, but many urban areas were still flooded. There is no or lack of drainage sewers. In the assessment by UN-Habitat, the existing rainwater drainage system of urban areas in the region is generally not sufficient for urban drainage [7]. Another serious challenge for urban areas in the region is the pressure of urbanization and changing urban surface. The process of urbanization and urban development has increased the impermeable surface, encroaching on natural canals, changing the natural water circulation. Existing drainage system must accommodate a large volume of rainwater. Along with the unreasonable spatial organization of urban planning, the concreting of urban surface has prevented rainwater from replenishing underground water source. At the same time, groundwater extraction for domestic use and agricultural production in the surrounding urban areas has contributed to the decline of groundwater levels and more serious subsidence. The final report of
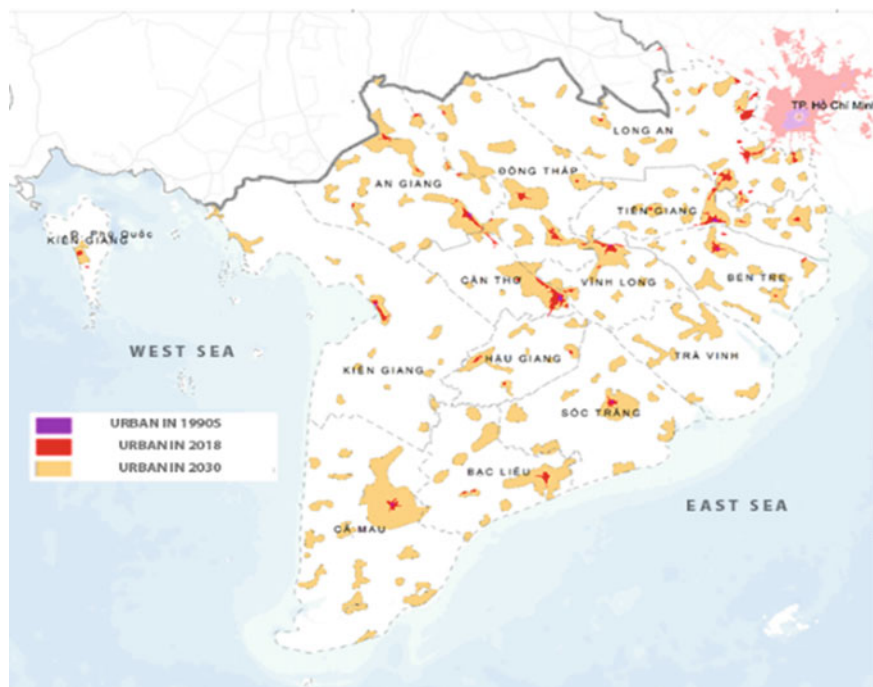
**Fig. 1** Mekong delta urban system [5]

Flood Proofing Program (FPP) from the German International Cooperation GmbH (GIZ) also stated that rapid urban development, uncontrolled housing construction in the floodplain and urban surface sealing works to prevent rainwater from naturally percolating to the surface in the Mekong Delta [8].

## 2.3 Sustainable Urban Drainage Solution for Mekong Delta Cities

In this context, Sustainable urban drainage system (SUDS) solution can be considered as the key to ensuring the restoration of the natural water circulation, and minimizing the impact of urbanization on the flooding problem in urban areas [9]. Experts from Vietnam Ministry of Construction stated that the sustainable urban drainage approach SUDS is essential to control surface water run-off, while maintaining the natural characteristics of the flow; replenishing groundwater resources due to increased permeability coefficient; contributing to urban greening, beautifying the landscape and environment through the development of regulated lakes, open canals and vegetation [8]. Legally, National Regulation QCVN 01:2021 also stipulates that the planning of surface water drainage systems must increase the surface area to absorb water for

traffic works, yards, grey infrastructure and other public areas, ensure storage volume and regulate surface water [10]. SUDS model has been piloted by the Urban Drainage and Flood Proofing Program (GIZ) in 3 cities: Long Xuyen (An Giang), Rach Gia (Kien Giang) and Ca Mau (Ca Mau) with small scale in 2017–2019 [8]. Through the preliminary assessment, the proposed solutions have contributed to solving flooding in the pilot area.

However, the method of determining the characteristics of naturally permeable surfaces in urban areas to meet the objectives of SUDS has not been specifically guided, causing difficulties in drainage planning, decisions making towards sustainable drain-age solutions and climate change response. Data management of urban surface properties (reports and management regulations) of specialized agencies is still limited. Local construction management authority data is only related to the land use function, there is no data on the properties of urban surfaces. Therefore, in order to meet the goal of "increasing the water-absorbent surface area" in the National Regulation QCVN 01:2021 become a criterion or indicator in the drainage planning and surface water drainage management plan, it is necessary to have a new approach in assessing the natural permeable surface of urban areas.

## *2.4 The Case Study of Vinh Long City*

Vinh Long city is a medium-sized provincial town in the Mekong Delta, which covers 48.1 kms and has a population of 137,870 in 2019 (Fig. 2). Classified as Grade II according to the Vietnamese urban system, Vinh Long has the same size and classification as the remaining 9 of the 11 provincial cities in Mekong Delta. Although the planning, construction, urban development and infrastructure investment in the city have developed strongly, the urban drainage system is incomplete and flooding has not been effectively controlled. In recent years, heavy rains and high tides frequently caused flooding in the urban center areas. Recently, the Vietnamese Prime Minister approved the investment policy of the project "Urban development and strengthening of climate change adaptation in Vinh Long city" funded by the World Bank and the Dutch Government [11]. One of the goals of this project is to combine the investment works with the development of a green infrastructure system, increasing the water surface area to create more bases for Vinh Long city in adapting to climate change; as well as towards a green, sustainable city. Due to the similar urban grade and rate of urbanization to other cities in Mekong Delta, the implementation of climate change adaptation project in Vinh Long can be considered as a pilot study to up-scale in other urban areas, especially for provincial capitals.

**Fig. 2**  Vinh Long city location in Mekong Delta, Vietnam

## 3   Urban Landcover Remote Sensing Analysis Theories

The application of GIS, especially remote sensing images in earth observation, is a scientific field that has developed rapidly in recent times. A recent study of extracting urban impermeable surfaces with high accuracy from Sentinel-2 multispectral images through linear spectral mixed analysis (MLSMA) extracted impervious surface maps with a resolution of up to 10 m, with an accuracy of up to 93% [12]. However, the analysis results focus on impervious surfaces and have not identified natural drain-age spaces in urban areas.

In Vietnam, a related study on using GIS to build a flow coefficient map of Bac Kan province has proposed a method to determine the flow coefficient based on surface characteristics according to a raster map of surface properties, area and slope [13]. Since then, this study has calculated more accurately the rainfall in a typical basin of Bac Kan. However, this method still requires input data maps and is difficult to regularly update the change in urban land surface. Another related study at Can Tho University applied normalized difference vegetation index analysis (NDVIs) with Landsat satellite data to determine the percentage of impervious surface of Can Tho from 1997 to 2016 [14]. Although the study analyzed the proportion of impervious surfaces, the distribution of natural drainage spaces in urban areas was not analyzed.

Related studies show the potential of urban surface analysis through GIS application, especially remote sensing data. For example, it is possible to analyze the flow coefficient by area with the raster calculation tool, but the input data source

does not meet the requirement of regular updating due to the change in the urban surface. Studies also show that NDVI analysis is an important basis for determining urban surface properties with satellite image data [12, 14]. Comparing the remote sensing image sources in the above studies, the Sentinel-2 image source provides higher resolution, which can provide more accurate analysis results in the urban center area. However, to analyze the distribution of natural permeation spaces in urban areas, the analytical method must add intermediate steps to bring the spatial statistical results of accessibility to the urban natural spaces for water infiltration. This approach enhances the possibility for urban drainage capacity comprehensive assessment.

## 4 Research Methodology

### 4.1 Research Approach

Urban context analysis in the Mekong Delta and related studies shows that the assessment of natural drainage capacity of urban areas is not only based on the ratio of the natural permeable surface, but also the distribution of natural permeable spaces in urban areas. The research approach does not only use remote sensing images to identify areas with impervious surfaces of urban areas but also applies spatial statistical analysis to assess the level of interleaving of green infrastructure systems in the urban core which have a high percentage of impervious surfaces. To achieve such analytical results, the study proposes to apply an analytical procedure as shown in Fig. 4. Combined with the digitized Area of Interest (AOI), the study can select suitable satellite image data for analysis. The NDVI and normalized difference water index (NDWI), and normalized difference built-up index (NDBI) combined analysis method was used to classify impervious surfaces, permeable plant surfaces, natural soil surfaces and water bodies. Based on field analysis with reference to the technical literature of Sentinel-2 [3], the thresholds of classification values are given in Table 1.

According to the principle of SUDS, blue and green spaces are considered as natural permeable spaces [9]. Proximity analysis with Geospatial Data Abstraction Library in Quantum GIS (QGIS) tool is used to determine the minimum distance

**Table 1** Classification threshold of urban land cover types with Sentinel-2 image source

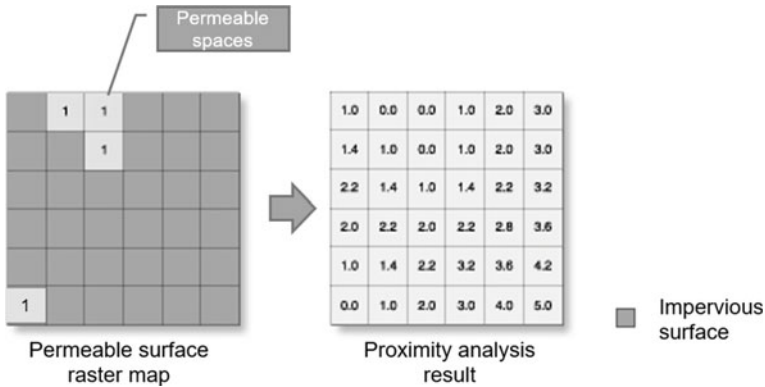| Landcover type | NDVI, NDBI value | NDWI value |
| --- | --- | --- |
| Water body | NDVI < –0.1 | >0.2 |
| Bare soil | – 0.1 < NDVI < 0.1, NDBI < 0.14 | |
| Impervious surface | NDVI < 0.1 or NDBI > 0.14 | |
| Low density vegetation (lawns, shrubs) | 0.2 < NDVI < 0.4 | |
| Plants and trees | NDVI > 0.7 | |

**Fig. 3** Proximity analysis method with Euclidean distance to the closest source

from any point in the city to these spaces (Fig. 3). Based on the raster value statistics tool of QGIS, the study can determine the statistical values of the natural spatial permeation distance such as: maximum value, mean, standard deviation and coefficient variability. The urban surface classification results help determine the ratio of these surface types within wards, communes or by urban drainage basins. On the other hand, the results of asymptotic analysis of the natural permeable space in urban areas help determine the distribution characteristics between this space and the spaces with impervious surfaces (Fig. 4). The coefficient of variation according to the statistical results of asymptotic distance is an important basis for assessing the degree of interweaving of natural permeable spaces between urbanized areas with a high percentage of impervious surfaces. From there, it is possible to assess the ability of surface water drainage in a sustainable way in urban areas. In other words, assess the porosity of urban surfaces.

## 4.2 Sentinel-2 Dataset

Sentinel-2 is an earth observation satellite developed by the European Space Agency and is part of the Copernicus program to perform monitoring, support and provide services such as: monitoring. forests, land cover change or disaster management. This system consists of two Sentinel satellites 2A and 2B with imaging systems in 13 spectral channels from the visible, near-infrared and short-wave infrared bands. With a resolution of up to 10 m and fast repetition time, Sentinel-2 will be an effective support tool in earth surface monitoring. Spatial resolutions range from 10 m (Red, Green, Blue, NIR) to 20 m (6 channels of shortwave and red-edge infrared) and 60 m (3 channels of atmospheric correction). Time resolution: 5 days (combining both satellites) and image size is 100 km $\times$ 100 km [3].
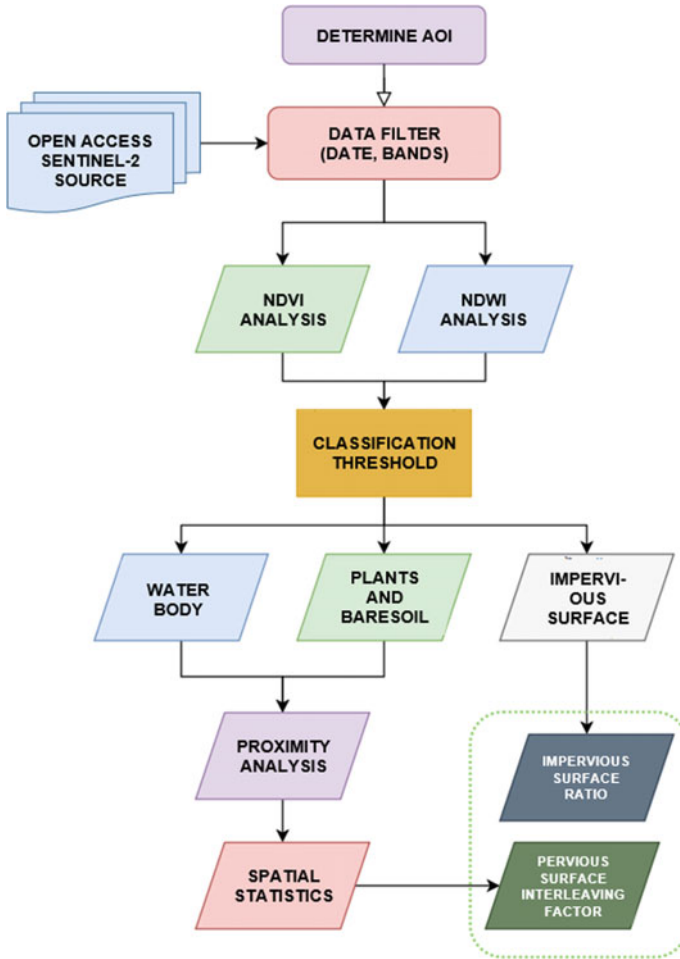
**Fig. 4** Proposed procedure to evaluate sustainable urban drainage with satellite images

In this study, Sentinel-2 data for urban surface analysis was obtained from Open Access Hub, filtered with less than 10% cloud ratio. Through preliminary filtering, the Sentinel-2 satellite image dataset was taken on March 16, 2020, with low cloud coverage to ensure accurate analysis results for the Vinh Long city area. The image data is cropped according to the boundary of Vinh Long city and converted from EPSG:4326 to the VN-2000 coordinate system prescribed for use in Vietnam. According to Sentinel Hub, the Sentinel-2 remote sensing indices NDVI, NDBI and NDWI are defined by bands B03, B04, B08 and B11 [3].

# 5 Analysis of Results

Applying the process proposed in the methodology, the results are a water cover map, vegetation cover map and impermeable surface map (Fig. 5). Based on the boundary map of the ward-level administrative unit of Vinh Long city, the study has performed spatial statistical analysis to get the results of the impermeable surface ratio of areas in Vinh Long city (Table 2).

The data from Table 2 show that there is a significant difference between the administrative units in Vinh Long city in terms of the percentage of impervious surface. However, proximity analysis of water surface and vegetation space for the remaining areas within Vinh Long city is performed as shown in Fig. 6. The spatially permeable spatial approach analysis map shows a significant difference in this value in Vinh Long city. The deep red areas have a natural permeable space approach distance of nearly 280 m. These areas are concentrated in existing urban areas such as Ward 1, Ward 2 and Ward 4, Ward 8. The results of statistical analysis in Fig. 7 also
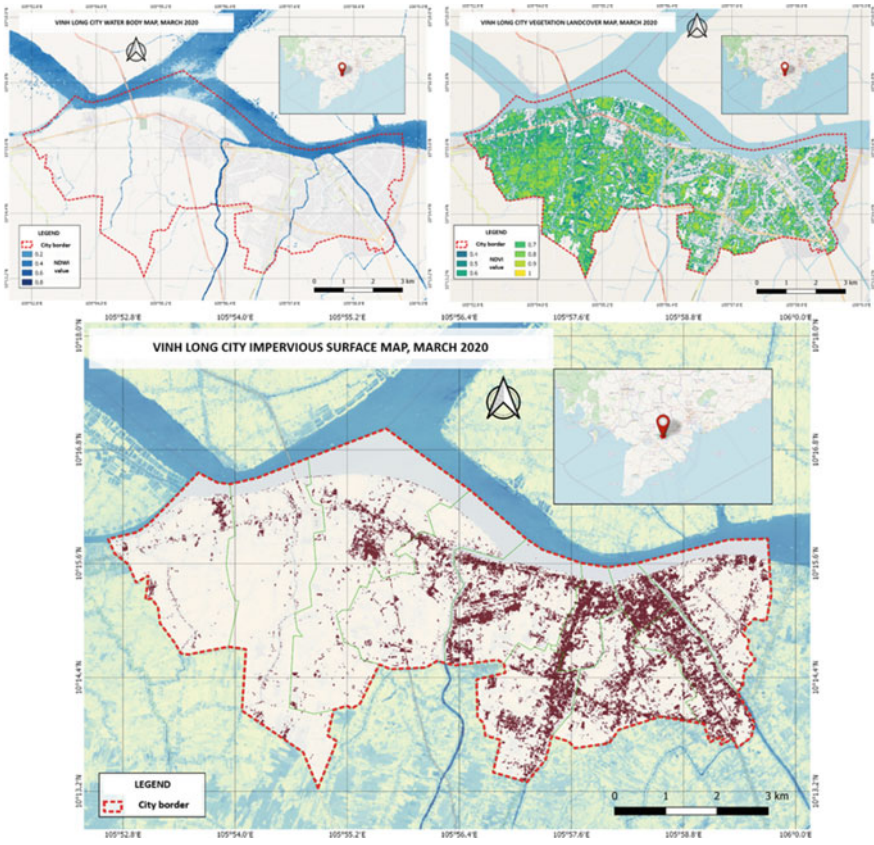


**Fig. 5** Classified landcover maps of Vinh Long city in March 2020

**Table 2** Percentage of impervious surfaces of areas in Vinh Long city

| No | Area | Area (km$^2$) | Impervious surface area (km$^2$) | Ratio of impervious surface (%) | Ratio of impervious surface to catchment area (%) |
|---|---|---|---|---|---|
| 1 | Ward 1 | 0.94 | 0.52 | 55.32 | 68.49 |
| 2 | Ward 2 | 1.53 | 0.61 | 39.63 | 45.57 |
| 3 | Ward 3 | 3.81 | 0.86 | 22.66 | 22.91 |
| 4 | Ward 4 | 2.36 | 1.00 | 42.20 | 43.91 |
| 5 | Ward 5 | 4.01 | 0.68 | 17.00 | 21.00 |
| 6 | Ward 8 | 3.45 | 0.89 | 25.77 | 26.20 |
| 7 | Ward 9 | 4.69 | 1.08 | 22.95 | 26.46 |
| 8 | Truong An Ward | 5.52 | 0.41 | 7.48 | 8.71 |
| 9 | Tan Ngai Ward | 8.93 | 0.57 | 6.35 | 7.56 |
| 10 | Tan Hoa Ward | 7.52 | 0.19 | 2.48 | 2.65 |
| 11 | Tan Hoi Ward | 5.22 | 0.20 | 3.92 | 4.16 |
| 12 | Vinh Long city | 47.98 | 7.00 | 14.60 | 16.28 |

prove the above statement. The average access distance to natural permeable spaces in the central areas is 3–8 times higher than that of the suburban wards. Meanwhile, the coefficient of variation shows the dispersion of the data, or in other words, the difference between the natural permeation spatial distances in the areas. For the entire Vinh Long city area, this coefficient is 1.88, showing that there is a significant difference between the spatial distribution of natural permeation in the study area.

# 6 Discussion

This study shows that analyzing the accessibility to the natural drainage space of an urban area is a crucial factor for assessing its sustainable drainage capacity. The analytical method is precisely quantified with the Sentinel-2 remote sensing image source for land cover classification and the spatial statistical analysis tools of QGIS. Typically, in the case study of Vinh Long, although the percentage of impervious surface is not high (16.28% by catchment area), the uneven distribution of drainage spaces in urban areas leads to a decrease in rainwater drainage efficiency and increased risk of flooding. In the urban center, the average and maximum access distances to natural drainage spaces are much higher than in the surrounding areas and correspondingly the level of flood risk in these areas. Through analysis, a low degree of interleaving of permeable surfaces with impervious surfaces has reduced urban porosity. Thereby reducing the capacity of sustainable drainage in the center of Vinh Long city.
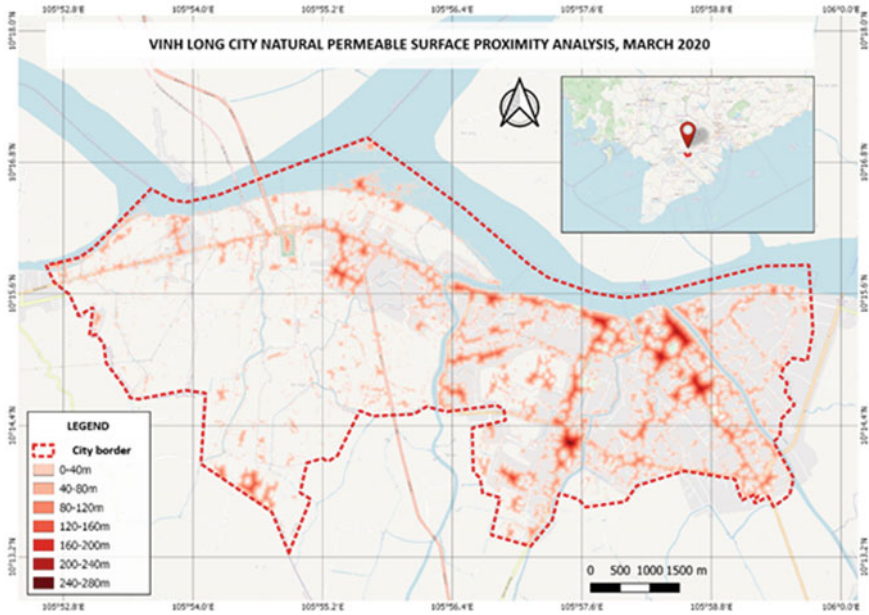
**Fig. 6** Analysis map of the distance to reach the natural permeable space in Vinh Long city in March 2020 with Sentinel-2 image data
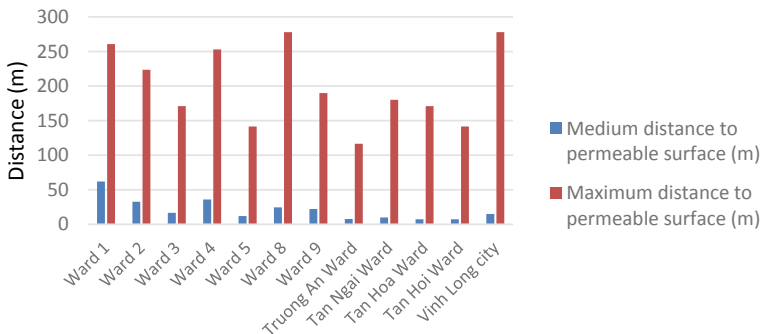


**Fig. 7** Statistical analysis results of the access distance to natural permeable space of areas in Vinh Long city

For other cities in the Mekong Delta, in the low-lying terrain of the region, the interleaving of permeable surfaces with impervious surfaces is a key factor in assessing the potential for surface water drainage towards sustainability. And this criterion can be evaluated through the asymptotic distance analysis method as presented in the article. The less accessible areas are to naturally permeable spaces, the greater the risk of flooding due to the high percentage of surrounding surfaces that are impermeable. In addition, these areas are more difficult to drain surface water, are

more susceptible to urban heat island effects, lack open spaces, and lack biodiversity in ecosystems. Therefore, for urban core areas, it is necessary to supplement natural permeable spaces with sustainable urban drainage solutions, nature-based solutions and low-impact development solutions. The goal is to subdivide the impervious surface, helping the city act like a sponge, thereby increasing its resilience to the increased risk of flooding due to climate change.

Due to the influence of the atmosphere, it is difficult to choose the recording time of satellite images in the Mekong Delta. The threshold of land cover classification value needs to be increased in accuracy to meet the needs of analysis at various times and locations of analysis. To improve accuracy in urban land cover classification, future research could apply machine learning algorithms that use training data to increase efficiency. To increase resolution and avoid limitations due to cloud effects, unmanned aerial vehicle low-altitude remote sensing (UAV-LARS) can be used but requires higher operating costs.

In the context of the lack of databases in planning and decision-making to respond to climate change in Mekong Delta cities, the approach from remote sensing and GIS technology has important implications for supplementing resource data for urban management. Thereby helping the city to be smarter and effectively respond to the impacts of climate change. The approach contributes to changing the city government's vision of applying technology in urban management, supporting the digital transformation process and increasing the efficiency of Industry 4.0 for Vietnamese cities. Future research should expand the applicability of this data and integrate the data of natural drainage analysis in urban planning. From there, develop indicators and criteria for sustainable urban drainage and legalize it in national standards and local drainage management regulations.

Looking towards wider collaboration between different disciplines and industries, the digitalization of sustainable drainage systems would bring critical topography datasets that can be combined with various building, engineering or architectural datasets on a single platform [16]. The potential benefits of interoperability will bring the Mekong Delta closer towards the realization of the Forth Industrial Revolution.

## 7 Conclusion

Urban development adapting to the water factor with solutions like SUDS is the right direction for cities in the Mekong Delta that are significantly affected by climate change and sea level rise. To comprehensively assess the urban sustainability of surface water drainage, in addition to assessing the percentage of impervious surfaces, an additional assessment of the interleaving distribution of natural permeable spaces is required. through asymptotic analysis. This distribution determines the porosity of the urban surface. For advanced assessment, the coefficient of variation applied in this paper can be used in conjunction with the ratio of impervious surfaces to determine the degree of flood risk due to the low porosity of an urban area. The analysis results are used to monitor the characteristics and changes of impervious

and naturally permeable surfaces in urban areas. From there, the data can be used for planning drainage in a sustainable way, building indicators and quantitative indicators in local regulations on drainage management. Moreover, with a temporal resolution of 5 days and a spatial resolution of 20 m, the open data source image from Sentinel-2 satellite is of great significance to meet the needs of urban surface analysis data in Mekong Delta region.

# References

1. Nhan H (2020) Evaluation of sustainbale drainage system aplication by pilot projects for urbans in Mekong delta. Vietnam J Construct 628:36–41
2. Tien N, Nhan H, Hoang N (2021) Urban drainage infrastructure in Mekong Delta river under the impact of climate change. Urban Rural Plann J 112+113, 156–163
3. Gatti A, Naud C, Castellani C, Carriero F (2021) Sentinel-2 products specification document. Thales Alenia Space
4. Anh V (2020) Annual Mekong delta economic report 2020. Vietnam Chamber of Commerce and Industry, Can Tho
5. Master Plan of Mekong Delta to 2030, with a further view towards 2050. Royal HaskoningDHV, Ha Noi (2020)
6. Master plan for irrigation in the Mekong Delta in the context of climate change and sea level rise. SIWRP, Ha Noi (2012)
7. Vietnam Cities Profile. UN-Habitat, Ha Noi (2014)
8. McGrath T (2019) From policy to citizen - final report of the urban drainage and flood proofing program responding to climate change (FPP). GIZ, Ha Noi
9. Woods Ballard B, Wilson S, Udale-Clarke H, Illman S, Scott T, Ashley R, Kellagher R (2015) The SuDS manual (C753). CIRIA, London
10. National Technical Regulations on "Construction planning", No. QCVN 01:2021/BXD. Vietnam Ministry of Construction, Ha Noi (2021)
11. Project on urban development and climate change adaptation in Vinh Long city - Vinh Long province. The World Bank, Ha Noi (2020)
12. Xu R, Liu J, Xu J (2018) Extraction of high-precision urban impervious surfaces from sentinel-2 multispectral imagery via modified linear spectral mixture analysis. Sensors 18:2873
13. Noi D (2018) Research using GIS to build the flow coefficient map of Bac Kan province Bac Kan: University of Transportation. J Transp - Univ Transp Commun. T5
14. Van Trung L, Nguyen Vu N (2018) Application of remote sensing and GIS for assessing the urbanization trend in Can Tho city. Sci Technol Develop J - Sci Earth Environ 2:52–57
15. Wong PF, Chia FC, Kiu MS, Lou ECW (2021) Potential integration of blockchain technology into smart sustainable city (SSC) developments: a systematic review. Smart Sustain Built Environ (in press). https://doi.org/10.1108/SASBE-09-2020-0140
16. Mellado F, Wong PF, Amano K, Johnson C, Lou ECW (2020) Digitisation of existing buildings to support building assessment schemes: viability of automated sustainability-led design scan-to-BIM process. Architect Eng Design Manage 16(2):84–99

# Dynamic Modelling of Wind Turbine Structure for Health Monitoring

**Amna Algolfat, Alhussein Albarbar, and Weizhuo Wang**

Autonomous structural health monitoring is crucial for the operation of wind turbines, particularly offshore ones. The application of the digital twin could be a practical approach. An accurate and computationally efficient structural dynamics model is an essential ingredient of a wind turbine digital twin; so that the virtual model could make online predictions and decisions with lively sensed data from the physical structure. In this study, simulation is built on real-world data to create a structural health monitoring paradigm of the offshore wind turbine blade. The National Renewable Energy Laboratory (NREL) 5 MW reference wind turbine blade is adopted to predict performance and detect damage. A computationally efficient finite element model was adopted to create a dynamics model for modal properties. The forced vibration analysis based on the strip theory is employed to find the dynamic response. The model is then implemented to identify the damage indices via reducing the stiffness locally and explicitly portraying the sensitivity characteristics of the blade' segments along its length span.

## 1 Introduction

The wave of industry 4.0 that will change many traditional industrial concepts in different ways is associated with smart manufacturing [1]. The smart industry is fast transmission and upgrading with big data and the digital twin. The digital twin is first

A. Algolfat (✉) · A. Albarbar · W. Wang
Department of Engineering, Manchester Metropolitan University, Manchester M1 5GD, UK
e-mail: AMNA.ALGOLFAT@stu.mmu.ac.uk

A. Albarbar
e-mail: A.Albarbar@mmu.ac.uk

W. Wang
e-mail: w.wang@mmu.ac.uk

created when the physical system can start providing data to the virtual system model to create a model instance that reflects any physical system's structural, performance, maintenance, and operational health characteristics [2]. Virtual models of specific physical objects are created digitally to simulate their behaviours in the real world [3]. According to [4], the digital twin is composed of three components: the physical one, the virtual models in the virtual world and the third component is the tie operation between the first two components, the physical and virtual components. The virtual processes can guide the physical process to perform the optimised solution [5]. The digital twin of a complex system such as the offshore wind turbines is the operation of transmitting the real-time data to the virtual model to complete the simulation and validation and make operational adjustments. By comparing the physical data with the virtual data, damage problems can be avoided before leading to failure and downtime. The integration between the virtual and physical systems that translate into the digital twin has facilitated these tasks. The real-time generated data may be challenging to analyse by traditional methods [1]. This study's contribution includes using the vibration analysis to create a virtual model of an offshore wind turbine blade built on the 5 MW NREL, extracting the dynamic parameters, aerodynamic forces, and dynamic response. Dynamic response variation due to structural damages may be simulated with a proper and lower-order finite element model. Such information may then be applied to health monitoring.

## 2 Modelling and Validation

The specification of a 5 MW NREL offshore blade, as shown in Fig. 1, is used for modelling and validation [6]. This study aims to build a virtual model that can integrate real-time sensing data to improve the blade capability and test different operating parameters. In this section, the structural model and aerodynamic model will be presented. Then the damage indexes based on the model parameters and dynamic response will be employed to portray the computational model of the offshore horizontal axis wind turbine blade under the flap-wise vibration. The blade is dominantly deformed due to the different and complex aerodynamic loads. Therefore, the study starts with health conditions to extract the intact model parameters, such as the stiffness, frequencies, mode shapes, forced response and the health strain energy. All these parameters will be affected according to the degree of the gradient deformation of the blade. As a result, the reversible use of pre-stressed stiffness will identify the damage occurrence and predict the blade performance. In the application of digital twins during operations, the computational efficiency of the digital model is crucial. So that 'online' decision can be made from the lower-order digital models' prediction with consideration of live sensing data. In the following sections, lower-order computational structural and aerodynamic models, i.e., beam theory and blade element momentum theory, will be studied and compared with 'higher-order' models such as 3D finite element and CFD (computational fluid dynamics) models.
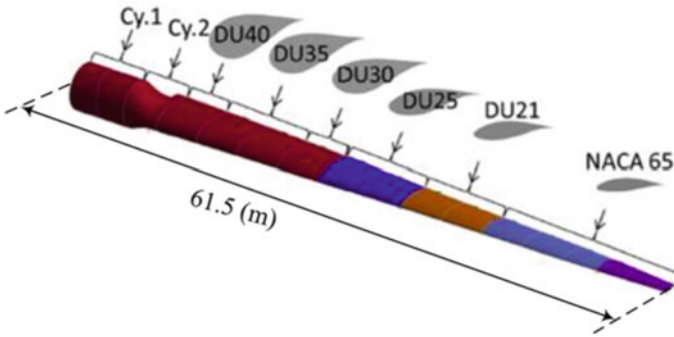
**Fig. 1** Division of the blade into 17 elements [16]
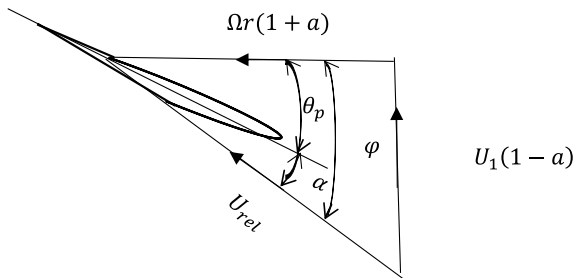


**Fig. 2** Blade element velocities



8.2+4.1+8.2+4.1+8.2+8.2+20.1=61.5m
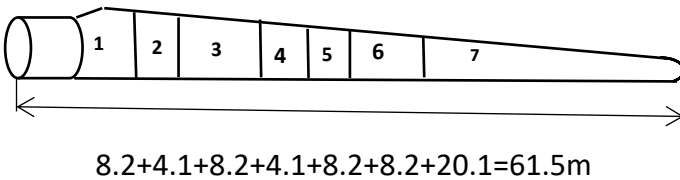
**Fig. 3** 5 MW NREL blade with seven different sections distributed along the blade span.

## 2.1 Structural Modelling

This study employs a nonlinear beam model based on the Ray-Leigh theory to extract the model parameters. The blade is considered to rotate at an angular velocity $\Omega$. The blade undergoes flap-wise bending vibration, and the governing equation can be obtained by using Hamilton's principle [10].

$$\rho A(x)\frac{\partial^2 w}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left(EI(x)\frac{\partial^2 w}{\partial x^2}\right) - \frac{\partial}{\partial x}\left(T(x)\frac{\partial w}{\partial x}\right) - \frac{\partial}{\partial x}\left(\rho I(x)\frac{\partial^3 w}{\partial x \partial t^2}\right)$$

$$+ \Omega^2 \frac{\partial}{\partial x}\left(\rho I^*(x)\frac{\partial w}{\partial x}\right) - k(x)w = f(x, t) \tag{1}$$

where $I^* = I(x)\cos^2(\phi)$ and $\phi$ is the pre-cone angle, $\rho$, represents the blade density, $A(x)$ its cross-sectional area in the flap-wise direction at distance $x$ relative to the blade span, $I(x)$ blade moment of inertia, $\Omega$ angular velocity of the blade, $\theta_p$ pitch angle, $w(x, t)$ is the flap-wise bending displacement; t is time; $T(x)$ is the axial force due to the centrifugal tension force at a distance $x$ from the centre of rotation, may be written as

$$T(x) = \frac{1}{2}\int\limits_x^L \rho A(x)\big[(\Omega^2 (R + x\cos(\phi)) - g\cos(\Omega t)\big]dx \tag{2}$$

where $L$ is the blade length, and

$$k(x) = -\frac{1}{8}\rho A(x)\big(2\cos(2\phi) + \cos(2(\phi - \theta_p)) + 2\cos(2\theta_p) + \cos(2(\phi + \theta_p)) - 6\big)\Omega^2$$

## 2.2 Aerodynamic Module

Blade element momentum theory is addressed to calculate the aerodynamic loads. This theory is based on integrating 2D airfoil aerodynamic loading along the blade span. In this theory, the force of a blade element is solely responsible for the change of momentum of the air that passes through the annulus swept by the element. According to [11] the components of aerodynamic forces on blade element is:

$$\overrightarrow{f_{aero}} = \overrightarrow{i_{YZ}}\left\{\begin{matrix} f_T \\ f_N \end{matrix}\right\} = \overrightarrow{i_{YZ}} = \left\{\begin{matrix} \mathcal{L}\sin(\alpha + \beta) - \mathcal{D}\cos(\alpha + \beta) \\ \mathcal{L}\cos(\alpha + \beta) + \mathcal{D}\sin(\alpha + \beta) \end{matrix}\right\} \tag{3}$$

$$\mathcal{L} = 0.5\rho_a c C_l U_{rel}^2$$

$$\mathcal{D} = 0.5\rho_a c C_d U_{rel}^2$$

where $f_T, f_N$ are the aerodynamic forces in the tangential and normal directions. $\mathcal{L}, \mathcal{D}$ are the lift and drag forces for a unit blade length. $C_l, C_d$ are the lift and drag coefficients. $\rho_a, c$ are the air density and chord length. $\alpha$ is the angle of attack. The total pitch angle $\beta$ is the sum of the element twist angle and the blade pitch angle at the considered blade section (Fig. 2).

## *2.3 Free Vibration Analysis*

The governing Eq. (1) together with approximate solution $w(x, t)$ is employed to formulate the weak form as:

$$\int_0^L \rho A(x) \frac{\partial^2 w(x, t)}{\partial t^2} \varphi_i dx + \int_0^l \frac{\partial^2}{\partial x^2} \left( EI(x) \frac{\partial^2 w(x, t)}{\partial x^2} \right) \varphi_i dx$$

$$- \int_0^L \frac{\partial}{\partial x} \left( T(x) \frac{\partial w(x, t)}{\partial x} \right) \varphi_i dx - \int_0^L \frac{\partial}{\partial x} \left( \rho I(x) \frac{\partial^3 w(x, t)}{\partial x \partial t^2} \right) \varphi_i dx \qquad (4)$$

$$- \int_0^L k(x) w(x, t) \varphi_i dx + \int_0^L \Omega^2 \frac{\partial}{\partial x} \left( \rho I^*(x) \frac{\partial w(x, t)}{\partial x} \right) \varphi_i dx - \int_0^L f(x, t) \varphi_i dx = 0$$

where, $\varphi_i$ are the test functions, and $i = 1, 2, \ldots N$.

The flap-wise deflection and the slop due to flexural vibration of the wind turbine blade are approximated by a weighted linear combination of trial functions [9]

$$w(x, t) = \sum_{j=1}^N \varphi_j(x) u_j(t) \qquad (5)$$

The second derivatives of the trial solution are expressed as:

$$\frac{\partial^2 w(x, t)}{\partial x^2} = \sum_{j=1}^N \frac{\partial^2 \varphi_j(x)}{\partial x^2} u_j(t) \qquad (6)$$

and

$$\frac{\partial^2 w(x, t)}{\partial t^2} = \sum_{j=1}^N \varphi_j(x) \frac{\partial^2 u_j(t)}{\partial t^2} \qquad (7)$$

The generalised eigenvalue problem may be expressed as:

$$-\omega^2 X + M^{-1} K X = 0 \qquad (8)$$

The elements of intact mass $M$ and stiffness $K$ matrices are:

$$M_{ij} = \int_0^L \rho A(x) \varphi_j(x) \varphi_i(x) dx + \int_0^L \rho I(x) \frac{\partial \varphi_j}{\partial x} \frac{\partial \varphi_i}{\partial x} dx \qquad (9)$$

and

$$K_{ij} = \int\limits_0^L EI(x)\frac{\partial^2\varphi_i(x)}{\partial x^2}\frac{\partial^2\varphi_j(x)}{\partial x^2}dx + \int\limits_0^L T(x)\frac{\partial\varphi_i}{\partial x}\left(\frac{\partial\varphi_j}{\partial x}\right)dx$$

$$- \int\limits_0^L k(x)\varphi_j\varphi_i dx - \int\limits_0^L \Omega^2\rho I^*(x)\frac{\partial\varphi_i}{\partial x}\left(\frac{\partial\varphi_j}{\partial x}\right)dx \qquad (10)$$

## 2.4 Damage Modelling to the Wind Turbine Blade

The structural deformation of the blade may occur in the flap-wise, edgewise, longitudinal, or torsional directions. All the damage indexes in the mentioned deformational directions must be considered based on their influences on reducing the dynamic parameters of the whole structure. In this study, the comparison was made between the intact vibrational data and damaged sections resulting from the flap-wise direction deformation [12–14]. Moreover, any change in the modal properties at any blade segment may be translated to the occurrence of damage. Damage is simulated to the wind turbine blade numerically by reducing the stiffness of a specific segment. The coefficient of stiffness reduction ranges from 50 to 90% from the value of the intact blade stiffness value. Figure 3 shows the wind turbine blade with seven different sections distributed along the blade span. The section number is denoted as shown in the figure, where the numbers increase from the root to the blade's tip.

The general form of the eigenvalue problem for the *ith* mode may be expressed as

$$[K - \lambda_i M]X^i = 0$$

If the damage determines a reduction of the stiffness matrix, then

$$[(K + \Delta K) - (\lambda_i + \Delta\lambda)M](X^i + \Delta X^i) = 0 \qquad (11)$$

with

$$\Delta\lambda_i = X^{(i)T}\Delta K X^{(i)} \qquad (12)$$

and

$$\Delta X^{(i)} = \sum_{j=1}^{N} \frac{X^{(j)T}\Delta K X^{(i)}}{\lambda_i - \lambda_j} \qquad (13)$$

which shows that the variation in the mode shapes displacement and frequencies are related to the healthy dynamic parameters of the undamaged structure.

## 3 Results and Discussion

The properties of the 5 MW NREL blade are represented in Table 1. The first ten natural frequency results of a single 5 MW NREL blade without aerodynamic forces are compared with data from [10, 15, 17], listed in Table 2. The results showed a good agreement with the literature results, which indicated the appropriateness of the presented model using the finite element model with beam theories. The aerodynamic load distribution along the blade span at the rated operating condition in the normal and tangent direction is illustrated in Fig. 4 as annotated in orange. The results agree well with different literature data, as indicated in overlapping curves in Fig. 4. The distribution of dynamic displacement response in the flap-wise and edgewise direction along the blade span is shown in Fig. 5. Both figures agree with literature results as indicated in the comparison of the dynamic response results in the edgewise and flap-wise direction, and the total rotor power and thrust force are shown in Table 3. The code results are compared with [15–19], which used computational structural dynamic model and BEM-ABAQUS coupling model are presented in Table 3.

Several damage scenarios were chosen along the blade's span. The dynamic response of the blade according to the location of damage and its severity can be investigated. The response is obtained from the health state of the blade, and it is

**Table 1** The gross properties of the NREL 5 MW wind turbine blade

| Parameters | Value |
| --- | --- |
| Rating | 5 MW |
| Blade number | 3 |
| Rotor diameter, hub diameters and height | 126 m, 3 m, 90 m |
| Cut in and out wind speed | 3 m/s, 11.4 m/s, 25 m/s |
| Rotor speed | 6.9RPM, 12.1RPM |
| Company | NREL |

**Table 2** Natural frequencies of a single blade without aerodynamic force at zero rotation speed

| Mode NO | Present | BModes (Hz) [17] | FAST (Hz) [17] | [15] (Hz) | [17] (Hz) |
| --- | --- | --- | --- | --- | --- |
| 1 | 0.680 | 0.69 | 0.68 | 0.673 | 0.68 |
| 2 | 1.11 | 1.12 | 1.10 | 1.11 | 1.10 |
| 3 | 1.98 | 2.00 | 1.94 | 1.926 | 1.98 |
| 4 | 4.10 | 4.12 | 4.00 | 3.96 | 3.99 |
| 5 | 4.54 | 4.69 | 4.43 | 4.43 | 4.66 |
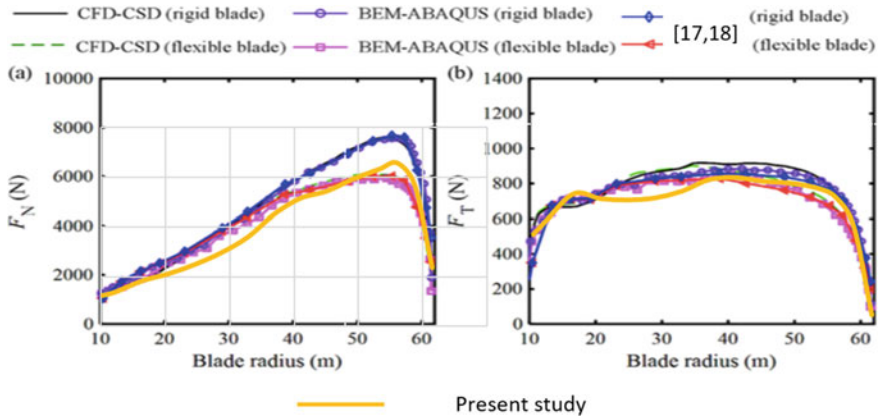
**Fig. 4** The aerodynamic load distribution along blade span at the rated operating condition; and the comparison by over-plotting the curves of the present study on the ones from the literature [16, 17]
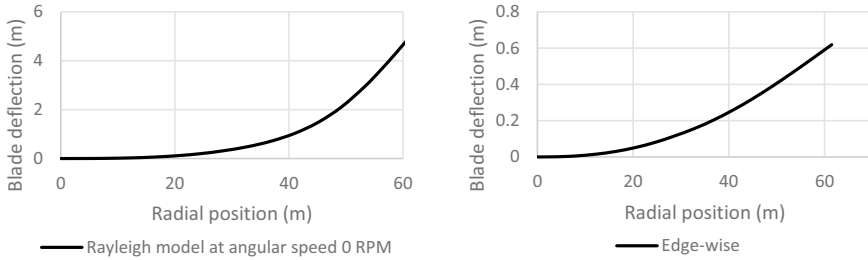


**Fig. 5** Span-wise distribution of blade defection in the flap-wise and edgewise directions

**Table 3** Comparison of the dynamic response under nominal working conditions

| Reference | Flap-wise deflection (m) | Edgewise deflection m | Rotor power MW | Thrust force KN |
|---|---|---|---|---|
| Zhanwei et al. [15] | 4.49 | 0.57 | 5.3 | 678.44 |
| Sapal and Gopa [16] | 4.41 | 0.57 | 4.97 | 690.72 |
| Jonkman et al. [17] | 5.47 | 0.61 | 5.28 | 814.45 |
| Yu and Kwon [18] | 4.72 | 0.63 | 5.22 | 656.43 |
| Ponta et al. [19] | 3.85 | 0.56 | 5.19 | 660.19 |
| Sabale and Gopal [16] | 4.55 | 0.62 | 5.17 | 676.12 |
| Jeong et al. [17] | 4.83 | 0.75 | – | – |
| Present | 5.00 | 0.618 | 4.93 | 602 |

**Fig. 6** First six mode shapes **a**. Intact blade, **b**. Damage blade with reduction coefficient $= 0.8$ at blade length 53.3 m

used as a baseline to identify the health behaviour of different blade sections. Then, damage detection is accomplished by comparing the healthy response baseline to its damaged counterpart. The degree of deviation along the blade span indicates the occurrence of damage, its location and severity.

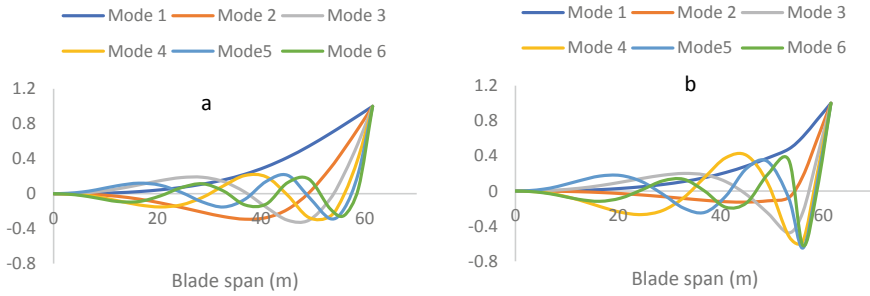The first six mode shapes of the healthy blade compared to the damaged blade at the critical section at 53.3 m from the blade root with a stiffness reduction of 80% is shown in Fig. 6.

The dynamic characteristics and forced response along the blade span are used to create a new health model, as shown in Fig. 7. The blade health model is used to compare the damaged occurrence by reducing the blade's stiffness by different percentages from 65 to 90%. From Fig. 7, it can be noted that the response deflection is increased as the stiffness reduction increases. The critical limit of stiffness reduction at length 53.3 m from the blade root is 65% compared to the healthy baseline. The response is completely changed when the value of stiffness reduces to 0.65, where the blade section's behaviour takes a different configuration.

## 4 Conclusion

In this study, the National Renewable Energy Laboratory (NREL) 5 MW reference wind turbine blade is modelled using beam theory for structure and BEM theory for aerodynamics due to their computational efficiency, which is an important factor in the application of digital twin to allow live comparison of model prediction to sensed data. Also, structural damage simulations were carried out on the blade. The healthy dynamic vibration response can be used as a baseline to predict the damage occurrence. Damage detection and identification may be possible via monitoring the changing of the blade dynamic response behaviour.
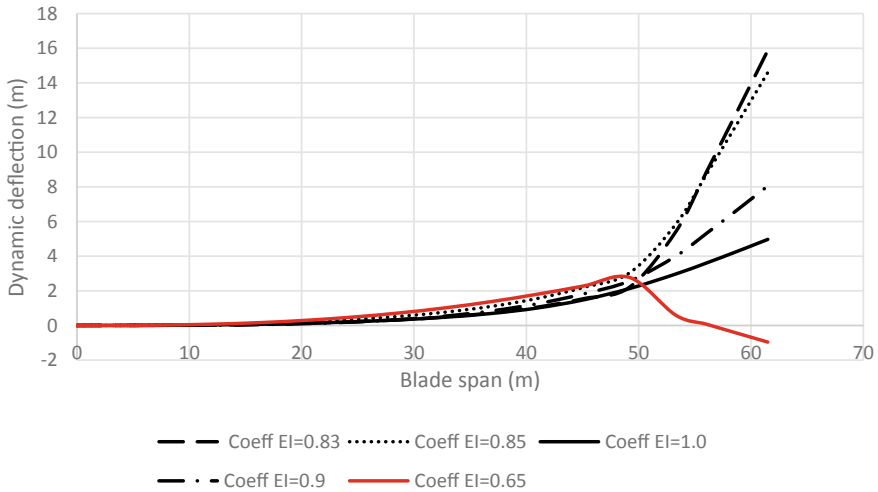
**Fig. 7** Blade dynamic deflection due to different stiffness reduction coefficient concentrated at 53.3 m from the blade root in the flap-wise direction

# References

1. Mittal S, Khan MA, Romero D, Wuest T (2019) Smart manufacturing: characteristics, technologies and enabling factors. Proc Inst Mech Eng Pt B J Eng Manuf 233(5):1342–1361
2. Madni AM (2015) Expanding stakeholder participation in upfront system engineering through storytelling in virtual worlds. Syst Eng 18(1):16–27
3. Hochhalter J, Leser WP, Newman JA, Glaessgen EH, Cornell SR, Willard SA, Heber G (2014) Coupling damage-sensing particles to the digital twin concept, 2014. NASA Report
4. Grieves M (2014) Digital twin: manufacturing excellence through virtual factory replication. White Paper 1:1–7
5. Fei T, Jiangfeng C, Qinglin Q, Zhang M, Zhang H, Fangyuan S (2018) Digital twin-driven product design, manufacturing and service with big data. Int J Adv Manuf Technol 94(9–12):3563–3576
6. Resor BR (2013) Definition of a 5MW/61.5 m wind turbine blade reference model. Sandia National Laboratories, SAND2013–2569, Albuquerque, New Mexico, USA
7. Griffith DT, Ashwill TD (2011) The Sandia 100-meter all-glass baseline wind turbine blade: SNL100–00. Sandia National Laboratories Technical Report, SAND2011–3779
8. Mandell JF, Samborsky DD (1997) Composite materials fatigue database: test methods, materials and analysis. Sandia Report SAND97–3002, Sandia National Laboratories, Albuquerque, NM, USA
9. Algolfat A, Wang W, Albarbar A (2022) Dynamic responses analysis of a 5MW NREL wind turbine blade under flap-wise and edge-wise vibrations. J Dyn Monit Diagn 13:208–222
10. Algolfat A, Wang W, Albarbar A (2022) Comparison of beam theories for characterisation of a NREL wind turbine blade flap-wise vibration. Proceedings of the Institution of Mechanical Engineers, Part A: J Power Energy 236(7):1350–1369
11. Burton T, Jenkins N, Sharpe D, Bossanyi E (2011) Wind energy handbook. John Wiley & Sons
12. Algolfat A, Wang W, Albarbar A (2023) The sensitivity of 5MW wind turbine blade sections to the existence of damage. Energies 16(3):1367
13. Fang SE, Perera R (2009) Power mode shapes for early damage detection in linear structures. J Sound Vib 324(1–2):40–56

14. Rezaei MM, Behzad M, Moradi H, Haddadpour H (2016) Modal-based damage identification for the nonlinear model of modern wind turbine blade. Renew Energy 94:391–409
15. Li Z, Wen B, Dong X, Peng Z, Qu Y, Zhang W (2020) Aerodynamic and aeroelastic characteristics of flexible wind turbine blades under periodic unsteady inflows. J Wind Eng Ind Aerodyn 197:104057
16. Sabale AK, Gopal NK (2019) Nonlinear aeroelastic analysis of large wind turbines under turbulent wind conditions. AIAA J 57(10):4416–4432
17. Jeong MS, Cha MC, Kim SW, Lee I, Kim T (2014) Effects of torsional degree of freedom, geometric nonlinearity, and gravity on aeroelastic behavior of large-scale horizontal axis wind turbine blades under varying wind speed conditions. J Renew Sustain Energy 6(2):023126
18. Yu DO, Kwon OJ (2014) Predicting wind turbine blade loads and aeroelastic response using a coupled CFD–CSD method. Renew Energy 70:184–196
19. Ponta FL, Otero AD, Lago LI, Rajan A (2016) Effects of rotor deformation in wind-turbine performance: the dynamic rotor deformation blade element momentum model (DRD–BEM). Renew Energy 92:157–170

# Evaluating a Covid-19 Vaccine Centre in the UK Using a DES Model

**Iain Reid, Saikat Kundu, and Muhammad Latif**

## 1 Introduction

The outbreak of COVID-19 has forced society to find suitable approaches to resolve this worldwide pandemic. It is often unknown how the virus may evolve, but a common belief among scientists is that an effective answer is mass vaccination, to finish the COVID-19 pandemic [1]. Many research organisations are collaborating globally on research and development of vaccines to support the vulnerable populations to become safe from infection [2].

It is clear that mass vaccination needs to be introduced and implemented as rapidly as possible to bring the pandemic under control and return to some kind of normality [1, 3–5]. A COVID-19 mass vaccination is probably the most challenging crisis of the decade in public health. In the UK, a programme of mass vaccination centres have been established often on make-shift sites in cities and towns.

The UK vaccine programme has been developed to inoculate the population against COVID-19. The success of the programme is dependent on an effective vaccine delivery which requires military style planning, preparation and delivery [6]. This includes vaccination priorities, delivery infrastructure, people awareness, and design of vaccination centres. The need to establish and maintain efficient vaccination centres during their operations has prompted this study. It is very evident that modelling and simulation tools have the ability to build capacities and facilitate efficient planning.

The paper introduces a software model, built to analyse the operations of a mass vaccination centre. The model is developed using a discrete event simulation (DES) method. The model allows users to evaluate patient flow for vaccination and the

I. Reid
Faculty of Business and Law, Manchester Metropolitan University, Manchester, UK

S. Kundu · M. Latif (✉)
Department of Engineering, Manchester Metropolitan University, Manchester, UK
e-mail: m.latif@mmu.ac.uk

resource levels to maintain the facility with varying situations and formations. The simulation scenarios have the potential to assist the planners to look ahead and improve current and future vaccination strategy implementations.

The paper is arranged as follows: In Sect. 2, related work is presented followed by the methodology in Sect. 3. The methodology introduces the operations in the vaccination centre. This is followed by data collection, data analysis, model building and model testing. Section 4 shows the results and experimentation. Discussions and conclusions are presented in Sect. 5, which concludes the paper.

## 2 Related Work

In our methodology, we have selected to use DES to evaluate a vaccination centre. The technique of DES is established and popular in healthcare applications [7]. DES application in healthcare has been popular [8]. Our work focuses on applying DES to the operations in a vaccination centre to determine patient throughput and flow time. Many publications in this domain tend to focus on the vaccine effectiveness on the global population level rather than at vaccination centre level [9, 10]. Those studies focusing at clinical level are usually looking at different infection contexts [11] such as influenza [12]. One recent publication used DES and agent-based simulation to model the operations of a drive-through mass vaccination clinic, however, this study has not been informed by empirical data [13]. Most studies of this type use the DES approach to stochastically simulate service systems with patients that flow through a healthcare environment, with limited resources such as appointment slots [14]. Other studies focus on patient concerns on vaccines and their hesitancy [1]. In summary, there appears to be limited literature on using DES to evaluate a mass vaccination centre at an operational level, and our study helps address this gap.

## 3 Methodology

The study was focussed on the vaccination centre operations and the KPIs of hourly capacity and patient flow time. Arrival of patients to the centre is assumed to be by car or walk-ins on the basis of 50:50 ratio. Data collection relied upon observational data of a vaccination centre.

### 3.1 Vaccination Centre Operations

Vaccination centres require careful planning and implementation and are governed by National Health Service (NHS) England guidelines [15]. Staffing levels must enable the individual workstations to be adequately manned. The processing capacity

of the centre (patients per hour) and the patient flow time (time in centre) are the two operational considerations. The capacity of a centre will dictate the quantity of centres that must be opened leading to an estimate of the overall time necessary to vaccinate the population. The flow time is a measure of the time-in-system for the patients within the centre. Patients require space and large queues of patients are to be avoided whilst they wait to receive treatment. If too many patients are in the centre, they cause crowding, confusion and congestion. The balance between centre capacity and flow time is very subjective in mass vaccination and requires to be regularly tweaked to meet operational targets.

A vaccination centre operates on a pre-booked appointment basis. This means that slots are available online for patients to book. The centre receives patients as walk-ins or by car. Either case the patients enters the site via a car park. Cars are directed onto the car park in a controlled manner by marshals (volunteers) who limit car arrivals to the site. Once parked the patient walks to the building and usually joins a patient queue that forms at the entrance. Walk-in patients also join the same entrance queue. The entrance queue moves slowly enabling patients to enter the building containing the centre. Upon building entry, patients have a temperature check done whilst in a moving queue. Patients follow an orderly queue that meanders along the entrance corridor to enter the main hall. Within the main hall, the patient's first stop is Sign-in that involves confirming basic personal details and collecting a personal data sheet. After Sign-in the patient follows the snake like queue and is directed to the next available Vaccination cubicle. Within the Vaccination cubicle, the personal data sheet is collected, information is given, and the vaccine is administered. The patient is then directed to take a seat in the Observation area. Volunteers manage the Observation area and allow patients to exit after a 15-min stay. Patients leave the building through a separate exit and walk through the car park and either drive through or walk through the site gates. A simple flowchart representing the drive-in patient is shown in Fig. 1.

For the vast majority of patients, the procedure described earlier reflects their experience. However, some patients are likely to leave the centre with or without
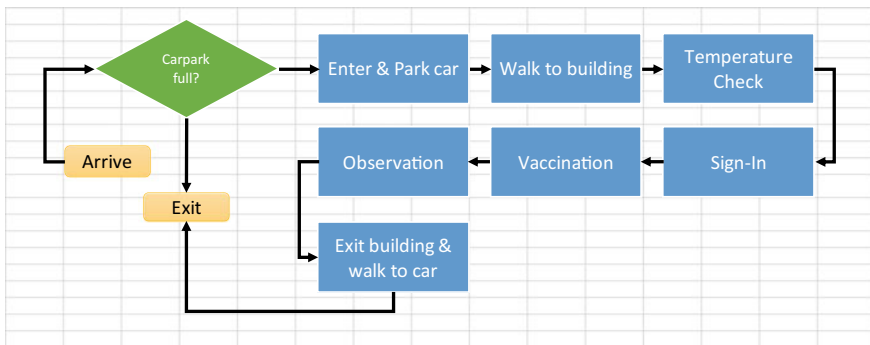


**Fig. 1** Process flowchart for a drive-in patient

vaccination because they have been unsuccessful at any of the stations. These patients have not been considered in this study because observational evidence suggests the failures are negligible.

## 3.2 Data Collection

Operations of the vaccination centre were observed by following the patient flow through the various stations in addition to secondary data [16]. The observations included patient arrivals and mode, queue lengths, walking speeds, distance and capacity of stations, staffing levels, and service times. The various stages involved:

- Arrival (car park—outside the centre)
- Joining the patient queue to enter the centre
- Temperature check whilst in a moving queue line
- Sign-in desk
- Vaccination cubicle
- Observation area
- Exit via the car park

Although the data collection was carefully planned, the actual data obtained was not complete and may have included inaccuracies due to limited accessibility and people for the study. Missing data was estimated from secondary data [16]. Still, the data were sufficient for constructing a valid simulation model.

## 3.3 Data Analysis

The raw data collected from the centre was entered into a spreadsheet. This enabled how long patients are at each workstation and ultimately the flow time in the centre. It was determined to separate the conveyance timings from station timings. This enabled Table 1 to be devised. It was deemed appropriate to use a triangular distribution for conveyance and station times due to the limited data and wide variability. Table 1 also depicts some key parameters and constraints of the centre.

## 3.4 Simulation Model

Discrete event simulation (DES) is a technique for understanding the operational behaviour of a real-world process. DES is progressively being used in health-care services [17]. Improvements in computers and the use of visually interactive simulation have enabled DES to be applied to complex problems.

**Table 1** Operational parameters

| COVID vaccine centre simulation | | | | | | | |
|---|---|---|---|---|---|---|---|
| (Car and walk-in patients) | | *Activity timing distributions (mins)* | | *Conveyance iming distribution (mins)* | | | |
| *Patient arrivals* | | 1. Temperature check | | 1. To park a car | | 2. Walk: car parl to temp-check | |
| Number of arrivals expected per hour | 100 | Minimum | 0.3 | Minimum | 1 | Minimum | 1 |
| Operational hours per day | 12 | Mode | 0.5 | Mode | 3 | Mode | 2 |
| New cars allowed on car park at any one time | 5 | Maximum | 0.8 | Maximum | 5 | Maximum | 3 |
| Enter by car (drive-ins) | 50% | | | | | | |
| | | 2. Sign in | | 3. Walk: Temp-check to sign-in | | 4. Walk: Sign-in to vaccination | |
| *Resources* | | Minimum | 0.3 | Minimum | 0.5 | Minimum | 0.5 |
| Temperature-check | 1 | Mode | 0.5 | Mode | 0.5 | Mode | 0.5 |
| Sign-in staff | 1 | Maximum | 1 | Maximum | 0.5 | Maximum | 0.5 |
| Vaccinators | 8 | | | | | | |
| Observation spaces | 25 | 3. Administer vaccine | | 5. Walk: vaccination to observation | | 6. Walk: observation to car | |
| | | Minimum | 3 | Minimum | 1 | Minimum | 2 |
| *Queue spaces* | | Mode | 5 | Mode | 1 | Mode | 5 |
| From car-park to temp-check | 50 | Maximum | 7 | Maximum | 1 | Maximum | 7 |
| From temp-check to sign-in | 10 | | | | | | |
| From sign-in to vaccination | 10 | 4. Observation | | 7. Car to site exit | | | |
| From vaccination to observation | 2 | Minimum | 15 | Minimum | 1 | | |
| Car park capacity | 50 | Mode | 15 | Mode | 1 | | |
| | | Maximum | 15 | Maximum | 1 | | |

DES is used to model real systems that can be split into a set of individual processes that progress over time. Each event marks a change of state in the system and occurs at a specific instant in time. DES assumes no variation in the system between events. DES is used to describe and evaluate queuing processes and networks of queues where there is an emphasis on the use of resources. The core elements are:

- Entities: are elements that move through the processes and have work done on them e.g. patients
- Resources: are elements that are used in the workflow to process entities e.g. healthcare staff
- Events: critical instants in the system's lifetime e.g. arrival of patients, process finishes
- Queues: waiting lines e.g. queue of patients.

DES is suited to those environments were entity flow exists such as patients flowing through a clinic with limited resources. This type of study allow patients to interact with resources based on their unique attributes.

Due to the superior balance of functionality and ease of use, Witness Horizon software was used to develop a model of the vaccination centre. A table of operational parameters were developed based on observation data and discussions with management of a vaccination centre. The operational parameters and their values are depicted in Table 1.

A mapping activity produced Table 2, enabling the real-world elements to be mapped to Witness Horizon elements.

A DES model was developed and iteratively refined to credibly represent the operations at the target vaccination centre. Figure 3 displays the vaccination centre after 12 continuous hours of simulation. One of the key drivers was to establish what level of service is achieved in relation to a patient arrival rate (hourly). An acceptable level of service was defined by three criteria:

- Average patient flow time is about 40 min
- Maximum patient flow time not to exceed 65 min

**Table 2** Element mapping

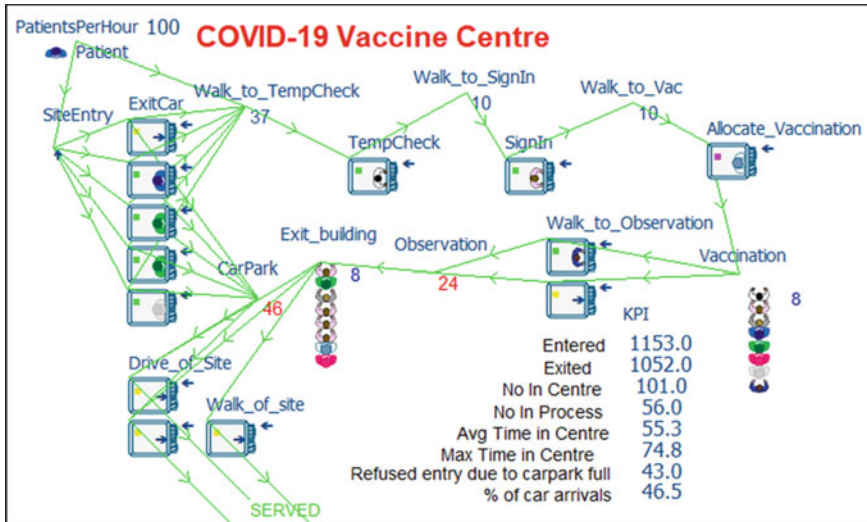| Mapping to witness elements | |
|---|---|
| Description | Witness element |
| Patient | Entity |
| Park car | Activity |
| Walk to building/Sign-in/Vaccination | Queue |
| Temperature check/Sign-in/Vaccination | Activity |
| Observation area | Queue |
| Drive/walk off site | Activity |
| Patient ID/mode of arrival | Attribute |
| KPI display | Variable array |

**Fig. 3** Simulation model @ 12 h

- Total number of patients who refused entry due to carpark being full should not exceed 25 patients a day.

To ensure variability and realism, the patient arrival rate (hourly) was implemented using an exponential inter-arrival time.

## 3.5 Model Testing

To be useful, the model must logically represent the process flow observed in the vaccination centre, known as verification. This was achieved by performing a structured walkthrough of the model code, scrutiny of the display, and numerous test runs.

The model behaviour was observed before the results were recorded. This enabled some confidence that the model is providing real-world outputs. Validation was achieved by confirming that the model behaviour was realistic and close enough to the real-world system. The model validation phase comprised of comparing model results with actual statistics. This step was critical to determine the correctness as it is deemed necessary before performing what-if analysis. As is quite normal in such studies our initial set of results did not closely replicate the performance of the vaccination centre. This was mostly due to poor estimates of the operational data. The model was adjusted to be better aligned with the vaccination centre [16].

A model demo with stake-holders provided a platform for commenting on the model behaviour and helped recognise any glitches. The trustworthiness of any model

is reliant on realistic and reliable data, this is not always possible in large institutions like the British Health Service. In essence, a three-step approach was utilised for model validation comprising of:

- Model building with high face validity.
- Model assumption validation.
- Comparison of the model input–output with the obtained data.

After the model had been validated it was executed for a 12 h of simulation and results were collected. The model during the simulation is imitating the behaviour of the current process. This base-line model provided a dynamic and visual representation of the vaccination centre operations. The dynamic animation provides a very useful insight to the sequence of events for patient flow and adds to the credibility of the model.

## 4 Results

Experimental results are presented alongside parameter variation and sensitivity analysis.

### 4.1 Base Experiment

The model with all work stations operational was run at full capacity. This runs the model for 12 continuous hours under a fixed rate of 100 patient arrivals per hour. The parameter and their values for the experiment are shown in Table 1. All of the parameters are changeable from within the model before or during the simulation at an appropriate stop point.

An arbitrary single simulation run of 12 h of operations is illustrated in Fig. 3.

Figure 3 reveals that most patients spend about 55 min in the centre on average. Patients early in the process spend less time. As more patients enter the centre then waiting for lines form and thus the total flow time increases. The maximum time patients spend in the centre, is shown to be 74 min.

Figure 4 shows the actual number of patients that were in the centre compared to the number of patients being processed. Patients being processed are defined from Temperature Check station to completion of Observation. As illustrated in Fig. 4, the number of patients who entered but were not in the process has steadily grown to around 100 patients. Whilst the number of patients being processed is relatively steady at around 55.

The key performance indicators (KPIs) discussed above reveal that acceptable service levels were not achieved in the base experiment replication.
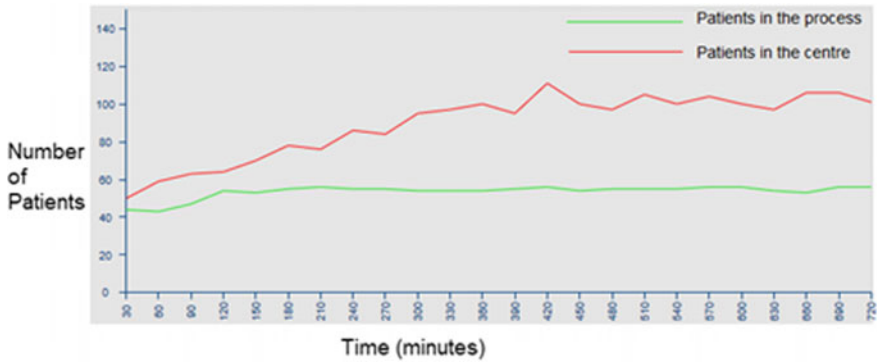
**Fig. 4** Patients in centre versus patients in process

## 4.2 Parameter Variations

Simulation results are presented in this section by varying different parameters including patient arrival rates, drive-in to walk-in ratio, and resource levels.

If we reduce the patient hourly rate to 90 per hour then the results produced are much more favourable (Fig. 5) as they achieve the set service levels at the cost of losing 2% output.

If we increase the ratio of drive-in patients to walk-ins to 60:40 for the base model, we find the results produced are very promising moving towards desired service levels as shown in Fig. 6.

If we increase the vaccination cubicles from 8 to 10 to the base model, understandably the patient output increases by 7%. In addition, the desired service levels are achieved as shown in Fig. 7.

Due to model stochasticity, each replication can produce different results. Therefore, some experimentation was deemed necessary. We have run the base model with
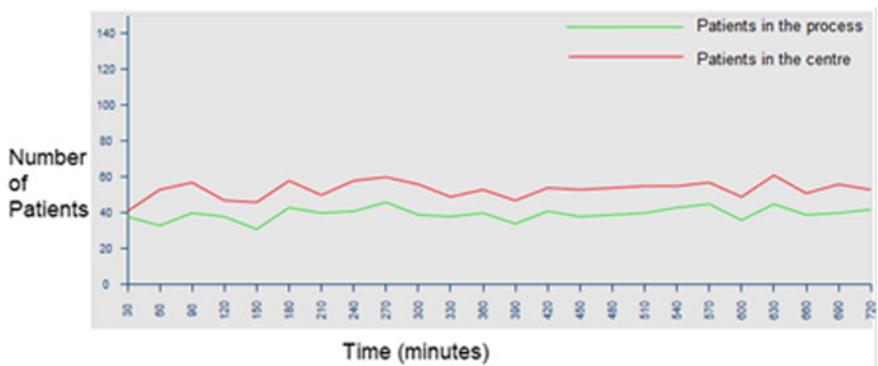


**Fig. 5** Patients in system with reduced arrivals of 90 per hour
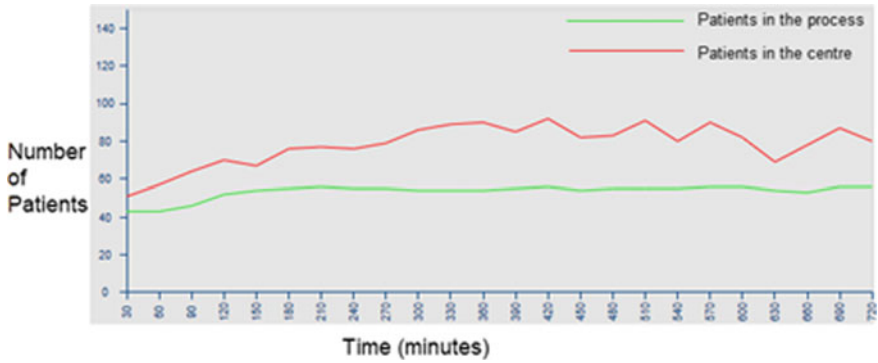
**Fig. 6** Patients in system with 60% drive-in patients
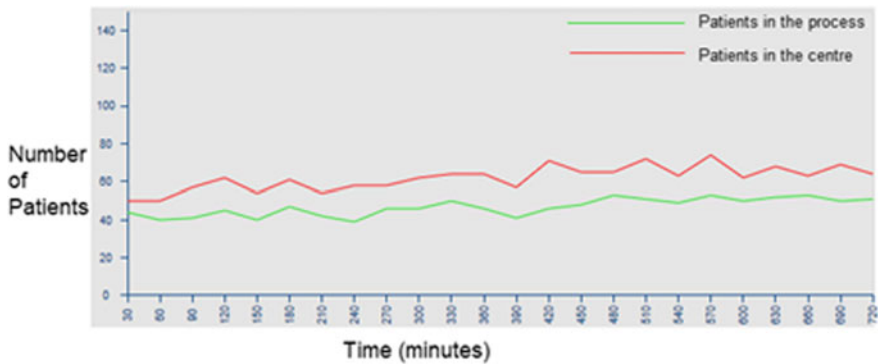


**Fig. 7** Patients in system with an increase in vaccination cubicles

the same parameter settings for 100 replications, shown as scenario 1 in Table 3. The results of the base model have shown relatively small variation but still demonstrate that desired service levels were not achieved. To experiment further we arbitrarily selected three controlling parameters: Patients per hour, Vaccination capacity, and Carpark capacity and explored their sensitivity on the patient output. These are detailed as scenarios 2 to 12 in Table 3. The results show that a reduction in arrival rates to 90 patients per hour yields good patient output and achieve the desired service levels with no changes. This is quite important as there is no costs associated with this change. However, for improved patient output, each scenario was run for 100 replications and average results were collated. The results for scenario 4 (Table 3) indicate a 7% improved patient output is possible by increasing the vaccination capacity by an additional 2 staff and achieving a productive level of service.

**Table 3** Experimentation results

| | Scenario Name | Parameter | | | Response | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | PatientsPerHour.Value | Vaccination.Capacity | CarPark.Capacity | Avg time in centre | Max time in centre | Refused entry due to carpark full | Patient output |
| 1 | Scenario 1 | 100 | 8 | 50 | 54.092 | 76.145 | 46.590 | 1053.360 |
| 2 | Scenario 2 | 90 | 8 | 50 | 35.587 | 48.728 | 0.000 | 1024.860 |
| 3 | Scenario 3 | 80 | 8 | 50 | 33.554 | 46.837 | 0.000 | 916.270 |
| 4 | Scenario 4 | 100 | 10 | 50 | 40.137 | 53.571 | 0.270 | 1124.030 |
| 5 | Scenario 5 | 90 | 10 | 50 | 34.067 | 46.400 | 0.000 | 1028.980 |
| 6 | Scenario 6 | 80 | 10 | 50 | 33.216 | 46.526 | 0.000 | 917.620 |
| 7 | Scenario 7 | 100 | 8 | 60 | 54.072 | 76.728 | 46.700 | 1053.530 |
| 8 | Scenario 8 | 90 | 8 | 60 | 35.587 | 48.728 | 0.000 | 1024.860 |
| 9 | Scenario 9 | 80 | 8 | 60 | 33.554 | 46.837 | 0.000 | 916.270 |
| 10 | Scenario 10 | 100 | 10 | 60 | 40.137 | 53.565 | 0.270 | 1124.000 |
| 11 | Scenario 11 | 90 | 10 | 60 | 34.067 | 46.400 | 0.000 | 1028.980 |
| 12 | Scenario 12 | 80 | 10 | 60 | 33.216 | 46.526 | 0.000 | 917.620 |

# 5 Discussions and Conclusions

We introduced a simulation model for evaluating a Covid-19 vaccination centre. Such tools can help with enhancing the service level and operational performance of such a facility. Witness Horizon simulation software was used to develop the model partially because it offers a higher degree of functionality, ease of use and visualization (2D/3D) capability over other available tools.

The results presented in Fig. 3 represent a single iteration of the simulation. The constraints set using observation data from the VC. The results of our single simulation produced were rather different to those observed due to stochastic elements. Some experimentation was conducted with 100 iterations resulting in the identification of the most appropriate patient arrival rate (hourly) that would achieve a satisfactory service level. Table 3, displays a set of average results when a combination of stepped changes are made to three of the critical parameters. The average values for patient flow time, max time in centre and number of patients refused entry (due to carpark capacity) are shown. Our results show that under a reduced hourly arrival rate, e.g., 90, patient congestion reduces enabling patient service levels to be achieved.

The model developed provides some insights regarding different vaccination centre parameters.

in terms of patient arrival rate (hourly), staff levels, queue capacities, vaccination cubicles and car park capacity.

Our work contributes to the literature in applying DES to Covid-19 vaccination centres at an operational level. Management of vaccination centres can use this simulation model to explore What If's for example the total number of people that can be vaccinated in a given number of days. In addition, the model can help in estimating how many vaccination centres would be needed to achieve a certain level of immunizations over a certain amount of time. Our work has shown that a DES model can effectively model patient flow through a vaccination centre with real-world constraints.

It is important to remember parameters like patient arrival rates do affect the results significantly [18]. For the base model we used 100 patients arriving per hour, however, this arrival rate can be made to vary during the day depending on numerous factors. It is evident that such an arrival rate will have a serious impact on the daily totals of vaccinations. It has been assumed that pre-registration has been done and all patients have pre-booked appointment slots. Previous studies, state that the registration stage is often the basis of bottlenecks in mass vaccination centres.

Observation data and published data were utilised to drive the simulation model [16]. One limit of this work that is worthy of further work is to consider patient and staff behavioural and user needs, e.g., patient's need for washrooms, patients who need further recovery time and to allow patients to leave without vaccination.

Observational data was the basis of this study which has inherent limitations. For improvements, actual patient arrival rates and patient flow times requires a real-world implementation. Such efforts, however, require financial support. As such, this is another limitation of the current study.

# References

1. DeRoo SS, Pudalov NJ, Fu LY (2020) Planning for a COVID-19 vaccination program. JAMA 323:2458–2459
2. New York Times, Coronavirus Vaccine Tracker (2021) https://www.nytimes.com/interactive/2020/science/coronavirus-vaccine-tracker.html Accessed on 12 Nov 2021
3. Schwartz B, Wortley P (2006) Mass vaccination for annual and pandemic influenza. In: Plotkin SA (ed). Springer, Berlin, Germany, pp 131–152
4. Kaplan EH, Craft DL, Wein LM (2002) Emergency response to a smallpox attack: the case for mass vaccination. Proc Natl Acad Sci USA 99:10935–10940
5. Gupta A, Evans GW, Heragu SS (2013) Simulation and optimization modeling for drive-through mass vaccination—a generalized approach. Simul Model Pract Theory 37:99–106
6. National Coronavirus Response: A Road Map to Reopening (2021) https://www.aei.org/researchproducts/report/national-coronavirus-response-a-road-map-to-reopening
7. Mustafee N, Katsaliaki K (2020) Classification of the existing knowledge base of or/ms research and practice (1990–2019) using a proposed classification scheme. Comp Operat Res 118:1–17. https://doi.org/10.1016/j.cor.2020.104920
8. Brailsford SC, Eldabi T, Kunc M, Mustafee N, Osorio AF (2019) Hybrid simulation modelling in operational research: a state-of-the-art review. Eur J Oper Res 278:721–737
9. Lee BY, Brown ST, Cooley PC, Zimmerman RK, Wheaton WD, Zimmer SM, Grefenstette JJ, Assi TM, Furphy TJ, Wagener DK, Burke DS (2010) A computer simulation of employee vaccination to mitigate an influenza epidemic. Am J Prev Med 38(3):247–257. https://doi.org/10.1016/j.amepre.2009.11.009
10. Crowe S, Utley M, Walker G, Grove P, Pagel C (2011) A model to evaluate mass vaccination against pneumococcus as a countermeasure against pandemic influenza. Vaccine 29(31):5065–5077. https://doi.org/10.1016/j.vaccine.2011.04.034
11. Lee EK, Pietz F, Benecke B, Mason J, Burel G (2013) Advancing public health and medical preparedness with operations research. Interfaces 43(1):79–98. https://doi.org/10.1287/inte.2013.0676
12. Van de Kracht T, Heragu SS (2020) Lessons from modeling and running the world's largest drive-through mass vaccination clinic. INFORMS J Appl Anal. https://doi.org/10.1287/inte.2020.1063
13. Pitt M, Monks T, Crowe S, Vasilakis C (2016) Systems modelling and simulation in health service design, delivery and decision making. BMJ Qual Saf 25(1):38–45. https://doi.org/10.1136/bmjqs-2015-004430
14. Asgary A, Najafabadi MM, Karsseboom R, Wu J (2020) A drive-through simulation tool for mass vaccination during COVID-19 pandemic. InHealthcare 8(4):469. Multidisciplinary Digital Publishing Institute https://doi.org/10.3390/healthcare8040469

15. NHS England, UK (2021) COVID-19 vaccination centres: operating framework. Version 1.1. https://www.england.nhs.uk/coronavirus/wp-content/uploads/sites/52/2021/01/C1034-operating-framework-information-and-guidance-on-operating-vaccination-centres-v1.1-20-january-21.pdf. Accessed 12 Nov 2021
16. The Strategy Unit, NHS, UK (2010) Strategy unit releases opensource model for planning vaccine centre capacity. https://www.strategyhunitwm.nhs.uk/news/strategy-unit-releases-opensource-model-planning-vaccine-centre-capacity Accessed 12 Nov 2021
17. Zhang X (2018) Application of discrete event simulation in health care: a systematic review. BMC Health Serv Res 18:687
18. Hassan I, Bahalkeh E, Yih Y (2020) Evaluating intensive care unit admission and discharge policies using a discrete event simulation model. Simul Trans Soc Model Simul Int 96(6):501–518

# Elevated CNN Based Secured Sensor Image Data Communication for HAR: IIOT

**P. Alli** [ID] **and J. Dinesh Peter** [ID]

## 1   Introduction

With the advancement in our day to day life, everything becomes smart such as smart communication, smart systems, and smart services and so on. In simple, our environment becomes smart, all are possible with the development of the Internet of Things (IoT) [1], which provides a facility to connect anyone to any device or any other person with the smart IoT devices [2]. It provides a way to the development of Industry, which incorporates physical manufacturing resources along with virtual system resources, and opens another manufacturing era [3].

With this advancement, the working place also becomes smart with smart monitoring devices to monitor the production as well as employee activity. In this case, the Human Activity Recognition (HAR), which uses time-series sensor data to infer actions and has gained a lot of attention due to its appeal in several application fields, has gotten more attention [4]. Specifically, the industrial IoT application domain covers a wide scope of associated "things", projects both inside and outside the factory [5]. For instance, inside numerous IoT-based processing mechanization and control projects incorporate complete smart industrial facility arrangements with various components, for example, creation of floor monitoring, wearable and augmented reality on the shop-floor, remote control systems, or robotized quality control systems [6].

Software Defined Networking (SDN) is a paradigm that enables simpler and faster innovation for network architectures. It differs from traditional networks where control functions are defined in hardware within the nodes, as it defines control functions in software [7]. This shift allows for a simplified and programmable approach

P. Alli (✉)
Velammal College of Engineering and Technology, Madurai, India
e-mail: allirajendran92@gmail.com; alli_rajus@vcet.ac.in

J. Dinesh Peter
Karunya Institute of Technology and Sciences, Coimbatore, India

to networking, enabling the rearrangement of the virtual network topology in real-time. This research aims to make use of the benefits of SDN to facilitate encrypted data transmission. The remainder of the article is sorted out as follows. Section 2 contains related works. Section 3 gives a detailed depiction of our proposed approach. Section 4 presents the outcomes of our model. Finally, we finish up the paper in Sect. 5 followed by the references.

## 2 Related Works

Castro et al. [8] proposed a Human Activity Recognition model based on IoT, which employs machine learning algorithms such as PCA and Naïve Bayes to recognize the following four activities: lie, sit, walk and jog. The experimentation was done using the smartphone with only a few number of users as well as features and obtained a classification success rate of 95.83%.

Zhang et al. [9] also presented a Human Activity Recognition model based on IoT. For better feature extraction and selection, it combines multihead convolution neural networks with attention processes. Wireless Sensor Data Mining (WSDM) simulation results on the dataset obtain a high recognition rate of 96.4%. In the future, a new dataset with various real-time activities is planned.

Hassan et al. [10] proposed a smartphone inertial sensor-based HAR based on deep learning techniques. The Deep Belief Network recognises human actions using Kernel Principal Component Analysis (KPCA) and Linear Discriminant Analysis (LDA). The average recognition rate of 89.61% and overall accuracy of 95.85% were achieved by the imitation outcomes. In addition, it is planned to concentrate on more strong characteristics and learning in order to recognize more complicated activities in real-time contexts.

Jain et al. [11] presented a descriptor-based method to classify the daily activities captured through smartphone sensors (UCI HAR dataset). It extracts the additional signals and features to apply the information fusion and classification for the effective activity recognition. It obtains a classification accuracy of 97.12% on 1st dataset and 96.83% on 2nd dataset with a minimum number of activities and signals.

Khan et al. [12] suggested a permissioned private blockchain-based approach to encrypt and secure the image. The cryptographic pixel estimates of a picture are stored on the blockchain in this strategy, ensuring the image information's protection and security. The proposed technique is quite effective at preventing data leakage and ensuring security, as seen by encrypted results. After offloading, security for photos on the cloud might be considered.

## 3   Proposed Methodology

### 3.1   Data Collection

Initially, raw information is collected from smart ID card sensors which are stored in the inbuilt data storage for human activity recognition using a motion sensor. This smart ID card camera embedded with accelerometer and gyroscope helps for capturing the human activity, the below Eq. (1) shows collected sensor data with different attributes.

$$D = p_1 + q_2 + r_3 + .... \tag{1}$$

Equation (1) can be expressed through three vectors p represents human motion image, q indicates sensor recorded employee activity and r represents the sound signal detected from human speech, body vibration, etc. that is collected with the help of the above proposed sensor.

### 3.2   Secure Sensor Data Communication Using SDN-SHODHS

As IIOT is becoming more famous in the implementation of the smart industry, the dimensions of the network is regularly extending, which leads to a greater complicated structure and feature of IIOT. Especially, information communication of a huge-scale verbal exchange community tends to bring about high stress on IIOT. Due to confined traffic bandwidth, the network is much more likely to have troubles which include congestion, latency, and so forth.

**Step 1:** The sensor data are collected as per Eq. (1), which represents the p,q and r. With respect to the block length s and the overlapping sizes tx and ty, it is divided into arbitrary blocks. The total number of overlapping blocks is D and it uses a pseudo-arbitrary generator to generate irregular numbers, which is controlled by the key k1. These arbitrary numbers are sorted and stored in an array p [N total], which is used to select the encoding image request. The detailed description is given below.

DualHelix scans are divided into two halves, as seen in Fig. 1. To begin, a randomly chosen starting place is chosen. Then, using a spiral scan, we visit block pixels from the start-point to generate a pixel sequence based on the order of visits, as illustrated in Fig. 1a. Similarly, as illustrated in Fig. 1b, we visit block pixels from the start-point with another spiral scan and acquire a new pixel sequence based on the visit order. To produce a new pixel sequence, we concatenate the first and second pixel sequences together. Finally, we may build the encrypted image block by filling it one pixel at a time from left to right and top to bottom using the new sequence. It's worth noting that each block pixel is only scanned once.
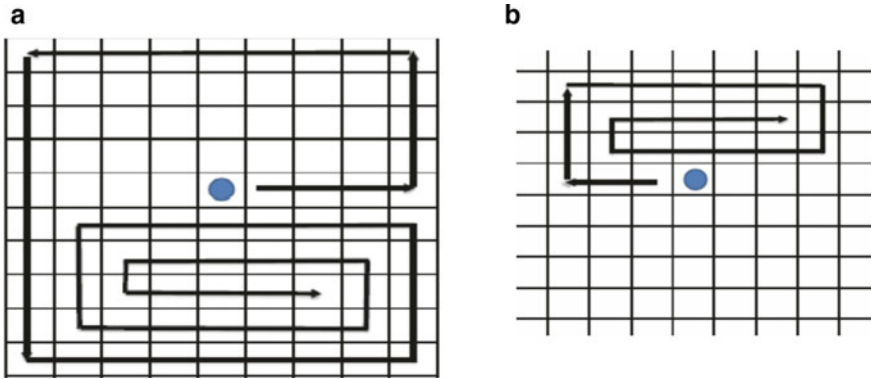
**Fig. 1**  **a** Initial scanning direction. **b** Dual scanning direction

**Step 2:** Let us keep in mind using honey encryption to encrypt the statistics, the communication message area consists of picture, audio, video, and so forth. Those 4 messages are taken care of alphabetically and allow us to count on that 4/8 protection and communication networks the seed area is a three-digit area. In keeping with the ones probability facts, we map those four messages to 4 levels in the seed area. When encrypting message m, the DTE randomly selects a seed from the corresponding seed variety. The relevant thing is XORed with the seed, and the cipher textual content is formed. However, in order to provide additional security, we use double spiral scans and write the encrypted result on the image. From $I = 1$ to N total, the behaviour block encryption repeatedly. Regardless of how the blocks are handled, an overlapping photo j can be obtained.

**Step 3:** For decryption, the code text is XORed with the way to accomplish the seed. At that point, the DTE conversely maps the seed to the original plaintext message. Inside the encryption system, a message ought to have a few planning choices and the planning is directional and irregular. Notwithstanding, in seeing that we type plaintext messages inside the message region and decide the seed assortment through the PDF and CDF of each message, it might be ensured that the seed levels are composed in the indistinguishable request and the total likelihood of the seed assortment in the seed zone is indistinguishable from the aggregate possibility of the message in the message region. Along these lines, we set up a reverse work area that incorporates mappings of the combined chance to the plaintext message. Finding the seed, we will decide on the seed assortment. Finding the seed assortment, we can choose the aggregate cumulative probability shared by utilizing the seed assortment and the relating plaintext message. At that point by methods for looking into the combined chance in the reverse table, we can find the special plaintext message and the code text is decoded.

## 3.3 *Human Activity Recognition Utilizing Elevated Deep Convolutional Neural Network*

After securing the information conversation, our one of the objective is to discover human activity based on combined sensor information. But, there exists widespread demanding situations that would influence the performance of the popularity gadget in realistic scenarios. Many system mastering strategies have been hired in human activity recognition. Nonetheless, there are numerous technological obstacles in this field. The difficulty with feature extraction is the main problem. For sensor-based activity recognition, activity recognition is essentially a classification task. Because there is inter-hobby similarity, feature extraction is more difficult. For starters, activity patterns are unique to each individual. Customers that are exceptional may engage in a variety of activities. Second, it is unrealistic to expect customers to maintain their activity patterns over time. Third, many sensor devices are haphazardly placed on human bodies or in settings. All three aspects contribute to variation in sensory input for activity recognition and must be addressed immediately.

### 3.3.1  Heterogeneous Statistical Feature Extraction for HAR Detection

Deep Learning (DL) Statistics is a safe method for analyzing, classifying, and predicting data that can be used to convey important information. As a result, DL statistics generates more detailed and high-quality insights from the perspective of human activity reputation, generating a set of features from a user's individual body movement.

   The time and frequency location statistical function is extracted from three axial accelerometer and gyroscope data during the feature extraction phase. The statistical properties of a smart ID card-captured image provide useful information, such as the total, mean, desired deviation, and variance of the pixel values. To extract the most statistics from the data, a set of base capabilities was applied to the data, which included both time domain and frequency statistical factors.

### 3.3.2  Feature Selection for Identifying HAR

All capabilities are not equivalently contributed in activity classification. To keep away from the overfitting due to extraneous dimensions, we've got carried out a clever function selection procedure to identify the crucial functions. The precise mastery method selects the great performance function subset from the training set, which can then be applied to the checking out set. This study used a hybrid method of feature extraction to reduce the dimensionality of the original feature space, which resulted in better classification final results.

   The DCNN is used to detect human activity using the established optimal feature sets. d is the sensor's input vector, l is the layer index, σ is the activation function,

$g_j^1$ is the jth characteristic map's time period, and m is the kernel/filter out size. In addition, the output of the lth convolutional layer may be calculated as follows using Eq. (2)

$$h_i^{1,j} = \sigma\left(g_j^1 + \sum_{m=1}^{m} c_m^{1,j} d_{i+m}^{0,j}\right) \tag{2}$$

The pooling layer uses $h_i^{1,j}$ to calculate a summary statistic of neighboring outputs. The pooling procedure utilized in this study, max-pooling, is defined as outputting the highest price among a set of close inputs, as determined by a set of inputs. In our proposed framework the sensor captured data with dissimilar size, pixel size and properties were securely communicated over the Internet with the help of SDN-SHODHS. The features from this transmitted data for each activity are extracted by utilizing our proposed hybrid feature selection framework. The optimal feature sets are then trained and tested with the elevated deep convolutional neural network to identify the human activity.

## 4   Result and Discussion

The goal of our experiment is to create a model that predicts human activities while also providing secure sensor image communication. The data for the proposed research project is collected from smart factory personnel using sensors (accelerometer and gyroscope) in the smart ID card. Walking, Walking Upstairs, Walking downstairs, and Sitting, Standing, or Laying are among the activities recorded. The experimentation outcomes and performance obtained were described in the upcoming subsections.

### 4.1   Application of IIOT as a Smart Factory

Industrial IoT may connect machines, equipment, and sensors on the shop floor, providing procedure engineers and managers with much-desired visibility into the manufacturing process. Producers can use this level of visibility to identify bottlenecks, find the source of problems, and improve at a faster rate with this level of visibility. Because there is a risk of incorrect product distribution to the wrong person if there are no sensing devices in place.

## 4.2 Experimental Setup and Data Set Description

The data was acquired from 30 people who used a smart ID card to complete various tasks while using sensors (gyroscope and accelerometer) in various ways. The sensor signals (accelerometer and gyroscope) are communicated over the internet. In order to avoid internal and external attacks, we proposed a cryptographic framework namely SDN based SHODHS which secures the smart factory employee sensor data and machinery information. After secured sensor data communication, data are automatically pre-processed and then the feature extractions are performed with hybrid feature selection framework. Then the features are given to the deep convolutional neural network for the human activity identification.

Figure 2 shows the average analysis of data provided by each user. The number of users is represented on the x-axis, while the number of samples is represented on the y-axis. All of this information was gathered from 30 personnel and divided into two sections for training and testing.

## 4.3 Simulation Result and Performance Analysis

This section describes about the simulation result and performance of our proposed framework. Our work contributes with two motivations one is secure sensor data communication and the other is human activity recognition for privacy and employee monitoring respectively.

Figure 3 shows the comparison of encryption and decryption time representation. The encryption time taken for the cryptographic algorithm is much higher than the
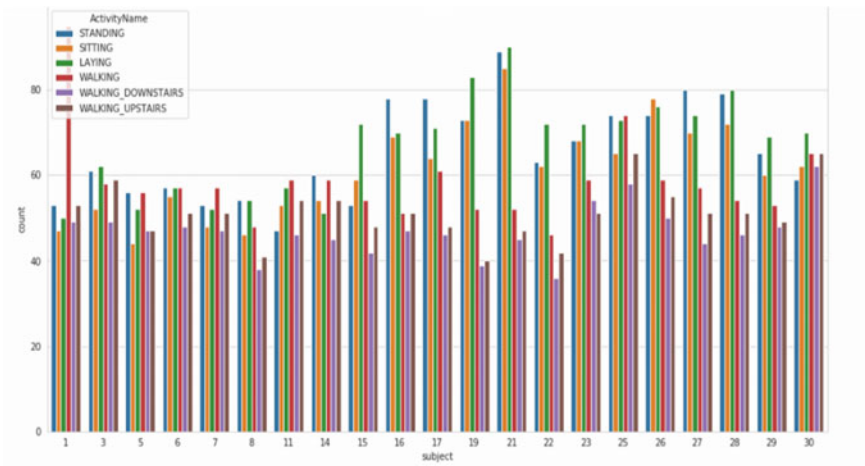


**Fig. 2** Data provided by each user

decryption time, and the time gets reduced in a certain file size and increased with some other file size, this is because of the data heterogeneity like different type, size, color, pixel size, block size and so on.

Figure 4 shows the overall classification report of our proposed human activity recognition framework which is tested over 2947 samples and obtained an accuracy level of 98.5% in terms of recall, precision and f1-score. The unbalanced datasets are not taken into account by the accuracy measure. In this scenario, the accuracy is heavily skewed in favor of the majority groups.

Figure 5 depicts the importance of feature extraction, selection and validation. It shows comparative analysis of the activity recognition with and without feature selection. In all the cases, identification with feature selection outputs an excellent level of human activity identification.
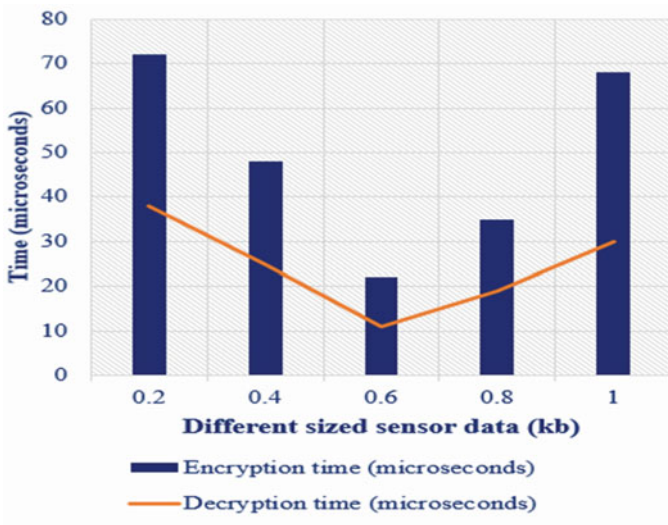


**Fig. 3** Comparison graph of encryption and decryption time

```
-------------------------
| Classifiction Report |
-------------------------
                     precision    recall   f1-score    support

            LAYING        1.00      1.00       1.00        537
           SITTING        0.98      0.97       0.98        491
          STANDING        0.94      0.98       1.00        532
           WALKING        0.99      1.00       0.97        496
WALKING_DOWNSTAIRS        1.00      0.97       0.99        420
  WALKING_UPSTAIRS        0.97      0.97       0.97        471

       avg / total        0.98      0.98       0.98       2947
```

**Fig. 4** Overall classification report with six activities

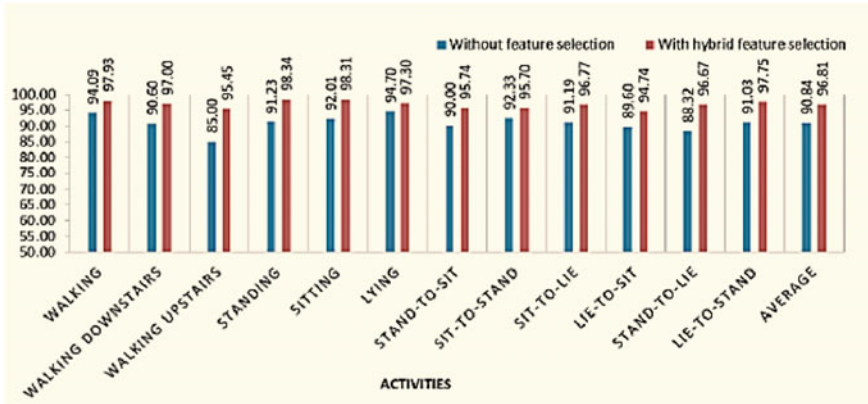**Fig. 5** Overall individual activity accuracies with and without feature selection

## 4.4 Comparison Analysis

This section shows the comparison results of our proposed framework with the existing works for both multimedia data communication and human activity recognition. Figure 6 depicts the human activity comparison with the existing methodologies with different activities. In all the activities our proposed ECNN achieved a highest accuracy ratio. The average accuracy ratio is obtained by our proposed ECNN framework as 98.5%, which is comparatively higher than all other algorithms in the comparison as well as in the related work sections. Because all the other algorithms obtained a maximum value of 96%. Finally, our experimental analysis proved the performance of our secured sensor data communication (image, video and audio etc.) and accurate human activity recognition with high success ratio.
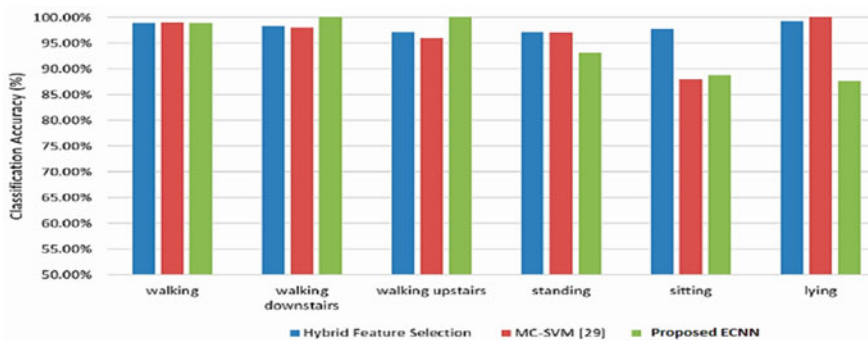


**Fig. 6** Comparison accuracy of different activities for different classification model

# 5 Conclusion

This research work concentrates on the application of IIoT. In this, we focus on the employee monitoring in a smart factory with a smart ID card for data set collection. The monitored data were communicated to the office dashboard over the Internet. In this situation, an effective cryptographic framework called Stochastic Honey Overlapping Based Dual Helix Scan (SAHDHS) is proposed to securely transfer the sensor captured data (image, audio, video and so on). The encrypted data were communicated with the help of Software Defined Networking to avoid the network resource handling, and latency issues. Then a hybrid feature selection mechanism is applied to extract the essential features for the effective human activity identification (employee monitoring). Finally, the optimal features are trained and tested with the help of our proposed ECNN for the identification of human activity. Finally, the experiments attain the accuracy ratio of 98.5%.

# References

1. Vermesan O, Friess P (eds) (2013) Internet of things: converging technologies for smart environments and integrated ecosystems. River publishers
2. Bhide VH, Wagh S (2015) I-learning IoT: an intelligent self learning system for home automation using IoT. In: 2015 International conference on communications and signal processing (ICCSP). pp 1763–1767
3. Nagy J, Oláh J, Erdei E, Máté D, Popp J (2018) The role and impact of Industry 4.0 and the internet of things on the business strategy of the value chain—the case of Hungary. Sustainability 10(10):3491
4. Ronao CA, Cho SB (2016) Human activity recognition with smartphone sensors using deep learning neural networks. Expert Syst Appl 59:235–244
5. Lin J, Yu W, Zhang N, Yang X, Zhang H, Zhao W (2017) A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J 4(5):1125–1142
6. Wang X, Ong SK, Nee AY (2018) A comprehensive survey of ubiquitous manufacturing research. Int J Prod Res 56(1–2):604–628
7. Jain S, Chandrasekaran K (2020) Industrial automation using Internet of Things. In: Security and privacy issues in sensor networks and IoT. IGI Global, pp 28–64
8. Castro D, Coral W, Rodriguez C, Cabra J, Colorado J (2017) Wearable-based human activity recognition using an iot approach. J Sens Actuator Netw 6(4):28
9. Zhang H, Xiao Z, Wang J, Li F, Szczerbicki E (2019) A novel IoT-perceptive human activity recognition (HAR) approach using multihead convolutional attention. IEEE Internet Things J 7(2):1072–1080
10. Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. Futur Gener Comput Syst 81:307–313
11. Jain A, Kanhangad V (2017) Human activity classification in smartphones using accelerometer and gyroscope sensors. IEEE Sens J 18(3):1169–1177
12. Khan PW, Byun Y (2020) A blockchain-based secure image encryption scheme for the industrial Internet of Things. Entropy 22(2):175

# Comparative Studies with Random Datasets Using Enhanced Faster R-CNN, Mask R-CNN, and Single Shot Detector

**M. Karthi, C. Niroshini Infantia, G. Subhashini, and V. Shyam Sundar**

## 1 Introduction

In this modern world, the most difficult and tricky thing is to handle data. In each field there exists an enormous amount of data, managing and monitoring them has become a problematic task. In recent decades, machine learning and deep learning are proving themselves as the best options to handle a large amount of data. The main reason for the essential use of machine learning and deep learning is the cost of handling data and the accuracy of the work done. Machine learning definition states "Machine learning is a field of study that gives the computer an ability to learn by itself instead of being explicitly programmed". Humans are naturally exposed to making errors, hence to reduce the error and reduce time machine learning algorithms are widely used.

Deep learning is a package of machine learning algorithms that plays a vital role in extracting main features and high-level features from a raw input using multiple and deep layers. The usage of hidden layers has great traditional techniques, especially in the case of pattern recognition. Among all Convolutional Neural Network has a unique trademark in deep neural networks. In solving a machine learning problem, the important section is to choose the best algorithm that properly fits that case. There are too many algorithms available and have some of the corresponding drawbacks. Finding the right algorithm for the problem plays a vital role in solving the problem. This project takes a step ahead on the selection of algorithms by implementing three algorithms namely, Faster R- CNN [1], Mask R-CNN [2], and Single Shot Detector [3] on various random datasets and find the best among the three algorithms. But, initially, the implementation process had started with using RCNN [4] algorithm which is a fine algorithm that uses specific linear SVM for detecting the class objects. But in RCNN the time taken is too high to perform object detection on an image. This

M. Karthi (✉) · C. Niroshini Infantia · G. Subhashini · V. Shyam Sundar
Information Technology, St.Joseph's Institute of Technology, Chennai, India
e-mail: mmuthukarthi@gmail.com

is because the training is not done in one single step, there will be different models to perform different parts. So, it becomes time-consuming. Machine learning is mainly known for its lower accuracy and time, so they are the most important parameters to be looked into.

Then, Fast R-CNN [5] came into play which is one step better than RCNN. The fast R-CNN algorithm uses a selective search method to find out the region proposals, which is also a time-consuming and slow task. It also directly affects the overall performance of the algorithm. Next, by extending the search for an efficient machine learning algorithm, Faster R-CNN [1] algorithm was found, and after using that it performed much better than the previous two. In Faster R-CNN, the usage of RPN made the algorithm more efficient and superior. Object detection task includes both localizing an object or objects within an image and classifying it. Mask R-CNN is an extended type of Faster R- CNN. Mask R-CNN [2] is a futuristic architecture in the field of image segmentation and instance segmentation with Convolutional Neural Network (CNN), introduced in the year 2018. Mask R-CNN, which has a Deep Neural Network detects objects in the image and provides the finest segmentation mask for each prototype.

To try a single convolutional network, the Single Shot Detector [3] architecture will predict the location of the bounding boxes and classify those locations in just one pass. Hence, it can be trained end-to-end easily. While using SSD, we had to just take one single shot to make the detection of multiple objects in an image, whereas, in RPN (Region Proposal Network), we need two shots for this process, one for generating region proposals and one for detecting the class objects within the image. Thus, SSD remains faster than other RPN-based approaches. Further, eight datasets are used, namely Airplane, Ambulance, Bicycle, Fire hydrant, Lighthouse, Monitor, Mouse, and Parking meter. This paper is about the implementation and recording of the performance of the above-mentioned algorithms [1–3] with random datasets. Section 2 explains the reason for the selection of these algorithms.

## 2   Related Work

Deep learning is a branch of machine learning which has a set of algorithms and is known for its structure and function of the brain called artificial neural networks. In addition, deep neural networks also have certain capabilities, they have the ability of automatic features extraction from the raw input. This process is called features learning. Deep learning has an excellent ability on problem domains where the inputs (and even output) are analog. Meaning, they are not a few quantities in a tabular format but instead are images of pixel data, documents of text data, or files of audio data. Asymmetric Tri-way (ATF) [6] has been using Faster R- CNN before. It is designed for domain adaptive object detection. Due to parameters sharing, the risk of source domain collapse has occurred. So, to avoid those problems ATF model with Faster R-CNN has been introduced. At the same time, the conventional NMS algorithm has been replaced with soft-NMS [7]. The replacement by soft-NMS minimized the

issue of not detecting the partially hidden, adjacent, or overlapping objects in the image.

Once automatic sensing the objects in remote sensing images had been a difficult task. Then, deep learning methods were introduced to prevent those difficulties. In that Mask R-CNN is one of the algorithms used [8]. Mask R-CNN has used a multi-class object detection algorithm. Then, of course,e Mask R-CNN is also used in digital images. For gathering information from the images or altering the images, Mask R-CNN has been used [9]. It is also used in blind image forgery detection.

Initially, Single Shot Detector (SSD) has a weak ability in detecting small or blurry objects in the images. But then it has been improved by using DenseNet (Dense Convolutional Network) and feature fusion. This is called DF-SSD [10]. Then after years, focusing tightly on improving predicted scores of objects and localizing accuracy, SSD is again enhanced by using certain techniques [11]. It is accomplished by adding a new branch to the model, Intersection Over Union (IoU) prediction loss branch. This then elevated the results of the Single Shot Detector model and made use widely. In final, all three algorithms are very much refined and fine-tuned to be used in real-time. But still, they have some drawbacks in it like increase in important parameters, quantization errors, etc.… So, some of them are processed and implemented here and their performance is recorded.

## 3   Selection of Algorithm

Nowadays, convolutional neural networks are applied and used everywhere. It has become the best and famous deep learning architecture. The recent take-off of the usage of CNN is because of its effective convnets and its immense popularity [12]. Among all CNNs, the starting algorithm, Faster R-CNN [1] has become an outstanding algorithm because of its RPN used for region proposals. The region proposal network takes a feature map and generates the anchors (the center of the sliding window with a unique size and scale). These anchors are then passed into the classification layer (which classifies that there is an object or not) and the regression layer (which localizes the bounding box associated with an object) [13]. The second algorithm is Mask R-CNN [2], this algorithm gives way for image segmentation and instance segmentation [14]. Mask R-CNN is an extended version of Faster R-CNN, it is built on the top surface of Faster R-CNN. The main attraction of exploring Mask R-CNN is because of its stunning feature of generating masks for the detected object. However, image segmentation is the process of splitting a digital image into multiple segments. The main goal of image segmentation is to convert an image to something meaningful and easy to analyze [15]. Mask R-CNN, a Region-Based Convolutional Neural Network, was developed on top of Faster R-CNN. Mask R-CNN was developed in reference using Faster R-CNN architecture. Mask R-CNN has an additional third branch that generates a superior object mask. The additional mask output is distinct from the class and box outputs, requiring the extraction of a much finer spatial layout of an object. The final algorithm is Single Shot Detector

[3]. In the initial, the obvious fact is Single Shot Detector is slightly different from the previous two algorithms, as SSD makes object localization and classification in a single shot. Whereas in all CNN algorithms, two shots take place, one for localization of the object and the other is for detecting the object in the image. SSD has small convolutional filters to predict object classes and offsets to default boundary boxes to improve the performance of the model in real-time [16]. It uses multi-scale feature maps for object detection.

## 4 Experiments

### 4.1 Datasets

In this project, 8 random and different datasets have been chosen for experiment and analysis. All samples in each dataset are taken in different dimensions and different angles of the objects in the samples. The datasets are collected from an open image dataset by google API [17]. Annotations and augmentations are done with respective tools available for all the three corresponding algorithms (like VGG annotator, labeling, etc.,) to be increased the accuracy of the trained models. The number of images allocated for training, testing, and validation are mentioned in Table 1. The airplane dataset comprises all types of aircraft like passenger aircraft, jets, and air force planes. The ambulance dataset is a package of ambulances used in various types from different countries. Because the balance used in different countries varies with its model, color, and siren. The bicycle dataset package has almost all types of bicycles like road bicycles, mountain bicycles, off-road cycles, folded cycles, BMX cycles. Fire hydrants are used for extracting water from pipelines and water distribution systems. It contains all kinds of fire hydrants from small to large.

   The package contains lighthouses of various structures from different countries. Because lighthouse built in different countries varies in their architecture, color, and shape. The monitor dataset comprises almost all types of monitors like LED monitors, OLED monitors, DLP and TFT monitors, and plasma screen monitors. The dataset package contains computer mouse of various types from vintage to modern. A computing mouse is actually of many shapes and types, like trackball mouse, laser mouse, and optical mouse. A parking meter collects money for the use of a parking spot. It contains all kinds of parking meters used in various countries. Then, the three algorithms are implemented in these eight different datasets. The results of every individual data in each algorithm are calculated.

**Table 1** Characteristics of datasets based on validation, test, and validation split

| Algorithm | | Airplane | Ambulance | Bicycle | Fire hydrant | Light house | Monitor | Mouse | Parking meter |
|---|---|---|---|---|---|---|---|---|---|
| Faster RCNN | Training | 210 | 200 | 223 | 210 | 200 | 231 | 200 | 210 |
| | Testing | 45 | 50 | 45 | 45 | 50 | 38 | 50 | 45 |
| | Validation | 45 | 50 | 45 | 45 | 50 | 35 | 50 | 45 |
| Mask RCNN | Training | 175 | 150 | 175 | 175 | 150 | 156 | 150 | 175 |
| | Testing | 38 | 43 | 40 | 38 | 43 | 35 | 43 | 38 |
| | Validation | 38 | 43 | 38 | 38 | 43 | 38 | 43 | 38 |
| SSD | Training | 245 | 225 | 245 | 245 | 225 | 245 | 225 | 245 |
| | Testing | 53 | 63 | 53 | 53 | 63 | 53 | 63 | 53 |
| | Validation | 53 | 53 | 87 | 53 | 53 | 53 | 53 | 53 |

## *4.2  Training*

This work has 3 different algorithms as mentioned above, Faster R-CNN, Mask R-CNN, and Single Shot Detector. Google Colab is used for training and testing purposes, that provides a path to use incredible GPUs and it does not require any setup in the training phase and their results are explained.

### 4.2.1  Faster RCNN

The dataset images are initially annotated using the labelimg [18] tool, used widely for annotating and labeling images in the dataset. Initially in Faster R-CNN, the annotated images and their values are passed into a CNN, here Dense-Net [19] is used and feature maps are extracted from each image. But in Faster RCNN by default Resnet50 is used as CNN, but due to the increase in the inevitable parameters with deepening of the network, many problems may occur to network optimization and experimental hardware. The datasets prepared specifically for **the R (Regression) Classification** and object detection have a very small sample size, so that the training may lead to overfitting. extracting features and it helped to solve all the above-mentioned problems. The Dense-net has a twin architecture of ResNet [20]. The main variation between the two architectures lies in the varying transfer functions for various network blocks as shown in Fig. 1. Dense-Net predefined weights have been used for getting better results. Now, the resultant feature maps are given as input for Region Proposal Network (RPN). The RPN analyzes each feature maps and returns object proposals. The role of Region Of Interest (ROI) that simplifies all the object proposals to the same dimension. These final proposals are then passed into fully connected layers to classify and predict any object(s) present in the image and its corresponding bounding box(s). The total time taken for preprocessing and training phase is 90 min.

### 4.2.2  Mask RCNN

Like Faster R-CNN, here the same Convolutional Neural Network is used for extracting feature maps from the dataset images. But, here ResNet101 [21] architecture has been used as CNN. Followed by same processes takes place as same as Faster R-CNN, feature maps are sent to RPN layer and returns candidate bounding boxes. Then, the ROI pooling layer brings down all the bounding boxes to the same size. In addition, by default, a single-layer feature to align each ROI. However, we turned the single-layer features into multilayer features, because each ROI needs to do ROI alignment operation with multilayer features, and then the ROI features of different layers will be fused so that each ROI feature will have multilayer features [22]. These proposals are then sent to fully connected layers for classification and
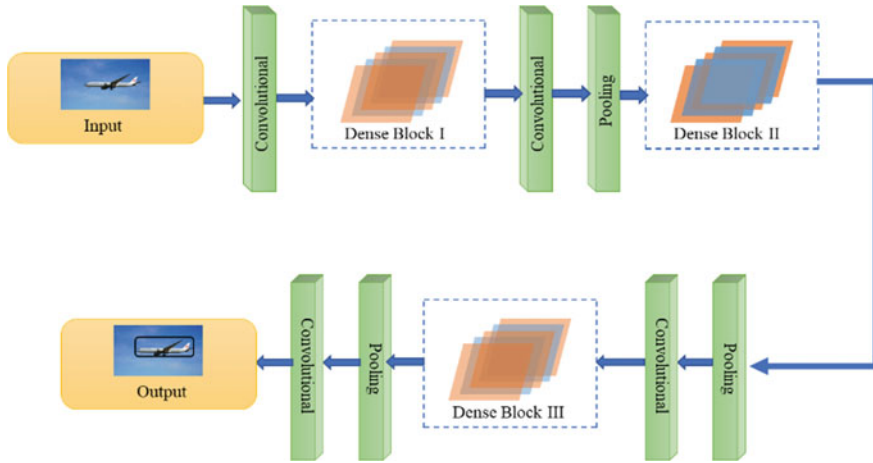
**Fig. 1** Enhanced faster RCNN architecture

regression. Then we get the resultant and finest bounding boxes for objects in the image.

Till this, the steps are as same as Faster R-CNN. But here comes the main role of mask generation of the detected object. For that, we initially calculate the region of interest so that computation time gets reduced. For all the resultant bounding boxes we calculate Intersection over Union (IoU) with the ground truth boxes. We can compute IoU by using the below Eq. (1):

$$IoU = Area\ of\ the\ intersection\ /\ Area\ of\ the\ union \tag{1}$$

Now, the calculated value of IoU is greater than 0.5, we consider that as a region of interest. Otherwise, we will avoid that region. Likewise, this operation is carried out for all the areas and the area with IoU greater than the threshold value will be selected. At last, the mask-generating block is added to the existing architecture as shown in Fig. 2. This returns each segmentation mask in $28 \times 28$ size for objects in the image. It took almost 110 min for the total train a model and test it.

### 4.2.3  SSD

The SSD represented here is built on VGG-16 architecture, but all the fully connected layers are ignored from it. The input to SSD is an image with ground truth boxes for each object in the image. VGG-16 performs the feature extraction operation and produces feature maps as output. Unlike Multibox, the priors are manually chosen, but in Multibox, this process is done so the IoU concerning the ground truth will be above 0.5. SSD should allow generalizing for any input type, without training previously for prior generation, **which is mentioned in this theory**. For instance,
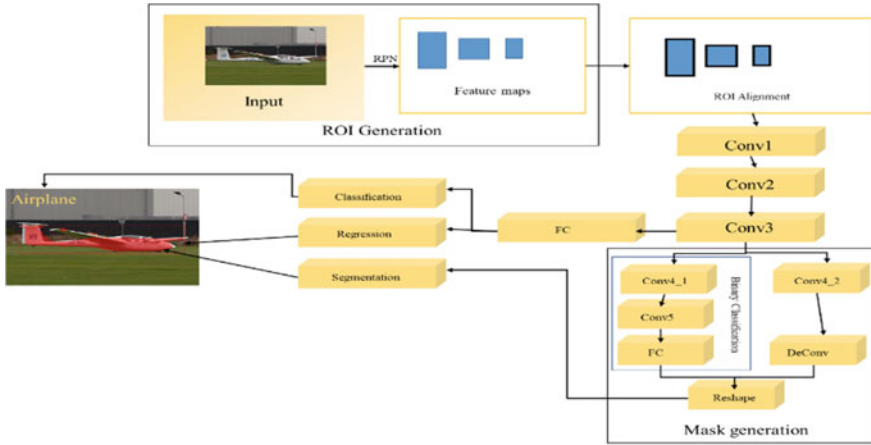
**Fig. 2** Enhanced mask RCNN architecture

assuming we have configured 2 diagonally opposed points (x1, y1) and (x2, y2) for each b default bounding boxes (BB) per feature map cell, and c classes to classify, on a given feature map of size, calculated using Eq. (2)

$$f = m * n, \tag{2}$$

SSD would compute bounding boxes using Eq. (3)

$$BB = f * b * (4 + c) \tag{3}$$

values for this feature map. The next convolutional layers evaluate anchor boxes of variant ratios at each location in several feature maps. SSD has multiple default boxes of various dimensions. It uses 8732 boxes. During training time, the default boxes are matched over aspect ratio, location and scale to the ground truth boxes. We select the boxes with the highest overlap with the ground truth bounding boxes. IoU (intersection over union) between predicted boxes and ground truth should be greater than 0.5. We finally pick up the predicted box with maximum overlap with ground truth as shown in Fig. 3.

## 5 Experimental Results and Analysis

After implementing all the three algorithms, Faster R-CNN, Mask R-CNN, and Single Shot Detector in each dataset separately, all the parameter values are tabulated in Table 2. Annotations and augmentation were done manually in this project using labelimg and VGG image annotator. The overall performance is shown in Fig. 4.
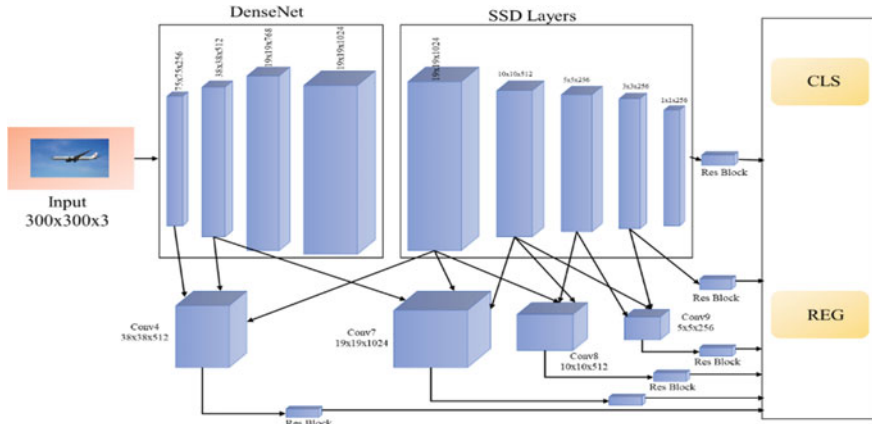
**Fig. 3** Enhanced single shot detector architecture

**Table 2** Comparative performance analysis of faster RCNN and SSD architectures

| Dataset/ Algorithm | Faster RCNN | | | Mask RCNN | | | SSD | | |
|---|---|---|---|---|---|---|---|---|---|
| | Loss | Accuracy | mAP | Loss | Accuracy | mAP | Loss | Accuracy | mAP |
| Airplane | 45.12 | 78.32 | 74.21 | 30.95 | 91.35 | 86.92 | 30.01 | 87.03 | 82.21 |
| Ambulance | 38.21 | 92.53 | 87.92 | 27.12 | 94.50 | 90.43 | 28.92 | 89.10 | 84.43 |
| Bicycle | 33.93 | 89.83 | 83.55 | 29.76 | 92.61 | 90.23 | 31.43 | 86.22 | 81.23 |
| Fire hydrant | 42.43 | 79.28 | 75.27 | 27.32 | 94.20 | 89.27 | 29.13 | 88.20 | 83.27 |
| Lighthouse | 29.59 | 91.73 | 86.12 | 25.84 | 95.00 | 91.00 | 33.71 | 85.36 | 80.12 |
| Monitor | 38.68 | 80.75 | 75.92 | 31.04 | 91.22 | 86.62 | 32.93 | 85.99 | 80.92 |
| Mouse | 27.33 | 89.42 | 84.23 | 37.45 | 85.03 | 80.23 | 35.22 | 83.12 | 78.23 |
| Parking meter | 26.28 | 85.50 | 79.43 | 34.26 | 88.22 | 83.12 | 23.95 | 94.53 | 89.00 |

In general, all the objects in each image will not be in the same dimensions, size, and angle. In the case of tiny objects or partially visible objects present in an image sample, Single Shot Detector works efficiently. This results in an accuracy of 94.53% and a mAP value of 89%. Therefore, this may conclude that Single Shot Detector is efficient in detecting small objects in an image sample. On the other hand, the presence of the RPN layer and anchor boxes made Faster R-CNN effective to find objects in different angles and aspect ratios. The ambulance dataset will be the best example for it. As the object in this case travels, the angle of the Vehicle varies on a large scale, in this perspective, Faster R- CNN made fine predictions with an accuracy of 92.53% and mAP of 87.92%. Almost every image sample has multiple objects in it, so there may be many chances for objects misclassification and due to plenty of objects, most of the objects may be partially visible or blurry, while using Mask
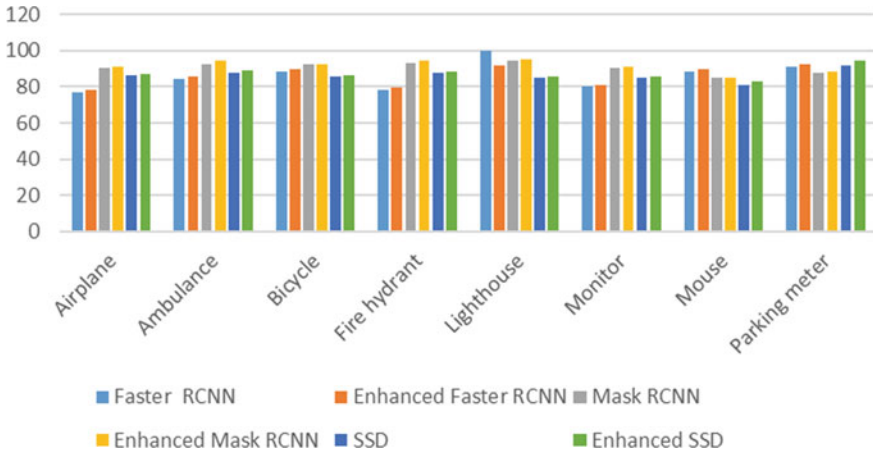
**Fig. 4** Overall performance of models

R-CNN this problem in the image sample is minimized and the overall accuracy is higher. Mask R-CNN has the highest accuracy of 95% in the lighthouse dataset with 91% mAP. As the lighthouse is present outdoor and it's very small in images, there may be many objects around it, but there is no misclassification in it. So, finally, in minimizing object misclassification, detecting tiny objects, and recognizing objects in various angles, Mask R-CNN is proved as the most efficient among the other three algorithms tested.

## 6   Inference and Findings

The main drawback in default Faster R-CNN is the CNN used, ResNet50 that resulted in deepening of the network with increase in the unavoidable parameters and resulted in model overfitting. Hence, DenseNet is used as an alternate CNN in enhanced Faster R-CNN that worked efficiently and also gave better results than ResNet50. In default Mask R-CNN, in ROI, single-layer feature is used but it took lots of time. So, it is turned into multilayer features in enhanced Mask R-CNN, because each ROI needs to do ROI alignment operation with multilayer features. Then the ROI features of different layers will be fused so that each ROI feature will have multilayer features. In Multibox used in default Single Shot Detector, the priors are chosen concerning the ground truth boxes with IoU above 0.5. But SSD should permit any input type without getting trained before. So, the priors are manually chosen in enhanced Single Shot Detector. At last, the predicted box with maximum overlap with ground truth is considered.

# 7 Conclusion

The three enhanced algorithms Faster R-CNN, Mask R-CNN, and Single Shot Detector have been implemented and tested on eight different and customized datasets. In considering small objects to be detected in test images, such as the lighthouse, enhanced Single Shot Detector has a better performance. However, with minimal misclassification of objects and detection of objects in various angles, such as ambulance, enhanced Mask R-CNN got better results. Compared with formal Faster R-CNN, Mask R-CNN, and Single Shot Detector, enhanced Faster R-CNN has 1.6% average improvement in mAP, enhanced Mask R-CNN has 2.7% average improvement in mAP and enhanced SSD has 1.9% average improvement in mAP and has given best solution. In an overall view, enhanced Mask R-CNN has been the best algorithm with low misclassification, detecting small objects and in various dimensions.

# References

1. Ren S et al. (2015) Faster r-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015): 91–99
2. He K, Gkioxari G, Dollár P, and Girshick R (2017) Mask r-CNN. In: Proceedings of the IEEE international conference on computer vision. pp 2961–2969
3. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC (2016) SSD: Single shot multibox detector. In: European conference on computer vision. Springer, Cham, pp 21–37
4. Raj A, Namboodiri VP, Tuytelaars T (2015) Subspace alignment-based domain adaptation for rcnn detector. arXiv preprint arXiv:1507.05578
5. Girshick R (2015) Fast r-CNN. In: Proceedings of the IEEE international conference on computer vision. pp 1440–1448
6. He Z, Zhang L (2020) Domain adaptive object detection via asymmetric tri-way faster-rcnn. In: Computer Vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16. Springer International Publishing
7. Zhang N, Feng Y, Lee E-J (2021) Activity object detection based on improved faster R-CNN. J Korea Multimed Soc 24(3):416–422
8. Mahmoud A, et al. (2020) Object detection using adaptive mask RCNN in optical remote sensing images. Int J Intell Eng Syst 13(1): 65−76.https://www.robots.ox.ac.uk/~vgg/software/via/
9. Ahmed B, Gulliver TA, alZahir S (2020) Image splicing detection using mask-RCNN. Signal, Image and Video Processing 14(5):1035–1042
10. Zhai S et al. (2020) DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion. IEEE Access 8:24344–24357
11. Li J et al. (2020) SSD object detection model based on multi-frequency feature theory. IEEE Access 8: 82294–82305
12. Sharif Razavian A et al. (2014) CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops
13. Mo X et al. (2018) An efficient approach for polyps detection in endoscopic videos based on faster R-CNN." In: 2018 24th international conference on pattern recognition (ICPR). IEEE
14. Zuo L et al. (2020) A robust approach to reading recognition of pointer meters based on improved mask-RCNN. Neurocomputing 388:90–101

15. Zhang Q, Chang X, Bian SB (2020). Vehicle- damage-detection segmentation algorithm based on improved mask RCNN. IEEE Access 8:6997–7004
16. Liu X et al. (2017) Optimizing read and write performance based on deep understanding of SSD. In: 2017 3rd IEEE international conference on computer and communications (ICCC). IEEE
17. https://storage.googleapis.com/openimages/web/download.html
18. https://github.com/tzutalin/labelImg
19. 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
20. Gladence LM, Karthi M, Maria Anu V (2015) A statistical comparison of logistic regression and different Bayes classification methods for machine learning. ARPN J Eng Appl Sci 10(14):5947–5953
21. Karthi M, Priscilla R, Syed Jafer K (2020) A novel content detection approach for handwritten english letters. Procedia Comput Sci 172:1016–1025
22. Shu JH, Nian FD, Yu MH, Li X (2020) An improved mask R-CNN model for multiorgan segmentation. Math Probl Eng

# Machine Learning Forecasting for Optimisation of Green Energy Generation in Non-domestic Buildings

**Connor Scott** and **Alhussain Albarbar**

## 1 Literature Survey

Large businesses consume 45% of non-domestic building energy [1] and the UK has plans to become the world leader in green energy [2]. Renewable energy (RE) cannot be considered a reliable source [3] but can provide a more reliable energy mix [4] with the promotion of RE by world leaders improving application [5]. 28 percent of energy generation in the UK is connected at the distribution level [6] which shows there is clear room for increased decentralised RE. RE generation contributes 17.18GW which is 43% of the UK's total energy demand [7], with 4 percent solar and 31% wind due to the 6500 wind turbines installed across the country. This shows the usability of various RE sources to contribute to the whole demand.

### 1.1 Photovoltaic Generation

Two types of solar energy available are thermal and photovoltaic (PV). They can provide more benefits than each other when used correctly, depending on the climate [8], with the integration of methods showing results too [9]. Solar thermal uses the heat from the sun's energy to run a generator whereas PV uses photons to generate electricity [10]. The application of solar technologies is just as important as generation as [11] shows various uses to aid building systems in reducing energy demand.

C. Scott (✉) · A. Albarbar
Manchester Metropolitan University, Manchester M15 6BH, UK
e-mail: connor_scotty@hotmail.co.uk

## 1.2   Renewable Energy Generation

Many methods of energy generation exist within branches of major methods. Kinetic floor tiles are capable of powering low-consumption electrical devices in [12] through 520mW per tile compression. The rotational energy from manual rotating doors can be harvested to produce 40 W per 180° rotation or 331 W per minute of constant use [13]. Wind energy can be produced in various ways such as installation on a high ridge. A model is 13 m long and produces between 4 and 12.5 kW of energy depending on wind speed [14] but any wind turbine cannot be too big as they produce too much noise and affect the comfort conditions of the building [15]. Biomass is a method that uses organic matter to produce energy which can be implemented in decentralised buildings [16].

## 1.3   Machine Learning

A machine learning algorithm (MLA) is an algorithm capable of forming mathematical bonds from inputs to outputs of a given dataset. Then when new data is added, the bonds remain, and the outputs can be predicted [17]. When forecasting a buildings energy demand, NN's provide the highest performance [18], with 7 types of data processing, 30 types of feature selection, and 128 types of MLAs used for prediction of a buildings energy demand [19]. Various activation functions allow for different amounts and types of data to be used, each using various computational power allowing for different accuracies [20]. Backpropagation involves working backwards through the created model to reduce the error of each input by altering the given weights. There are various ways of doing this with the Levenberg–Marquardt (LM) showing significant benefits over other types [21, 22]. ML forecasting is used to develop a V2G strategy, saving between 35 and 65% of total energy consumption [23]. The prediction of all building characteristics can be accurately predicted through MLAs without a vast knowledge on the subject. Instead, the MLA uses collected data to predict the characteristics while using less computational power than mathematical calculations. Some algorithms still require large computational power, but applications in the field of BMS forecasting remain usable.

## 1.4   Building Management Systems

BMSs are an increasingly researched topic that should consider all aspects of a building including physical models, environmental conditions, comfort, safety, occupants' preferences, thermal, and visual specifications [24]. The accuracy of prediction techniques and local climate conditions are critical for an optimisation system [25] with BMS's reducing annual energy consumption by an average of 16% [26] and

saving costs in the range of 11% [27]. Combination of a BMS and smart-design features can lower cooling by up to 97%, and energy consumption by up to 49% [28]. ML techniques for BMSs consider five main functions: observe, predict, adjust, manage, and interact [29].

The paper is presented in four parts: the method, results, discussion, and conclusion. The method explains the MLA, the BMS, and the data collection, processing, and prediction of PV generation and energy demand. The results show the accuracy of the created MLA with applicability to the BMS. The discussion and conclusion explain the results and further work in greater detail.

## 2 Methodology of Energy Generation Optimisation

The proposed MLA is used to forecast installed PV generation with validation of the model. The results of the MLA are applied to a case study building to optimise energy use, through storage, use, and trading through a BMS. The MLA is trained using previously collected weather data and solar generation that corresponds with that day. The MLA is a three inputs, two layers, one output feedforward neural network with a hyperbolic tangent activation function and Levenberg–Marquardt backpropagation training method in use. Processing of training data consists of removing wrong and missing data while characterising data samples through an RF out-of-bag classifier to determine which inputs contain the most importance. The BMS works through a binary system, as when the demand is at peak time, the energy from the PV generation storage will be used instead of purchasing from the grid. Then, when demand is off peak, the PV generation can be collected and stored in on-site storage until needed. If predicted PV doesn't match the predicted energy demand, the energy can be purchased from the grid for storage at a cheaper price or generated through other on-site RE methods until the demand is matched. The benefit of forecasting is that the BMS calculates how much energy needs to be generated or purchased at a cheaper rate before the day has started, allowing system optimisation. The BMS method is shown in Fig. 1.

The predicted PV and energy demand of the building is evaluated with the aim for PV to provide the same or more energy than the building is using. If this is true the building can rely on just PV. If this is false, then the energy storage is evaluated, with the aim for the energy storage and generation to provide the same or more energy than the demand. If this is true on-site storage can be used. If this is false too, because it is known before the tariff rates have changed, the required energy to match the demand can be purchased at an off peak rate and can be stored until required.

The accuracy of forecasted solar generation is verified by comparing it with actual solar generation data of the same day. Once verified, the generation can be compared to the energy demand of the building to show how an optimal BMS would work through storage or using generated energy to regulate the electricity demand from the grid. The results are verified against 13 days of real collected data. The method for creating a suitable MLA is shown in Fig. 2:
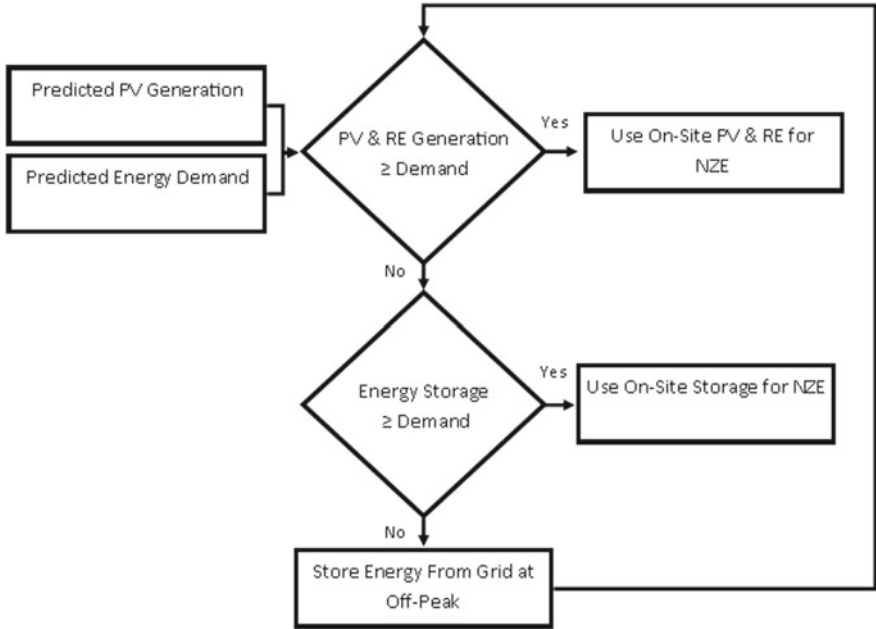
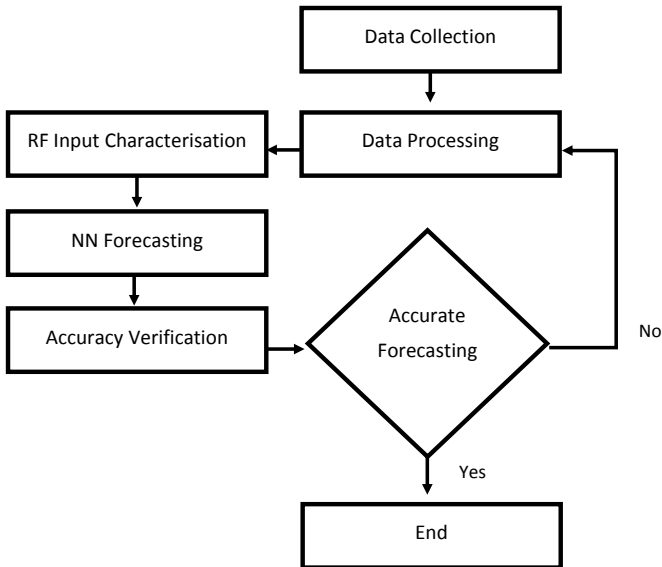**Fig. 1** The proposed methodology for the BMS to follow for energy optimisation



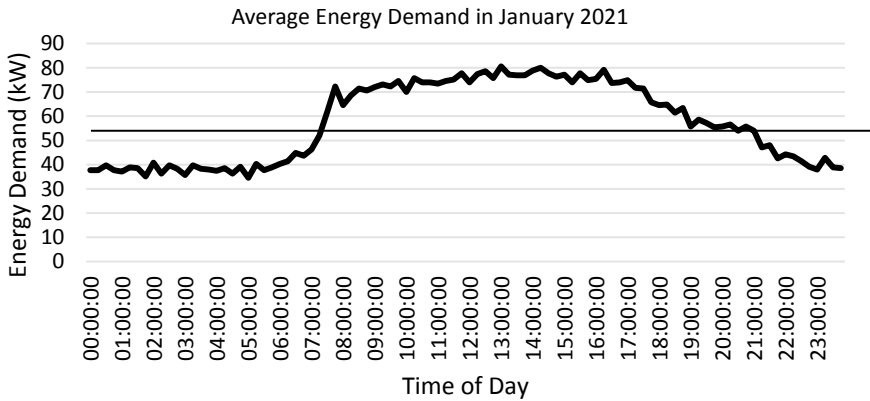**Fig. 2** The method of creating an MLA suitable for forecasting

**Fig. 3** The average energy demand of the MMU Business School for January 2021 from midnight to midnight, measured in 15-min increments

The collected data is in daily increments for outdoor temperature, cloud coverage (%), hours of sunlight, radiation (W/m2), rain (mm), and wind. This is then sent through to processing where clearly wrong data (extreme highs and lows), or any missing data for the day is removed along with the rest of the samples for that day. The cleaned data is sent through an RF characterisation algorithm which randomises the inputs but keeps the number the same. The error is then measured with each sample being out of place and the inputs with the highest effect on the output can be determined. The processed data is then added to the NN which forecasts the output which is verified by comparing it with collected data.

Daily demand is shown in Fig. 3.

The bar in the middle of the graph at 57.47 kW represents the difference between 'peak time' and 'off peak'. Off peak times are anything below 57.47 kW and peak times are above so ideally, energy is generated/stored at below average and used/sold at above average. Peak time is from 7:30 until 18:45.

Figure 4 is shown.

## 3 Results

Results from currently installed PV generation, PV generation to cover peak time, PV generation with installed storage capacity, and decentralised BMS capable of generating all necessary energy from on-site RE are shown below as well as prediction accuracy.

The forecasted PV generation compared to the actual PV generation is shown in Fig. 5.

The predicted PV generation is used to show the effectiveness of the installed system, and how improvements can reduce the buildings emissions. Results from the
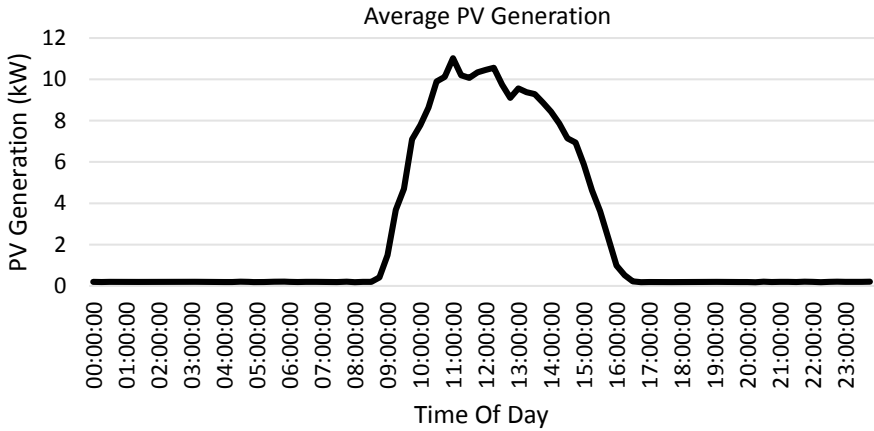
**Fig. 4** The average daily solar PV generation in January from midnight to midnight in 15-min increments. Peak PV generation is around 11:00 with generation in the bounds of 9:00–16:30
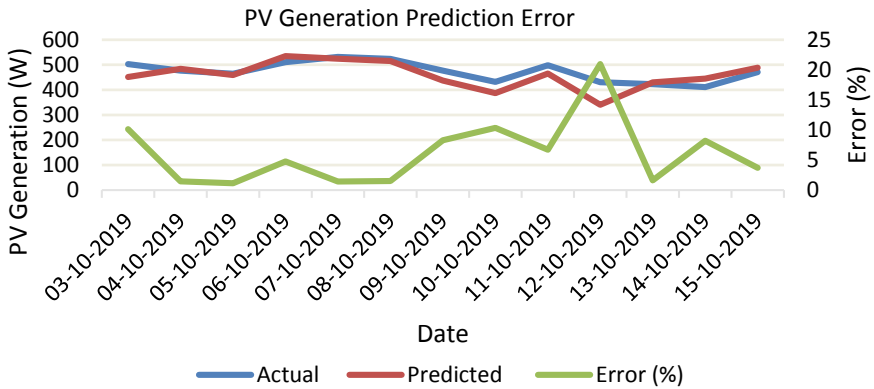


**Fig. 5** The daily predicted PV generation was verified against the actual. The error has a peak of 20.95% and a low of 1.1% with an average of 6.17%. This error is explained through the sudden change in input characteristics as the radiation levels decrease dramatically for that day while cloud coverage increases but solar PV is still recorded to have a high output

BMS algorithm show that currently installed PV generates 4.2% of overall demand with results shown in Table 1.

The first method uses the currently installed PV panels on the building to show the benefit of having just one RE generation installed. The second method shows a theoretical improvement on installed PV or RE generation that if it can produce enough energy to cover the peak time load, off peak energy can be bought for cheaper. The third method shows the financial benefits of having only storage and no RE generation. The energy can be bought and stored at off peak rates before the day, using the predicted demand, and can be used instead of buying it when the rates are

**Table 1** Showing the results for the four methods of energy optimisation

| Improvement | Direct PV | Peak time PV | Storage | Decentralised BMS |
|---|---|---|---|---|
| Energy (kW) | 233.3 kW | 3,146.4 kW | 0 kW | 5,518 kW |
| Profit (£) | £31.50 | £471.96 | £47.19 | £792.13 |
| Kg/Co$_2$e | 54.35 | 733.11 | 0 | 1285.69 |

high. The decentralised BMS allows the prediction of on-site renewables to match the predicted energy demand of the building, allowing the building to sell any extra to the grid at any tariff rate, because the prediction for the next day is already known. The improvement in Co$_2$ is recorded to show environmental improvements with each method.

## 4 Discussion

More RE sources produce a more dynamic system as wind, solar, kinetic, and biomass are independent of each other meaning when one isn't producing, another one almost certainly will be. Storage should be avoided when possible because it adds extra cost to the system for the initial purchase of the storage or efficiency losses, depending on the method of storage. Larger buildings usually have a higher energy demand, but also have more space to instal REs so they come hand in hand. An actual optimised BMS usually wouldn't be completely net zero as a large building as the purchase and installation of REs is expensive. Instead, the building can aim for coverage of peak time demand with REs which can be determined using the NN prediction process.

Energy demand can be predicted using the same NN process as proposed in the paper which completes the optimised BMS as REs can be predicted and compared against energy demand prediction to evaluate the optimisation route. The 6.17% error on prediction can be considered when analysing outcomes of BMS applications which still provide good and usable results. The reason for the 6.17% error is the volatility of the collected data. The weather data for the prediction day with the highest error has very low radiation levels with high cloud coverage. The algorithm predicted a low PV output, but it was still relatively high, meaning more varied data or more data is needed to create a more accurate algorithm. The prediction with the lowest error has very average data, meaning none of the data was high or low, so the algorithm had seen more of that data before and trained it better.

## 5  Conclusion

A feedforward neural network with a hyperbolic tangent activation function and a Levenberg–Marquardt backpropagation method is proposed and created using collected weather data surrounding the MMU Business School. The NN was able to predict the installed solar photovoltaic panels output with an average of 6.17% error which was verified by comparing prediction results against collected data. Four scenarios are explained which show the benefits of increased renewable energy generation with storage methods, allowing an optimised BMS. The results from the MLA are applied to a building management system. Firstly predicting the energy demand and on-site renewable energy production, then using prediction results to optimise, trade, storage, and use of the energy.

## 6  Further Work

Further work could include more research on prediction technologies to would benefit the application of neural networks in this use as a higher accuracy leads to better results. Prediction and application of other renewable energy sources will improve the method as only PV could be verified, but collected data allows the application of other energies.

## References

1. (2018) Business energy statistical summary. [Online] Available https://www.gov.uk/govern ment/publications/business-energy-statistical-summary
2. (2020) New plans to make UK world leader in green energy. [Online] Available https://www. gov.uk/government/news/new-plans-to-make-uk-world-leader-in-green-energy
3. Ahmed SU, Ali A, Memon A (2018) Renewable energy's reliability issue and possible solutions: A meta-analytic review. J Inf Commun Technol
4. Goggin M (2017) Renewable energy builds a more reliable and resilient electricity mix. American Wind Energy Association 01/05/2017
5. Lu Y, Khan ZA, Alvarez-Alvarado MS, Zhang Y, Huang Z, Imran M (2020) A critical review of sustainable energy policies for the promotion of renewable energy sources. Sustai 12 (12). https://doi.org/10.3390/su12125078
6. Grid N (2021) Annual report and accounts 2020/2021. National Grid, 2021
7. Stolworthy M (2021) GB fuel type power generation production. GridWatch. https://gridwa tch.co.uk/. Accessed 26 Nov 2021
8. Valančius R, Černeckienė J, Vaičiūnas J, Jurelionis A, Fokaides P (2018) Solar thermal systems versus Photovoltaic systems. Case study: Single family building in Lithuania, pp 1–6
9. Joshi SS, Dhoble AS (2018) Photovoltaic–Thermal systems (PVT): Technology review and future trends. Renew Sustain Energy Rev 92:848–882. 01 Sep 2018. https://doi.org/10.1016/j. rser.2018.04.067
10. Reddy S, Mallick T, Chemisana D (2013) Solar power generation. Int J Photoenergy. https://doi.org/10.1155/2013/950564

11. Tzivanidis C, Bellos E (2018) Solar energy utilization in buildings. pp 119–165
12. Jintanawan T, Phanomchoeng G, Suwankawin S, Kreepoke P, Chetchatree P, U-viengchai C (2020) Design of kinetic-energy harvesting floors. Energies 13 (20). https://doi.org/10.3390/en13205419
13. Partridge JS, Bucknall RWG (2018) Potential for harvesting electrical energy from swing and revolving door use. Cogent Eng 5 (1):1458435. https://doi.org/10.1080/23311916.2018.1458435
14. The power collective. RB2 Commercial. https://thepowercollective.ca/products/rb2/. Accessed 28 Nov 2021
15. Bošnjaković M (2013) Wind power buildings integration. J Mech Eng Autom 3:221–226. https://doi.org/10.17265/2159-5275/2013.04.005
16. Khan S, Paliwal V, Pandey V, Kumar V (2015) Biomass as renewable energy
17. Georga EI, Fotiadis DI, Tigas SK (2018) 6–Nonlinear models of glucose concentration. In: Georga EI, Fotiadis DI, Tigas SK (eds) Personalized predictive modeling in type 1 diabetes. Academic Press, pp. 131–151
18. Walker S, Khan W, Katic K, Maassen W, Zeiler W (2020) Accuracy of different machine learning algorithms and added-value of predicting aggregated-level energy performance of commercial buildings. Energy Build (209):109705. https://doi.org/10.1016/j.enbuild.2019.109705
19. Zhang et al. L (2021) A review of machine learning in building load prediction. Appl Energy 285:116452. https://doi.org/10.1016/j.apenergy.2021.116452
20. De Ryck T, Lanthaler S, Mishra S (2021) On the approximation of functions by tanh neural networks. Neural Netw 143:732–750. https://doi.org/10.1016/j.neunet.2021.08.015
21. Pulido M, Melin P, Castillo O (2014) Particle swarm optimization of ensemble neural networks with fuzzy aggregation for time series prediction of the Mexican stock exchange. Inf Sci 280:188–204. https://doi.org/10.1016/j.ins.2014.05.006
22. Sánchez D, Melin P, Castillo O (2020) Chapter 12–Modular granular neural network optimization using the firefly algorithm applied to time series prediction. In: Yang XS (ed) Nature-inspired computation and swarm intelligence. Academic Press, pp 199–216
23. Scott C, Ahsan M, Albarbar A (2021) Machine learning based vehicle to grid strategy for improving the energy performance of public buildings. Sustain 13(7). https://doi.org/10.3390/su13074003
24. Eini R, Linkous L, Zohrabi N, Abdelwahed S (2021) Smart building management system: Performance specifications and design requirements. J Build Eng (39):102222. https://doi.org/10.1016/j.jobe.2021.102222
25. Mariano-Hernández D, Hernández-Callejo L, Zorita-Lamadrid A, Duque-Pérez O, Santos García F (2021) A review of strategies for building energy management system: Model predictive control, demand side management, optimization, and fault detect & diagnosis. J Build Eng 33:101692. https://doi.org/10.1016/j.jobe.2020.101692
26. Chaouch H, Çeken C, Arı S (2021) Energy management of HVAC systems in smart buildings by using fuzzy logic and M2M communication. J Build Eng 44:102606. Accessed 12 Jan 2021. https://doi.org/10.1016/j.jobe.2021.102606.
27. Farinis GK, Kanellos FD (2021) Integrated energy management system for Microgrids of building prosumers. Electr Power Syst Res 198:107357. https://doi.org/10.1016/j.epsr.2021.107357
28. Salerno I, Anjos MF, McKinnon K, Gómez-Herrera JA (2021) Adaptable energy management system for smart buildings. J Build Eng 44:102748. https://doi.org/10.1016/j.jobe.2021.102748
29. Alanne K, Sierla S (2022) An overview of machine learning applications for smart buildings. Sustain Cities Soc 76:103445. 1 Jan 2022. https://doi.org/10.1016/j.scs.2021.103445