Internet of Things

Joel J. P. C. Rodrigues
Parul Agarwal
Kavita Khanna *Editors*

IoT for Sustainable Smart Cities and Society

Springer

Internet of Things

Technology, Communications and Computing

Series Editors
Giancarlo Fortino, Rende (CS), Italy
Antonio Liotta, Edinburgh Napier University, School of Computing, Edinburgh, UK

The series Internet of Things - Technologies, Communications and Computing publishes new developments and advances in the various areas of the different facets of the Internet of Things. The intent is to cover technology (smart devices, wireless sensors, systems), communications (networks and protocols) and computing (theory, middleware and applications) of the Internet of Things, as embedded in the fields of engineering, computer science, life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in the Internet of Things research and development area, spanning the areas of wireless sensor networks, autonomic networking, network protocol, agent-based computing, artificial intelligence, self organizing systems, multi-sensor data fusion, smart objects, and hybrid intelligent systems.
** Indexing: *Internet of Things* is covered by Scopus and Ei-Compendex **

More information about this series at http://link.springer.com/series/11636

Joel J. P. C. Rodrigues • Parul Agarwal
Kavita Khanna
Editors

IoT for Sustainable Smart Cities and Society

Springer

Editors
Joel J. P. C. Rodrigues
College of Computer Science and Technology
China University of Petroleum (East China)
Qingdao, China

Instituto de Telecomunicações
Covilhã, Portugal

Kavita Khanna
Campus Director
Delhi Skill and Entrepreneurship University
New Delhi, India

Parul Agarwal
Department of Computer Science and Engineering
Jamia Hamdard
New Delhi, India

ISSN 2199-1073 ISSN 2199-1081 (electronic)
Internet of Things
ISBN 978-3-030-89553-2 ISBN 978-3-030-89554-9 (eBook)
https://doi.org/10.1007/978-3-030-89554-9

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The title of the book is *IoT for Sustainable Smart Cities and Society* under the book series Internet of Things.

This book focuses on "**smart**" and "**intelligent**" technologies which have penetrated our lives and are driving force for the future. This forms the base for the fact that the entire world is looking for computational techniques for their applications in smart domains. IoT and its enabling technologies, which include AI, machine learning, deep learning, cloud computing, and big data analytics, have the potential to provide solutions to the global challenges faced by smart cities. At the same time, it is realized that IoT-driven technology innovations should be resilient enough to secure the future transport, education, healthcare, energy, home, water, and waste. This book provides a sound theoretical base and an extensive practical expansion which enables us to understand IoT-driven solutions in smart cities.

With a good understanding of IoT and smart cities and the associated communication protocols, this book provides an insight into its applications in several areas of smart cities, thus giving a holistic understanding of the topic. Models and architectures that provide solutions have been the key focus areas. The main challenges associated with IoT and the connected network are security, privacy, and authenticity, and the same has also been covered extensively in this book. A well-illustrated knowledge base forms an important feature of any book. This book serves as a knowledge-sharing platform of ideas focusing on current and emerging IoT enabled smart cities.

Book Contents

This book ***comprises 14 chapters***, designed to capture the core ideas of IoT and smart cities for sustainability, and are organized as follows:

Chapter 1: Role of Machine Learning and Deep Learning in Internet of Things enabled Smart Cities

Tarana Singh, Arun Solanki, Sanjay Kumar Sharma

This chapter introduces the latest developments, classifications, application areas, and services of machine learning (ML), deep learning (DL), and IoT in the field of smart cities. A rigorous review on IoT-enabled cities using ML and DL has been presented, highlighting that there are various approaches in ML and DL which, when used in IoT, results in a powerful solution for the implementation of smart city applications.

Chapter 2: Understanding New Age of Intelligent Video Surveillance and Deeper Analysis on Deep Learning Techniques for Object Tracking

Preeti Nagrath, Narina Thakur, Rachna Jain, Dharmender Saini, Nitika Sharma, Jude Hemanth

This chapter is intended at understanding intelligent surveillance methods through deep learning–based multi-object tracking. Various object tracking approaches along with the object detection classifiers used in tracking have been shared in detail. A comparative study is tabulated on different algorithms on state-of-the-art multi-object tracking challenge datasets and is summarized with the future prospects of object tracking. It also includes a case study of deployment of drone technology powered by IoT in different realms in a smart city.

Chapter 3: Tech to Take Care: IoT-Based Smart Solution for Real-Time Supervision

Srishti Sharma, Virendra Pratap Singh

This chapter proposes a dependable, real-time smart supervision system for basic supervision tasks of infants, senior citizens, and the physically disabled. By the aid of the Internet of Things and Thingspeak cloud, a solution is recommended that records and tracks numerous diverse vitals of babies, the elderly, and the incapacitated. This also includes the tracking of their routine, communicating weekly analysis of the same, and generating crucial warnings to the custodians' mobile from anywhere in alarming situations.

Chapter 4: IoT in Healthcare – A 360-Degree View

Rishika Mehta, Kavita Khanna, Jyoti Sahni

It discusses the IoT architecture including the various sensors and technologies related to data communication, storage, analytics, and visualization that have enabled an unbounded expansion of IoT-based healthcare services. This work describes different operational areas of healthcare including disease monitoring, age-based monitoring, physical abnormality monitoring, and work profile–based monitoring. It also discusses the role of cloud as well as fog for processing a large amount of data collected through an IoT ecosystem and their associated security aspects. It presents the related work and research gaps in each of these areas, thus providing a comprehensive survey to till-date research in the field.

Chapter 5: Industrial IoT and its Applications

Jyotsana Grover

This chapter focusses on Industrial IOT (IIOT) and its applications in the food industry, manufacturing industries, and the healthcare. It presents the challenges faced by all these industries and how these challenges are addressed using various IOT applications. It further highlights that the specific requirements with a particular industry is basically mostly dealt with in the application level, and at the device level, more or less the concepts remain similar; thus, in all the different IIoT applications, the core technological ideas remain the same, the only thing that changes is basically the type of sensors that would be used.

Chapter 6: An Interactive Analysis Platform for Bus Movement: A Case Study of One of the World's Largest Annual Gathering

Emad Felemban, Faizan Ur Rehman

This chapter presents a unique interactive platform that can track the movement of all the buses during the Hajj and provide numerous solutions in the process. The prototype studies the movement, routes, parking spaces, establishments, offices, travel times, speeds, and violations in the Mashaer area at different and multiple times during the day for the entire period of the Hajj pilgrimage. The platform is fit for storing, analysing, and visualizing big spatial data and could be very helpful for all stakeholders, organizers, and decision-makers working in the area of intelligent transport systems.

Chapter 7: Vehicle Payload Monitoring System

Nishant Yadav, Nishita Yadav, Anjali Garg

A prototype hardware model is developed for the payload monitoring system which will be helpful in resolving the problem of vehicles unable to reach weighing stations to weight their cargo owing to long lines. The proposed model has numerous applications both for the driver or the owner of the vehicle and the government in implementing various regulations. The proposed system saves time and money to weigh the load of the vehicle which is required during transportation and load balancing thereby saving many accidents. The application of the proposed system can be extended further while designing bridges and flyovers, wherein the measurement of weight is an important component to maintain the life of the system in which it is implemented.

Chapter 8: Implementation and Comparison of MQTT, WebSocket, and HTTP Protocols for Smart Room IoT Application in Node-RED

Simran Kaur, Vandana Khanna

This chapter gives an overview of IoT Architecture, different components of IoT, along with some application areas. IoT-based smart room application has been implemented and presented in detail. The hardware and software setup used in experimentation is also briefed. The response time delays of all the IOT protocols like MQTT, HTTP, and WebSocket for the smart room test bed have been found and compared along with the comparisons with the existing literature and future prospects.

Chapter 9: Comparative Study of Static and Hybrid Analysis Using Machine Learning and Artificial Intelligence in Smart Cities

Shagil Chaudhary, Ramesh Amgai, Shouvik Das Gupta, Nida Iftekhar, Sherin Zafar, Anil Kumar Mahto

Malware penetration is getting more regrettable day by day and is considered as one of the greatest security dangers to the Web. This chapter examines the two widespread methods, static and dynamic methods, that are utilized to viably perform malware examination and location on venture frameworks to diminish the harm of malware assaults. It is concluded that static and hybrid analysis when done together provides a much better understanding of how malware is developed and works.

Chapter 10: Automated Weather Monitoring Station Based on IoT for Smart Cities

Shaifali M. Arora, Mishti Gautam

Climate monitoring is a crucial process since rapid differences in atmospheric condition cause impact on diverse community, budgetary, and substantial aspects along with safeness, fitness, nutrition consumption, and travel. Hence, acquisition of appropriate weather data is necessary to produce correct end result. This chapter details these parameters and implements a smart, real-time, efficient, low-cost, accurate, low-power, portable, and high-speed IoT-based automated weather monitoring station.

Chapter 11: Energy Harvesting for Sustainability

Parul Agarwal, M. Afshar Alam, Sheikh Mohammad Idrees, Ajay Vikram Singh, Joel J.P. C. Rodrigues

This chapter presents a significant focus on energy and energy-harvesting methods that rely on solar, thermal, kinetic, and radio frequency. The implications and reasons for harvesting energy have been discussed. It concludes that energy harvesting is one of the crucial aspects which needs to be widely adopted in order to save the environment and achieve sustainability. It covers the harvesting techniques and how the benefits can be derived for smart cities.

Chapter 12: A Review of Machine Learning Models in Renewable Energy

Anuj Gupta, Kapil Gupta, Sumit Saroha

This chapter examines the machine learning methods for forecasting renewable energy resources and demonstrates the process of implementing the machine learning models, and analyses the renewable energy forecasting models based on mean absolute percentage error and correlation coefficient. Finally, several possible future work opportunities have been identified.

Chapter 13: Security and Privacy Issues in IoT-Enabled Smart Cities

Aditya Sam Koshy, Nida Fatima, Parul Agarwal, Joel J.P. C. Rodrigues

IoT devices embrace novel endless opportunities, thus making life easier for people, but they also increase the risk of data breaches, unsuspected users, and malicious attacks on IoT framework. So, preserving security and privacy from threats and attacks are endowing challenges that are faced by IoT devices. This chapter presents major security issues and discusses smart city applications with their threats and solutions precisely to preserve concealment of Internet of Things (IoT) devices. Privacy and security issues along with threats and attacks have been highlighted.

Chapter 14: Efficacy of Bio-absorbent Concept in Textile Effluent Treatment Technology Using Low-Cost Materials by Implementing Banana Bark and Orange Peel

Arivoli. A, Agnello J. Naveen

The elimination of dye colour from the industrial wastewater assimilation process gives a classification of treatment technology, mainly if the adsorbent is cost beneficial and naturally available. The current chapter has given an insight on removal of colour dyes in the effluent using bio-adsorption material such as orange skin (peel) and banana bark. The experimental outcomes have shown that the efficacy of the materials have the superior capability to eliminate colour from wastewater and acts as a cost-cutting adsorbent material**.**

Hope you delve deeper into the varied aspects of the book and enjoy reading and learning about IoT-enabled smart cities and aspects of sustainability. Authors form an integral part of a book. We acknowledge the reviewers for their comments, as it helped us to improve the quality of the book. Gratitude to the publisher, Springer, and all other people involved directly or indirectly in the completion of this book. We dedicate this book to our family members without whose support this volume was not possible. Last but not the least, our gratitude to God, for showing us the light to start this project and blessing us at every step from its conceptualization to its implementation.

Qingdao, China — Joel J. P. C. Rodrigues
Covilhã, Portugal
New Delhi, India — Parul Agarwal
New Delhi, India — Kavita Khanna
30 August 2021

Contents

Chapter 1
Role of Machine Learning and Deep Learning in Internet of Things enabled Smart Cities

Tarana Singh, Arun Solanki, and Sanjay Kumar Sharma

Abstract Nowadays, smart city is the latest research domain that is continuously attracting new researchers in its different domains like smart transportations, smart grids, and smart education. It is a very well-known fact that the world's population is growing at an extraordinary rate today, and not only does half of the population live in urban areas, but it is also estimated to rise by 50% by the year 2050. The amount of population living in these megapolis, thus, puts enormous strain on the environment, which needs to be managed smartly, and thankfully, smart technologies like the Internet of Things (IoT) combined with machine learning (ML) and deep learning (DL) have the potential to tame the pressures of urbanization by creating new and smarter experiences for making day-to-day living more comfortable. The concept of IoT has always been considered the key infrastructure in smart cities since its introduction, and in this chapter, we aim to explain the role that IoT and ML, as well as DL, play in smart cities. There are various approaches in ML and DL that, when it comes to IoT, will result in a powerful solution for the implementation of smart city applications. This chapter provides a rigorous review on IoT-enabled cities using ML and DL.

Keywords Smart city · Machine learning · Deep learning · Internet of Things · Big data

1 Introduction

Due to the rapid development of global urbanization, a large world population will flow into cities. This trend will raise the challenges for land utilization management, sustainable urban development, food supply, safety, security, and overall human

T. Singh (✉) · A. Solanki · S. K. Sharma
Department of Computer Science and Engineering, Gautam Buddha University, Greater Noida, Uttar Pradesh, India
e-mail: sanjay.sharma@gbu.ac.in

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*, Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_1

Fig. 1.1 Smart city and components

well-being. Emerging technologies and new smart city concepts are very important. In the city of the future, everyone pledges to encourage a better future [7]. As shown in Fig. 1.1, artificial intelligence applications such as the Internet of Things (IoT), machine learning (ML), deep learning (DL), and big data are important to support technology and smart city development. This method contributes to a variety of applications such as liquidity management and monitoring, urban planning, resource allocation, energy supply and demand, food supply and production forecasting, air pollution monitoring, and forecasting [2].

AI, ML, and DL models, alongside the Internet of Things, are turning out to be progressively indispensable pieces of different businesses. Right now, these technologies are moving toward smart city planning. Its motivation is to computerize and push the exercises and general activities of nearby governments. At the point when a city is, for the most part, thought to be a smart city, it implies the city is running a type of IoT and AI methods to gather information from various focuses. Smart urban areas have an assortment of utilization cases dependent on AI and IoT innovation, from keeping a better climate to advancing public vehicles and security [6]. By utilizing AI, AI calculations, and the IoT, urban communities plan better smart transportation arrangements by permitting inhabitants to move starting with one spot and then onto the next in a protected and effective way. AI gathers information from various angles and gives it to a focal worker for additional execution. When the information is gathered, it ought to be utilized to make the city smarter.

2 Role of IoT in Smart City

The world is changing quickly because of the impact of data innovation. Progressively dynamic connectable gadgets are changing businesses and the metropolitan climate everyone is acclimated with. Data innovation and IoT gadgets have been coordinated into the metropolitan framework, prompting the rise and improvement of the "smart city" idea. There is no interesting and precise meaning of the term smart city. A few scientists characterize the sustainability commission as: "The Sustainability Commission is a city that utilizes advanced technology to improve the personal satisfaction of its residents, and deal with its metropolitan foundation and environment." Despite the Covid episode, individuals proceed to work and move to visit urban communities for a variety of reasons, including professional openings and ways of life [6]. At the point when urban communities face population thickness-related issues, numerous issues can emerge, including air contamination, absence of freshwater, a lot of trash, and expanded traffic [26]. The Internet of Things assumes a significant part in taking care of these issues. Figure 1.2 shows some IoT administrations in smart urban areas. To outline the part of the Internet of Things in smart urban areas, here is a portion of the IoT applications in smart urban communities:

- **Smart Infrastructure:** Computerized innovation is getting progressively significant as urban communities have the conditions for the manageable turn of events. Structures and metropolitan foundations should be arranged in a more powerful and maintainable manner. Additionally, we need to invest resources into electric and self-driving vehicles to keep carbon dioxide outflows low [14]. Indeed, brilliant innovation can be energy saving and harmless to the ecosystem framework. For instance, to decrease power interest, brilliant lights possibly sparkle when somebody passes by them. Changing brilliance levels and following everyday utilization are significant pieces of smart lights.
- **Air Quality Management:** Smart urban communities likewise carry out gadgets that can gather constant contamination information and foresee productions. Having the option to precisely foresee air contamination permits urban areas to distinguish the reasons for outflow issues and formulate methodologies for restricting the measure of air contamination [29].
- **Traffic Management:** Perhaps the greatest test confronting huge urban areas is discovering approaches to improve traffic. Be that as it may, discovering an answer is not inconceivable. For instance, Los Angeles is perhaps the busiest city on the planet, carrying out smart transportation answers to control the traffic stream. On-street sensors send ongoing traffic stream updates to focal traffic at

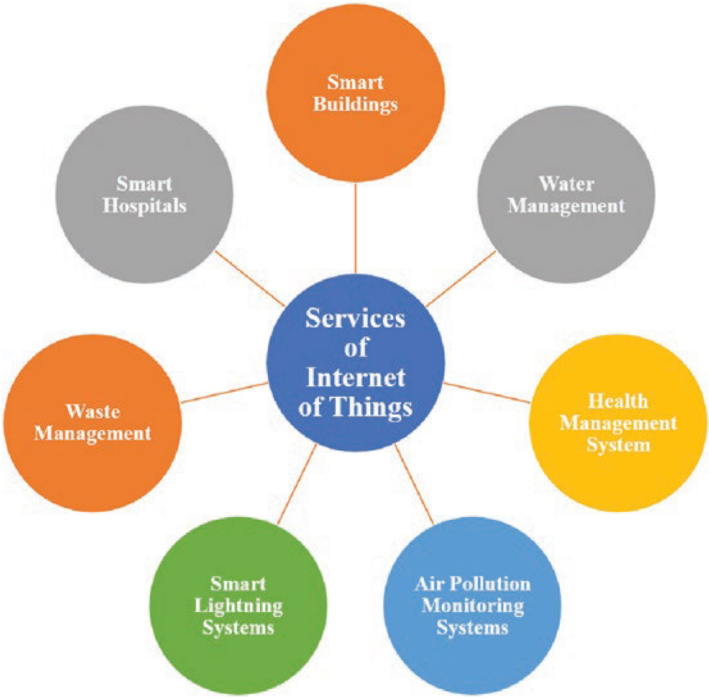


Fig. 1.2 Services of Internet of Things to justify the role of IoT in smart city

the executive's stage. Focal traffic in the executive's stage can dissect the information and consequently change the sign to the traffic conditions in no time [28]. Simultaneously, it utilizes verifiable information to foresee where traffic may go. None of these cycles requires human mediation.

- **Smart Parking:** The city additionally utilizes smart leaving answers for perceiving when vehicles leave the parking garage. The sensor is coordinated on the ground and shows the area of the free parking spot by means of a portable application downloaded by the pilot. Others use vehicle criticism to find the opening and guide the vehicle to follow the way with the least opposition [31]. This smart city application is appropriate for medium-sized city arranging, as smart stopping has become a reality without a complex foundation and huge assumptions.
- **Smart Waste Management System:** This system helps to improve waste assortment productivity and diminish working expenses while satisfactorily resolving all-natural issues related to wasteful waste assortment. In these arrangements, the waste holder is outfitted with a fluid-level sensor [32]. At the point when a specific limit is reached, the transporter at the executive's stage gets a notice through cell phone. The message maintains a strategic distance from half-void waste and helps to avoid any kind of confusion.
- **Smart Water Management System:** Water protection is vital on the grounds that metropolitan regions are squeezing water supply. Luckily, in any case, the

IoT can likewise be utilized for water management, and the gadgets that help this execution are called smart meters. These meters permit water organizations to screen water utilization and investigate normal water utilization for every family, action area, and so forth. The use of AI helps to contrast the current use of the services [33]. Further, AI algorithms can be utilized to break down reported information to anticipate future water utilization in these homes and ventures. These meters additionally help to dissect pressing factors, temperature, and water quality and improve them if necessary. Furthermore, these gadgets can proactively distinguish likely holes by checking underground pipelines and understanding the information, assisting by forestalling genuine floods. The united nations are an example that has effectively implemented the smart water management system.

- **Smart Public Safety Management:** Public security is one of the significant worries of urban areas, and to expand the security of these urban communities, individuals use IoT-based innovation to give continuous observation, examination, and dynamic instruments. The mechanics of these frameworks are as per the following: These gadgets gather information from hearable sensors and CCTV cameras dispersed all through the city and consolidate this information to anticipate potential crime locations and rapidly send security arrangements on a case-by-case basis. In this way, these frameworks permit security facilities to capture and effectively track expected wrongdoers. Perhaps the main use cases for well-being are the executives using shot recognition arrangements [15]. This arrangement may utilize an associated amplifier introduced around there. These receivers consistently gather sounds and send them to the cloud stage and use AI calculations to break down these sounds to distinguish discharges. These gadgets can likewise appraise the situation of the weapon by estimating the time it takes for the sound to arrive at the mouthpiece. The United States has effectively executed this arrangement in more than 90 urban communities.
- **Smart Street Lighting System:** In view of the Internet of Things, smart urban communities strive to make streetlamps smart and control beneficial. They do this by attaching streetlamps with sensors and interfacing them to a cloud in the board framework. Sensors help to gather the relevant information about the street lightning, individuals, and vehicle developments, public vehicle plans, etc. are just the beginning. Ongoing information is then joined with reported information and contribution to AI calculations to help examine various circumstances and the measure of light needed in every circumstance. Subsequently, smart lighting automatically turns on, faints, turns off, or turns on depending on actual ecological conditions, in this manner improving the general lighting plan [5]. For instance, when a passerby or vehicle passes a specific area, the lights around the crossing point will change to a brilliant setting. Miami, Paris, Madrid, and different urban areas have effectively executed brilliant road lighting arrangements, of which Miami is driven by more than 500,000 interconnected streetlamps.

Consequently, the above applications or administrations of the Internet of Things demonstrate the job of the Internet of Things in smart urban areas. With the assistance of Internet of Things innovation, the city executive's office can uphold the passage of enormous quantities of individuals into metropolitan regions. The Internet of Things is not the solitary effective innovation for SC activities. With the assistance of few supporting advancements, SC can be effectively executed.

3 Role of Machine Learning in IoT-Enabled Smart City

The quick improvement of smart urban areas is likewise a stressful issue for created nations. As the market interest of assets and administrations keeps on developing, it has advanced into one of the world's most perplexing frameworks. In this advanced age, smart gadgets are fundamental for building the critical foundation of smart urban communities. As the populace develops, the administrative difficulties of brilliant urban areas are expanding. The information can be gathered utilizing an organization of smart items like sensors, cameras, and actuators [8]. Figure 1.3 shows the smart city information flow.

However, information warehousing, information examination, and perception of the gathered information are likewise a piece of the cycle. Here, the utilization of artificial intelligence-based calculations joined with AI can be a critical thinking measure for establishing a supportable climate. AI, with the assistance of the drawn-out information investigation, will be exceptionally useful in improving the general execution of the smart urban communities' administrations. AI, with the help of IoT,

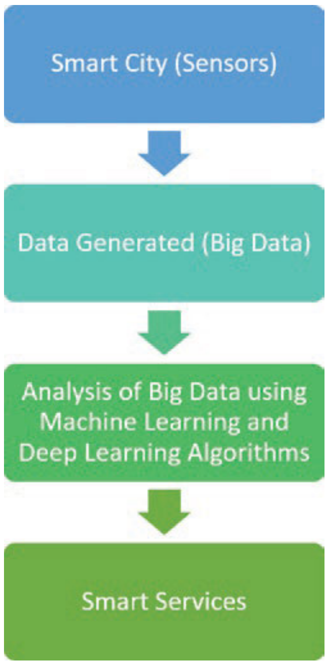


Fig. 1.3 Data flow in smart city

can address any kind of issue identified with the spaces of medical care, energy, transportation, training, and numerous others [30]. Likewise, smart strategies include different relapse models and neural organization-based choice methodology in deciding the example of multifaceted information. Besides, the information foundation assumes an imperative part in the dynamic of brilliant urban areas. The tremendous measure of tangible information recovered from IoT gadgets can be processed utilizing smart learning approaches. AI and ML can tackle huge scope, complex issues in smart urban communities through numerical calculation. Simulated intelligence and ML can change the manner in which smart urban communities work in numerous spaces. Anyhow, in AI- and ML-based smart urban areas, the acknowledgement and reconciliation of ICT foundation programming and equipment stages, structures and calculations, hypothetical arranging, and numerical computational models are basic.

3.1 Machine Learning Approaches for IoT-Enabled Smart City

ML is a subfield of AI that requires the framework to consequently learn by getting to the information given and utilize that information to adapt freely and focus on programming. The expression "AI" was coined by Arthur Samuel in 1959. From making the primary counterfeit neural organization perceptron to utilizing the "nearest neighbor" algorithm to tackle nearest neighbor issues, AI has advanced from fundamental example acknowledgement devices. AI has advanced forms of fundamental example acknowledge devices which performs complex tasks and coordicate a wide scope of spaces of AI and DL. SC sensors and smart gadgets have a lot of information to check separately, so a framework is required that can learn and develop on its own depending on the experience. With the improvement of smart city applications and their creation surroundings, this requires a general, dynamic, and persistent learning system [36]. Consequently, it is needed to investigate the capability of ML and big data in the advancement of SC customized administrations, which are fundamental for improving effectiveness. AI techniques are developing quickly with interest in improved calculations, improved information catch strategies, improved PC organizations, new sensor/IO units, and self-personalization of client conduct. The essential objective of AI calculations is to accurately decipher information that can never be seen and make expectations that go past preparing tests like genuine information. Table 1.1 shows a few calculations utilized in AI and smart urban communities. Exactness and execution can additionally improve the precision of the calculation by expanding the measure of preparing information and improving the calculation's learning capacity [21]. Figure 1.4 shows the order of ML.

There are three kinds of ML: supervised, unsupervised, and reinforcement. The gathered/created information should be preprocessed, blended, rebuilt, and neaten up (eliminating invalid qualities) to prevent the subsequent model from delivering incorrect or wrong expectations. The two principal boundaries to consider when

Table 1.1 Machine learning algorithms and their application domains

Domain	Machine learning algorithms	Application domain
Smart city	Pattern recognition Semantic reasoning	Smart health Public safety Smart transportation
Smart city	Multi-agent learning	Convenient smart homes Real-time traffic routing
Smart home	Reinforcement learning	Convenient smart homes
Smart city	Hidden Markov Model	Smart pipeline
Internet of Things	Rule-based	Smart healthcare
Smart city	Semi-supervised deep reinforcement learning	Energy, water, agriculture, transportation, healthcare

utilizing AI methods are the computational force and the speed of a given strategy. The best calculation is chosen dependent on the client application and ought to be adequately quick to stay aware of changes in the information and produce the ideal yield over the long duration [22]. AI calculations fabricate numerical models dependent on information tests (called "preparing information") and settle on forecasts or choices dependent on them. The development of a directed ML classifier is known as the learning stage, which frames a particular classifier from a bunch of named tests. The presentation of the classifier increases as the measure of preparing information increases. Here are the absolute most regularly utilized ML calculations:

- **K-Means:**

- K-means is one of the easiest unaided bunching calculations. Move a predefined number of bunches to various groups depending on how close the centroids (implies) are and move the information point. K-implies bunching is not difficult to apply to enormous datasets and is regularly utilized as a pre-handling venture for different calculations [23]. These applications incorporate market division, computer vision, space science, and numerous different regions.
- **Support Vector Machine:**
- SVM is a regulated learning model and is vital for information arrangement. In SVM, every information thing is drawn as a point in n-dimensional space (n is the quantity of substances/factors accessible), and the worth of every element is the worth of a specific arrangement. The characterization is then done by discovering the max edge hyperplane (MMH), which separates the two classifications quite well. A hyperplane is a plane or choice space that partitions gatherings of items in various classifications. The SVM support vector is the information point nearest to the hyperplane and characterizes the separating line [25]. The distinction between two information focuses in various classifications is the edge. This can be determined as the upward separation from the line to the help vector. Enormous benefits are viewed as great benefits, and little benefits are viewed as appalling benefits. SVMs are broadly used to tackle certifiable grouping issues in IoT-viable smart urban areas. For instance, in an individual medical care situa-

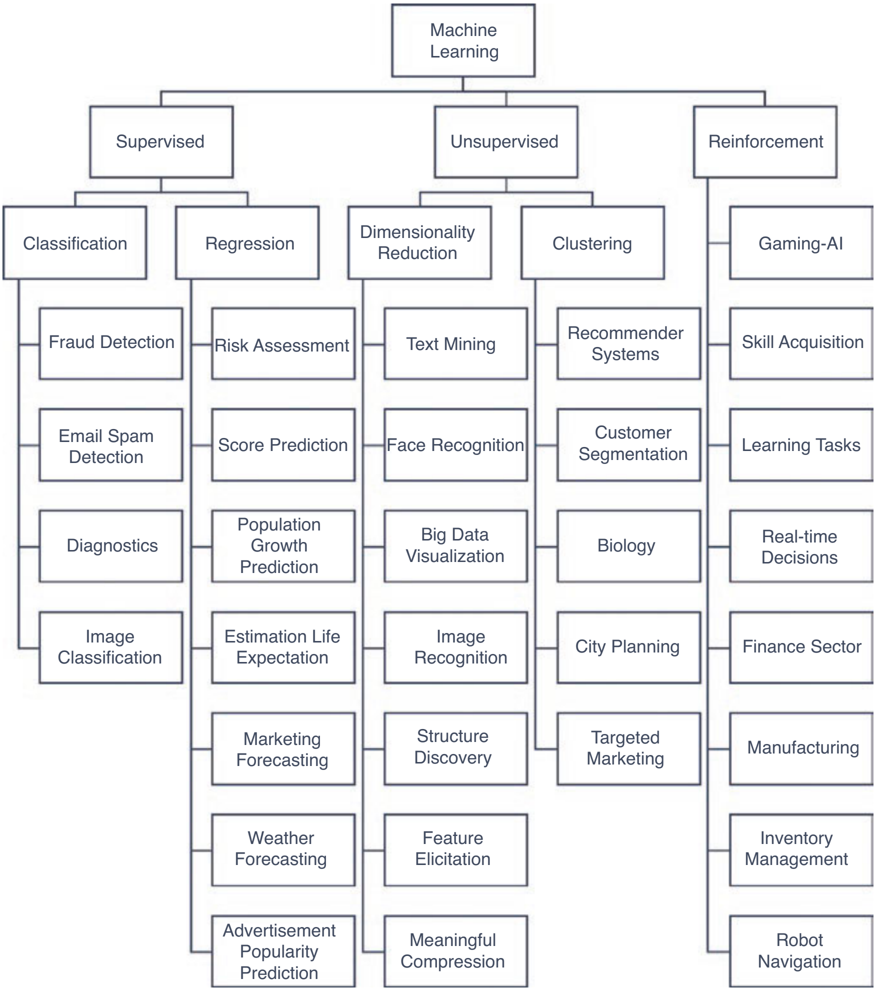


Fig. 1.4 Taxonomy of machine learning

tion, health records observed by a wearable IoT sensor can be shipped off an SVM classifier for a precise conclusion. In the space of organization interruption identification, SVM classifiers can be utilized to distinguish irregularities from a bunch of traffic information starting from correspondence between IoT gadgets [11].

- **Random Forest:**
- RF is an administered ML calculation consisting of choice trees, where each tree relies upon the worth of an autonomously tested arbitrary vector and has similar dispersion for all trees in the forest. This troupe technique lessens overfitting and mistakes because of deviations in the mean outcomes and gives better outcomes.
- **Fuzzy K-Means:**

- Like conventional k-means, which doesn't have a place with a solitary bunch by any means, each point in a fluffy group can have a place with numerous bunches, like the nearest one, and tracks down the most adaptable group. Fluffy analogies contain more data than discrete classes (for example, progress districts between bunches), which can improve the examination. This is a significant device for picture handling and offers a significant benefit as qualities (information focuses) can have a place with different groups. Accordingly, it ought to be generally utilized in bioinformatics to recognize qualities that are co-managed or co-communicated.
- **Naïve Bayes:**

- In light of Bayes' hypothesis, the gullible Bayes classifier is a stochastic classifier, and its working rule is that the suspicions don't rely upon a couple of highlights. The gullible Bayes model is generally utilized in the clinical field for diagnosing coronary illness patients since it tends to be handily built without muddled, tedious boundary assessment. Bayes' hypothesis computes the likelihood of speculation, i.e., the posterior probability $P(c|x)$, from $P(c)$, $P(x)$, and $P(x|c)$:

$$P\left(c|x\right) = P\left(x|c\right).P\left(c\right)$$

$$P\left(x\right)$$

$$P\left(c|X\right) = P\left(x_1|c\right) \times P\left(x2|c\right) \times \ldots . \times P\left(xn|c\right) \times P\left(c\right)$$

where
P(c|x) is the posterior probability of class (target) given predictor (attribute).
P(c) is the prior probability of class.
P(x|c) is the likelihood.
P(x) is the predictor prior probability.

- **Artificial Neural Network:**

- This is an AI subdomain brightened by the science of the human brain. Like neurons that are interconnected in the human brain, counterfeit neural organizations likewise have neurons called hubs, which are connected to one another at different layers of the organization. ANN predominantly comprises three layers. The info layer acknowledges client contribution to numerous configurations, and the secret layer plays out all computations to discover covered-up highlights and designs, and sends the outcomes through the yield layer [35]. It has applications in numerous spaces, for example, vehicle control, direction expectation, example and target acknowledgement, and clinical malignancy conclusion.
- **Genetic Algorithm:**

- GA depends on Darwin's hypothesis of "natural selection." Here, from all potential answers for a given issue, the arrangement goes through recombination and change, bringing about another arrangement, and this interaction is rehashed for ages. Every up-and-comer arrangement is appointed estimated esteem (in light of the target work esteem), and the inexact arrangement might be created to coordinate with more "surmised/ideal" arrangements until the "stop standards" are met. As such, contrasted with customary AI, regardless of whether the info changes marginally or clamor is created, it won't be harmed, and the condition of huge space can be streamlined. The hereditary calculation isn't reasonable for basic issues, so it is unimaginable to expect to acquire inferred data [1]. Some GA applications (for example, intermittent neural organizations and transformation tests) are utilized to limit application speculation blunders, for example, foreseeing stopping accessibility.

Accordingly, the above calculation assumes a significant part in the effective execution of SC. Various applications in various controls help give well-being, unwavering quality, and maintainability to occupants of smart city regions. The utilization of these different advancements additionally assists with giving occupants personal satisfaction.

4 Role of Deep Learning in IoT-Enabled Smart City

Advances in data and correspondence innovation have understood the idea of SC. At SC, various IoT sensors are sent to numerous destinations to successfully gather information on traffic, outflow, and occupant versatility. Top-to-bottom exploration has been generally applied to sensors produced by the Internet of Things in smart urban areas by a few specialists. This is a part of AI where the model consequently takes in highlights from crude information gathered utilizing different sensors. Figure 1.5 shows the essential information on profound learning. This figure gives data about engineering and a kind of profound learning. There are three essential sorts of profound learning design, distinguishing proof, age, etc. These kinds of models are additionally sorted and appear in Fig. 1.7, which presents the profound learning characterization [34]. The essential learning types are supervised, unsupervised, and reinforcement. All these learnings are classified in Fig. 1.4.

With the improvement of the Internet of Things, huge information investigation, AI has understood smart urban areas. The thought behind smart urban communities is to successfully serve the inhabitants of smart urban areas utilizing trend-setting innovation and investigation of the information gathered by the sensors. Metropolitan smartness depends on an innovation-driven foundation, resolving ecological issues similar to a public vehicle, and the utilization of innovation to diminish crime and robbery and guarantee the security of residents. AI utilizes calculations to break down information, gain from it, and settle on choices dependent on the information learned [12]. Deep learning, then again, can learn all alone and settle on astute choices. Figure 1.6 gives a visual comprehension of the principal contrast between AI and deep learning. Essentially, deep learning is a subfield of AI that improves the

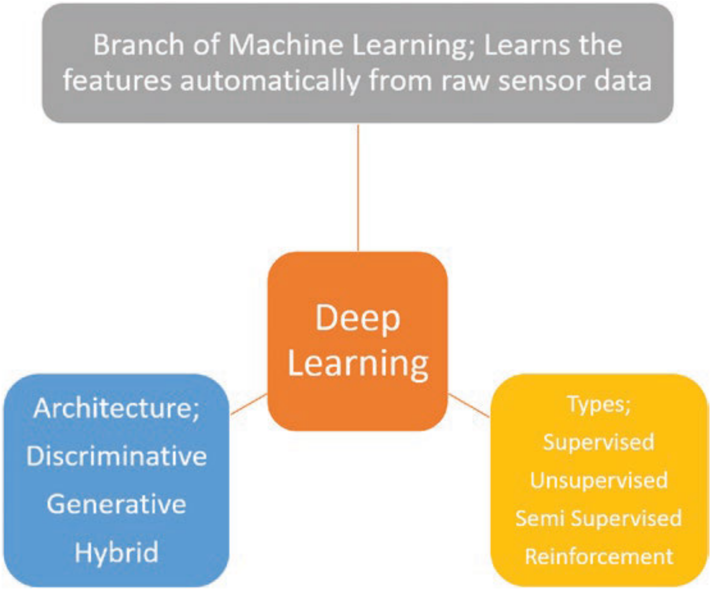


Fig. 1.5 Basic understanding of deep learning

working of AI. The relationship of DL is that rocket motors are composed of DL models, and fuel is a lot of information that specialists and researchers can accommodate in these calculations.

Profound learning (DL) is an AI innovation that can be adequately used to acquire bits of knowledge from information, comprehend designs from information, and characterize/anticipate information. This is an outline of the utilization of various vertical DL/deep neural organizations in smart urban communities.

- **Deep Learning for Urban Modeling for Smart City:**

- In the course of the most recent 50 years, the extent of the number of inhabitants in urban communities all throughout the planet has multiplied. Metropolitan and country regions are continually developing because of mechanical advancement while furnishing inhabitants and travelers with new freedoms for public security availability and worldwide encounters. The creator utilizes smart computer vision and profound learning models to foresee traffic speeds. This model plans to distinguish bland contrasts among created and lacking regions. The creator proposes traffic stopping utilizing profound learning, the Internet of Things, and remote correspondence. This proposition intends to lessen the time needed to discover free parking spots, for example, in huge shopping centers and train stations. With profound learning (DL) and IoT sensors, it is not difficult to decide the best spot to stop in a short measure of time [18]. Unique regard for network accessibility, network transfer speed, network backing, and organization geography empowers multifunctional availability in metropolitan regions. Smart road

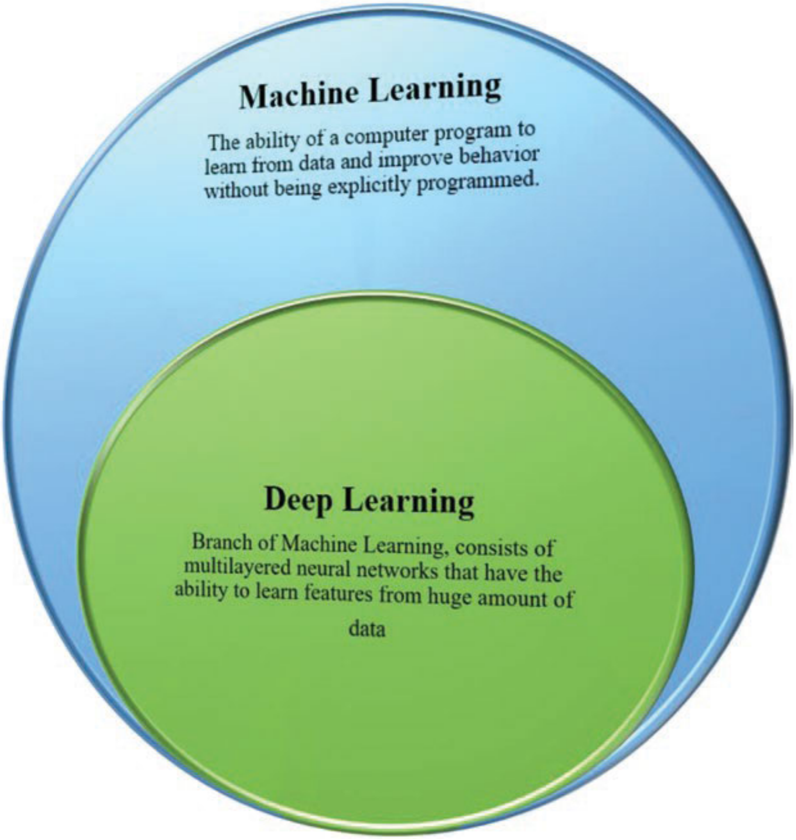


Fig. 1.6 Difference between machine learning and deep learning

overviews utilizing AI methods can set aside half of the energy. Feasible energy dispersion can be accomplished by zeroing in on power age, dissemination, and transmission.

- **Deep Learning for Intelligent Infrastructure of Smart City:**
- By 2030, 60% of the total populace will live in urban communities. Right now, most urban areas are encountering gridlock, and unreasonable and irreversible issues. With quick populace development and an unnatural weather change, tending to every one of these difficulties is presently more significant. Foundation is the foundation of the city and assumes a significant part in addressing these difficulties. The creators of their examination paper have incorporated profound learning innovation into a smart framework model [38]: smart frameworks screen power utilization and dynamic rates as indicated by the seriousness of force utilization. A portion of the scientists additionally proposed a profound learning astute directing component that cycles a lot of information created by different sensors to lessen network blockage issues.

- **Deep Learning for Smart Mobility and Transportation:**
- Transportation frameworks are conveyed through iterative cloud stages, man-made consciousness, associated vehicles, faculty, foundation, and coordination accomplices. In urban communities, the sensors of programmed dump trucks identify all developments around and consequently react to likely mishaps, guaranteeing the security of residents and forestalling mishaps [9]. The creator utilizes profound learning, information investigation, and correspondence advances to interface individuals, streets, and vehicles to address an assortment of traffic-related issues. Their work centers around an assortment of drives pointed toward accomplishing a protected, vehicle-driven climate and agreeable transportation—a protected driving help framework intended to keep drivers from disregarding or ignoring red lights. The person on foot recognition framework utilizes an AI 3D sound system camera to identify different close-by walkers. Bypassing deterrents and vulnerable sides advantage the driver.
- **Deep Learning for Smart Urban Governance:**
- Today, the vast majority live in huge urban areas, and this is the fundamental justification for constructing a brilliant city. The creator audits public administration ideas that assist in investigating the arrangements and qualities of smart regional authorities. Public administration is likewise essential to more readily comprehend the powerful necessities of brilliant urban communities. The general assessment is a significant factor in improving the public authority's urbanization strategy.
- **Deep Learning for Sustainability and Resilience of Smart City:**
- Data creation is expanding step by step. There is a pressing need to foster a viable organization to share this data for use in improving brilliant urban areas. The principal challenge to be acknowledged is the restriction of ecological assets. This can be addressed by planning brilliant frameworks that lessen contamination levels and help individuals have better existences [19]. The creator utilizes grouping and profound learning innovation to execute a smart deficiency recognition strategy. There are different utilities/devices to help you settle on electrical framework choices. Utilizing these devices requires electrical framework skills and they are not accessible to everybody. Another stage called SureCity has been proposed. It can address existing difficulties and help assemble smart and practical urban areas.
- **Deep Learning for Smart Education:**
- The current training framework acquaints new learning strategies with improving understudies' capacities. One of these basic learning strategies is the flipped classroom learning method (FCLM). The creator breaks down the execution of the online flipped classroom learning (OFCLM) technique for youngsters in distance learning. The practice has been exhibited through Web-based meetings to generate new ideas that this strategy can improve students' higher reasoning capacities. Creators of different examination papers have created face identification frameworks that utilize student looks to assess student communications and assess the viability of substance when utilizing versatile CRS for business venture courses. Huge information examination offers individuals the chance to

break down the brain research of lean individuals. The creator additionally proposes another feeling delicate strategy that can decide students' inclinations dependent on head position and facial feelings. The fundamental test of Internet learning is upkeep [38]. This can be because of an assortment of elements, including content showcases and absence of association with coaches. The creator additionally examined time-arrangement grouping issues and utilized intelligent occasions incorporated into the e-learning framework to foresee understudy maintenance.

- **Deep Learning for Smart Health Solutions:**
- Man-made consciousness helps in making shrewd clinical arrangements utilizing new and progressed ideas like profound learning. Ideas, for example, metastasis learning and profound learning models, are exceptionally useful in ordering bosom malignancy dependent on bosom disease cytology pictures. Bosom disease tumor picture characterization and created forecasts end up being more precise than customary profound learning structures [18]. Comparable signs have been seen in the picture division of cervical malignant growth dependent on remote organization innovation. Albeit profound learning has been utilized to ascertain and break down the information produced by the MRI picture division technique. The aftereffects of profound learning and remote organization models show that their proficiency is practically twofold that of conventional profound learning strategies. DL models can be utilized and acoustic recreations to investigate the child's crying, distinguish the strength of the infant, and take the fundamental well-being measures if the child is discovered to be strange. The utilization of profound learning methods works with the advancement of robotized frameworks that improve the personal satisfaction of people. To this end, a programmed electroencephalogram (EEG) pathology recognition framework dependent on profound learning has been created. The framework can catch cerebrum signals as EEG signals in spatiotemporal portrayal to recognize likely human pathologies [12]. Different portable application arrangements are broadly used to gauge diet and screen ascribes identified with general well-being and prosperity. These multi-access actual observing gadgets are ordinarily utilized as wearable gadgets and incorporated with versatile applications to recognize actual anomalies because of lacking nourishment in various age gatherings, establishing a shrewd and solid climate.
- **Deep Learning for Security and Privacy of Smart City:**
- The current century's smart urban areas are the consequence of significant developments in data and correspondence innovation. Residents of these brilliant urban areas are interconnected through cell phones incorporated with the IoT and refined devices, giving unbelievable accommodation and way-of-life enhancements. While gadgets like shrewd meters, smart machines, and brilliant clinical gadgets can accomplish this accommodation, there are connected difficulties as far as data security and protection and keeping up information trustworthiness are concerned (unapproved access is precluded). Profound learning and related advancements have effectively given answers for these security

weaknesses in applications conveying huge information innovation and the IoT [38]. One such execution is to convey the Internet of Things (AD-IoT) abnormality discovery utilizing an arbitrary backwoods calculation to distinguish oddities on the dispersed haze hubs of the Internet of Things gadgets. To recognize the interruption of smart city gadgets into IoT organizations, botnets can compromise the security of such gadgets. A profound learning structure is sent to recognize verification conduct from botnets at the application layer of the Domain Name System. The profound learning system can arrange the ordinary and strange conduct of space names. This incorporates profound learning procedures to distinguish administration advancement competitors and their conveyance measures. This arrangement utilizes a between-hub correspondence convention to decrease network traffic between edge interchanges [35]. Correspondence between the haze layer and the cloud layer utilizes a safe and dependable figuring arrangement. The presentation of this arrangement is better than customary semi-agreeable and non-helpful innovations.

4.1 Deep Learning Approaches for IoT-Enabled Smart City

Profound learning is a subset of AI, while AI is a subset of man-made reasoning. Man-made consciousness is an overall term for innovation that permits PCs to mirror human conduct. AI is a bunch of calculations prepared on information, all of which make this conceivable. Profound learning, then again, is a kind of AI propelled by the design of the human mind. Profound learning calculations attempt to make humanlike inferences by persistently dissecting information in a specific, consistent design [20]. To accomplish this, profound learning utilizes a diverse construction of calculations called neural organizations.

The plan of the neural organization depends on the construction of the human cerebrum. Actually, like human cerebrums to distinguish designs and arrange various kinds of data, neural organizations can be instructed to play out similar errands on the information. The individual layers of neural organizations can likewise be considered as a kind of channel that works from gross to inconspicuous, improving the probability of recognizing and yielding the right outcome. The human cerebrum works comparatively [17]. At whatever point some new data is gotten, the mind attempts to contrast it and known articles. A similar idea is additionally utilized by profound neural organizations.

When planning a brilliant city, one of the fundamental objectives is to chip away at conveying a proficient foundation and administrations while diminishing expenses. The meaning of a smart city given by the European Smart City Platform and the European Life Lab Network is that the smart city "utilized more exhaustive knowledge, incorporated ideas, and individual new innovation applications (for example, RFID and the Internet of Things)." Smart urban areas coordinate broadcast communications innovation to decrease costs, enhance asset utilization, improve intuitiveness, and improve personal satisfaction for inhabitants [40]. So,

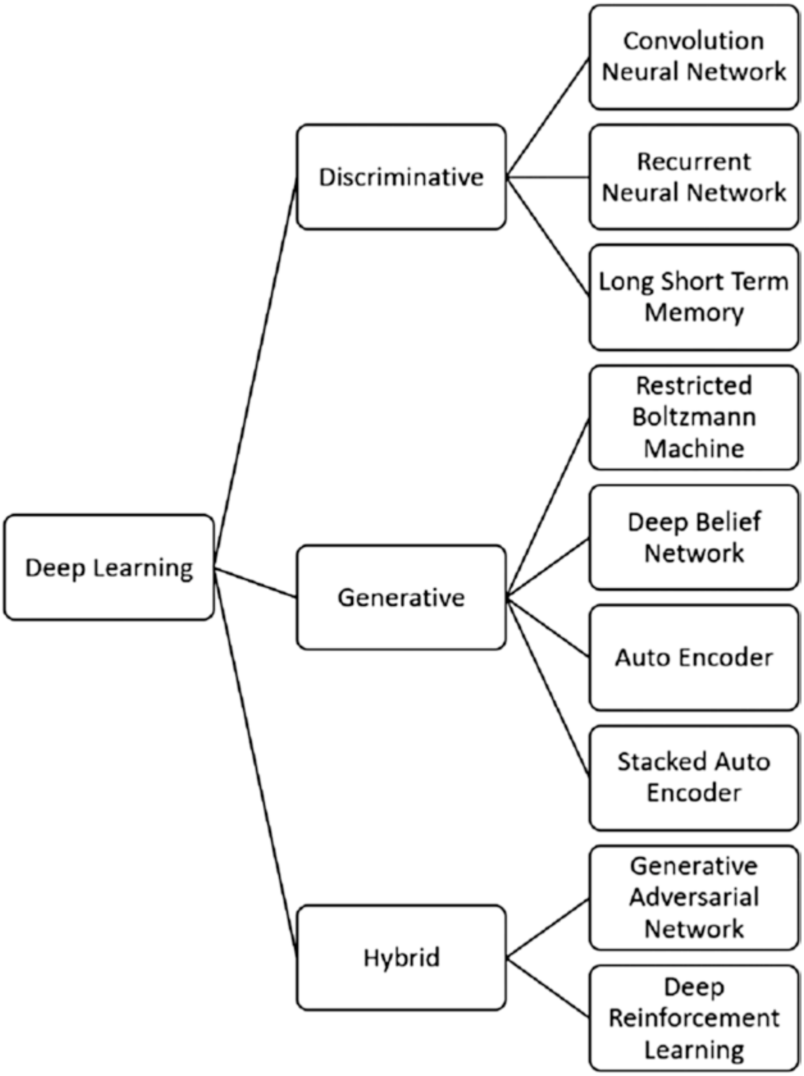


Fig. 1.7 Taxonomy of deep learning

smart urban areas need to coordinate reasonable advancements in occupant's day-to-day existence, for instance, agribusiness, air contamination, transportation, crises, energy and utilization, government, clinical consideration, and water. Figure 1.7 shows the complete classification of deep learning.

- **Convolution Neural Network**
- In profound learning, convolutional neural organizations (CNN/ConvNet) are a kind of profound neural organization and are most regularly utilized for visual

picture examination. CNN utilizes an extraordinary strategy called convolution. Nonetheless, in science, convolution is a numerical procedure on two capacities, delivering a third capacity that addresses how the type of one capacity is changed by another. There are numerous frameworks that can utilize these sorts of organizations, smart urban areas by utilizing a high-thickness organization of traffic and surveillance cameras to give continuous restricted permeability information on all parkways and streets. To accomplish SC administrations, Giyenko et al. carried out a CNN with three convolution layers and prepared it on a Korean CCTV camera dataset. With their strategy, an exactness of 84% or higher can be accomplished [41]. These organizations in brilliant urban communities (cover recognition in broad daylight places, mishap identification on interstates, burglary location in jam-packed spots, and so on) may have different occupations.

- **Recurrent Neural Network**
- RNNs are perhaps the most encouraging calculations being used in light of the fact that they are incredible and strong neural organizations and the solitary calculation with interior memory. In the same way as other profound learning calculations, intermittent neural organizations are moderately old [39]. They were initially made during the 1980s. However, until the most recent couple of years, their actual capability of RNN has been taken note of. Combined with the creation of long-haul memory (LSTM) during the 1990s, the expanded figuring power and the enormous measure of information that should be handled now truly make RNNs stick out. RNN can store significant data about the things it gets, so the model can foresee what will occur next precisely. That is the reason it is a decent calculation for consecutive information (time arrangement, sound, text, monetary information, sound, video, climate, and so on). Contrasted with different calculations, intermittent neural organizations can frame a more profound comprehension of groupings and their specific situation. RNN is the kind of neural network that is valuable for information arrangement [37]. RNNs are obtained from direct-reaction neural organizations, whose conduct is like that of the human cerebrum.
- **Long Short-Term Memory**
- LSTM is an augmentation of repetitive neural organization. The RNN utilizes the yield of the last advance as the contribution for the current advance. The LSTM was planned by Hochreiter and Schmidhuber. It tackled the issue of long-haul reliance on RNNs, where RNNs can't anticipate words put away in long-haul memory. However, they can make more precise forecasts dependent on ongoing data. In the event that the span length is long, the RNN won't give viable execution. Naturally, LSTMs can hold data for significant stretches of time. It is utilized for preparing, anticipating, and positioning dependent on time-arrangement information [27]. Notable applications for LSTMs incorporate language displaying, machine interpretation, picture subtitles, penmanship age, and Q&A visit robots.
- **Restricted Boltzmann Machine**
- The restricted Boltzmann machine is a calculation concocted by Geoffrey Hinton that is helpful for dimensionality decrease, grouping, relapse, community-

oriented sifting, including learning, and theme demonstrating. The Boltzmann machine is an irregular, generative neural organization that can learn interior portrayals, address troublesome combinatorial issues, and tackle them (inadequate time). These are named after the Boltzmann conveyance (otherwise called the Gibbs appropriation). The Boltzmann dissemination is a basic piece of factual mechanics and assists us with understanding the impacts of boundaries like entropy and temperature on quantum mechanics. That is the reason they are called energy-based models (EBMs) [24]. The Boltzmann machine is a nondeterministic (or arbitrary) created profound learning model with just two "covered-up" hub types and "apparent" hub types. There is no leave hub. It might sound odd, yet this gives them this vulnerability. They don't have the run of the mill type 1 or type 0 yield from which the model can be prepared and enhanced utilizing stochastic angle drop. They are exceptionally uncommon in light of the fact that they don't learn models.

- **Deep Belief Network**
- With the improvement of AI and the coming of profound learning, a few instruments and realistic portrayals for interfacing a lot of information have been presented. Profound conviction networks are basically generative realistic portrayals. That is, it delivers all potential qualities that can be produced for the pertinent case. Probabilities and insights are joined through AI and neural organizations. A profound trust network comprises numerous layers with values, with connections between the layers, yet not between the qualities. The primary objective is to permit the framework to characterize the information into various classifications.
- **Autoencoder**
- An autoencoder is a specific kind of feedforward neural organization that has similar information and yield. It packs the contribution to bringing down dimensional code and afterwards recreates the yield from that portrayal. This code is a smaller "outline" or "pressure" of the info, otherwise called an inert spatial portrayal. The autoencoder comprises three segments: encoder, code, and decoder. The encoder packs the contribution to produce code, and the decoder utilizes just that code to reproduce the info. To construct an autoencoder, three things are required: an encoding technique, an interpreting strategy, and a misfortune work that analyzes the yield to the objective. The following segment depicts them. Autoencoders are fundamentally dimensionality decrease (or pressure) calculations for certain significant qualities:

1. **Data specific:** The autoencoder can just altogether pack the prepared information. It varies from standard information pressure calculations (for example, gzip) in light of the fact that it highlights explicit-to-explicit preparing information. Subsequently, it can't anticipate shaping an autoencoder written by hand numbers to pack scene photographs.
2. **Irreversible:** The yield and contribution of the autoencoder are not the very same, they are inexact, but rather they are debased.

3. **Unsupervised:** Run the first information to drive the programmed encoder. Autoencoders are viewed as unaided learning strategies since they can be drilled without unequivocal names. However, more decisively, it is self-checking on the grounds that it creates its own name from the preparation information.

- **Stacked Autoencoder**
- A stacked autoencoder is a neural organization consisting of a few meagre autoencoder layers, with the yield of each secret layer associated with a ceaseless secret layer input. The secret layer is prepared by a solo calculation and enhanced by a directed strategy. Stacked autoencoders include three primary advances:

4. Train the autoencoder utilizing the information to get the learned information.
5. The information gained from the last layer will be utilized as a contribution for the following layer and will proceed until preparation is finished.
6. After preparing every secret layer, utilize the backpropagation calculation to limit the expense capacity and utilize the preparation set to refresh the loads for tweaking.

- The most recent advancement of stacked autoencoder is to give a rendition of the first information with exceptionally definite and promising element data. This variant is utilized to prepare an exercise manual in a particular setting and is more exact than utilizing the first information for preparing. The stack-type autoencoder improves the exactness of profound learning by installing a high-commotion autoencoder in each layer. These autoencoders are utilized to recognize P300 parts and characterize 3D spinal models of juvenile idiopathic scoliosis in the local clinical area. The rich and complex order of inconstancy of spinal disfigurements is fundamental for correlation among treatment and long-haul patient development.
- **Generative Adversarial Network**
- Generative adversarial network (GAN for short) is a generative displaying technique that utilizes profound learning strategies (for example, convolutional neural organizations). Since the produced representation is a solo learning undertaking of AI that includes finding routineness or examples of information and AI, new models that may have been taken from the first dataset utilizing this example. GAN characterizes an issue as an administered learning issue with two sub-models: a generative model prepared to produce new models and a separate model that endeavors to group the models as real (from the space). Train these two models together in a dubious lose-lose situation until the discriminator model is tricked for about a fraction of the time. That is, the generator model produces potential models. GAN is an energizing and quickly developing field with the potential for generative models that can create reasonable models in the scope of issue areas. The most well-known of these are picture-to-picture interpretation undertakings like photograph interpretation. In the evening, reasonable photos of scenes and characters are manufactured, making objects that are unclear to people.

- **Deep Reinforcement Learning**
- Profound support learning consolidates a counterfeit neural organization with a supportive learning engineering to empower programming characterized specialists to become familiar with the ideal activities and accomplish their objectives in a virtual climate. All in all, it consolidates work estimation and objective advancement to plan state-activity sets to anticipated returns. Neural organizations are answerable for the most recent leap forwards in AI in PC vision, machine interpretation, and time-arrangement expectation, however, in the mix with support learning calculations to make incredible things like Deepmind's AlphaGo, which beats GoWorld title games. Support learning is an objective-arranged calculation that can figure out how to accomplish complex objectives or amplify them along with specific measurements in numerous means. For instance, numerous activities can amplify the focuses procured in the game. The RL calculation can begin clearly and accomplish champion execution under the correct conditions. Like pets that are drawn to snarls and medicines, these calculations are rebuffed when they settle on some unacceptable choice and compensated when they settle on the correct choice. This supports learning. Improved calculations joined with profound neural organizations can beat numerous human specialists playing Atari, StarCraft II, and Dota 2 computer games. This may appear to be trifling to non-gamers. However, it's a huge improvement over past accomplishments in support learning, and its specialized level is advancing quickly. Support learning takes care of the issue of quick partner activities with the exhibition defers they create. Like people, support learning calculations may need to hang tight for some time before they can see the consequences of their choices. They work in a postponed way to bring the climate back. In this climate, it may be hard to comprehend which activities lead to which activities and many time steps happen.

5 Literature Review of the Related Domain

The fast development of metropolitan populaces all throughout the planet presents new difficulties to the everyday lives of residents, including contamination observing, public well-being, and gridlock. Through the improvement of more astute urban areas, new advancements have been created to oblige this fast development. By coordinating the Internet of Things (IoT) into the existences of our residents, new shrewd assistance can be developed, and applications that serve various spaces of the city (well-being, observation, horticulture, and so forth) get significant data and thoughts to help improve the personal satisfaction for your residents. AI (ML) and profound learning (DL) are new spaces of computerized reasoning (AI) that have, as of late, shown the possibility to improve the effectiveness and execution of enormous information investigation IoT. This piece of this section audits some new examination articles. There are numerous creators who offer executions of different

SC applications utilizing the most recent innovation. The Internet of Things, AI, and advanced libraries foster smart urban areas.

In 2020, Atitallah and so forth proposed a research paper on the utilization of profound learning and huge information IoT investigation to help the improvement of smart urban areas to consider future bearings around here. The creator initially characterizes the IoT, records the qualities of enormous information produced by the IoT, and afterwards records the different IT frameworks utilized for IoT huge information investigation, like cloud, haze, and edge figuring. At long last, the creator examines mainstream DL models, audits ongoing examinations into creating shrewd applications and administrations for smart urban areas utilizing IoT and DL, and talks about the difficulties present in the improvement of brilliant city administrations.

In 2018, Giyenko et al. clarified the chance of utilizing CNN for permeability assessment utilizing the current CCTV camera network around there. The creator carried out a CNN with three convolution layers and prepared it on a Korean CCTV camera dataset. The creators test elective organization arrangements and designs; test various kinds of organizations, for example, new encouraging neural case organizations; improve test information in drives; and stretch out the framework to recognize current climate conditions, and more later on.

In 2018, Park et al. introduced the significance of IoT innovation to smart urban areas. Accordingly, the focuses and fundamental components for the fruitful advancement of smart urban areas are proposed. In 2019, Chen et al. introduced the most recent exploration on profound learning, shrewd city combination has been directed from two points of view, and situated innovation surveys center around famous and expanded profound learning models, or smart city applications. The creator's reason that this new region actually faces difficulties is because of the intricacy of profound learning and the wide scope of utilization for smart city applications. The creator additionally proposes future headings identified with the adequacy of profound learning, new profound learning standards, information combination, and security assurance, further working with the advancement of examination identified with these bearings. The creator anticipates giving data to appropriate insight around there.

In 2020, Nosratabadi and others introduced the DL and ML methods utilized in brilliant urban areas. Through the new order technique, the advancement of model turn of events and new application fields in feasible metropolitan turn of events and smart urban areas will appear. The outcomes show that the five DL and ML methods are the most broadly used to tackle various parts of shrewd urban areas. They are counterfeit neural organizations, support vector machine, choice tree, set, Bayesian, mixture, fluffy neural, and profound learning. It likewise shows that energy, well-being, and metropolitan vehicles are key spaces of smart urban communities and that DL and ML techniques can help take care of issues.

In 2019, Shrivastav et al. proposed different strategies for breaking down enormous information utilizing AI. In different distributions, the creator has tracked down that the Internet of Things can help foster a more reasonable city. The Internet of Things permits shrewd urban communities to more readily distribute common assets. This additionally implies that AI apparatuses can be extremely useful in breaking down information and gauging future interests.

In 2021, Muhammad et al. summed up the most recent improvements of profound learning applications in shrewd urban communities, new characterizations, difficulties, and openings for future exploration openings. The creators express that their exploration compositions may furnish specialists in the local examination area with the chance to think of new ways to deal with the advancement of their fields of study. In 2018, Yu et al. classified the techniques and apparatuses as per four perception objectives with an emphasis on showing profound learning ideas, design assessment, model investigating and improvement devices, and relating representation depictions. Specifically, the creator gives a table containing the technique or apparatus name, year, representation target, and the strategy or organization type to which the instrument can be applied to assist the client with tracking down the accessible device and strategy. The creator accentuates the significance of visual understanding for profound learning and presents research in this space exhaustively.

In 2019, Zantalis et al. looked into and proposed research compositions that are autonomous surveys of IoT applications in ML innovation and intelligent transport systems (ITS), plainly comprehend patterns in the above regions, and point out conceivable inclusion regions. Smart transport system is viewed as an overall term that covers course advancement, parking areas, streetlamps, mishap anticipation/discovery, street abnormalities, and foundation applications. From the evaluated article, the writer found that brilliant lighting frameworks and smart stopping applications may need ML inclusion. What's more, the creators bring up that course enhancement, stopping, and impact/occurrence discovery are regularly the most well-known ITS applications among analysts.

6 Discussion of Challenges in the Domain of Smart City

The writing audit above features a portion of the difficulties in ANN's profound learning applications at SC. This part centers around the primary difficulties experienced during the examination and gives potential headings to future exploration. This examination shows that the use of regular heuristic calculations in profound learning of CS information investigation is a lacking region. However, you can improve its proficiency and adequacy by advancing the boundaries of the profound learning design utilizing normal disclosure calculations, for example, cuckoo search calculation, firefly calculation, consonant hunt, and molecule swarm streamlining. These are a portion of the difficulties in the space of smart urban areas, and there are not many conceivable exploration openings around here.

Analysts and researchers face numerous hindrances during the time spent accomplishing a solid SC. These issues ought to be maneuvered carefully to stay away from expected breaks the administrations gave, and protection penetrates. The information produced by the SC is so powerless against digital assaults that its secrecy, honesty, and possession are significant issues. Many preparing occupations manage touchy information (for instance, clinical information recorded by clinical IoT gadgets). This can prompt the divulgence of delicate data during the preparation cycle [10]. Altering or messing with information records in any piece of the organization sharing interaction is a significant digital danger that can make ML classifiers wrong. Information suppliers may fail to keep a grip on their information since members approach shared datasets and different members are allowed to duplicate them. The restricted foundation is another test while making an SC. To understand a smart city, it is important to take care of the difficulty that smart gadgets and sensors can't be utilized similarly. A virtual organization of IoT innovation can fabricate just if everybody has limitless admittance to proficient sensors and shrewd gadgets. Picking the best calculation is another significant errand for organizing and preparing the gathered information [3]. Every calculation is known for its properties (exactness, speed, vigor, imperatives, related boundaries, and so forth). Select as per the job that needs to be done.

Another test is the powerlessness to create and assess preparing models in numerous application regions utilizing genuine world datasets. Real information accessibility doesn't really ensure its convenience. Most Electrical Health Data (EHD) information is unstructured information such as different comments, reports, release rundowns, pictures, and sound and video accounts. It is hard to evaluate this information and enter it into a prescient model. Setting processing undertakings should be settled at different levels, including work area, Web, portable, and sensor organizations [13]. For instance, the understanding of human feelings (for instance, shocks) caught by a camera out and about is unique in relation to the translation of a similar response in the house.

Specialists and researchers additionally face different difficulties when working in the shrewd city space, like security and classification, enormous information IoT, profound learning restrictions, and cost and nature of administration. Security and protection are among the principal challenges confronting different SC-related IoT applications. As of late, different sorts of dangers in the DL model have been recognized. These dangers influence the presentation of these models and decrease their precision, adequacy, and dependability. Then again, extraordinary IoT gadgets create a lot of information, which causes numerous issues away, correspondence, preparing, and investigation. Enormous information examination requires important experiences and unique procedures for extricating bits of knowledge [4]. These necessities incorporate superior processors, adaptable distributed computing administrations, mist and ideal edge models, and programming for large information investigation (like Apache Hadoop).

Another test is the constraints of deep learning. DL models have been applied in different SC situations and have delivered great outcomes. Likewise, DL applications have accomplished high precision in grouping and low paces of blunder for expectation in numerous situations. However, there are a few circumstances where these models can't take care of the issue, as follows:

1. DL models require enormous datasets to give great outcomes, particularly under administered learning procedures that require many pictures.

2. Considering enormous datasets can make the preparation cycle complex, computationally serious, and extremely long.
3. In a few cases, particularly for huge pictures (for instance, pictures gathered from satellites), it may require preprocessing prior to preparing the dataset. This preprocessing task is tedious and costly to plan.
4. Researchers have restricted the accessibility of datasets in different controls, particularly those identified with farming. Much of the time, scientists are compelled to create their own datasets, which requires a great deal of time and exertion.
5. DL models permit the fostering of powerful answers for explicit issues. In any case, it cannot be summed up for comparison. However, they do not have indistinguishable purposes. Likewise, it can't reach past the constraints of the underlying dataset.
6. The DL model resembles a black box, where the client doesn't have a clue how the interaction functions, sends contributions through the model, and gets yield likewise. Running as a black box can improve the exactness of model outcomes over the long run. It can likewise be helpless, particularly in prescient applications, if the DL model gains from input information during the preparation cycle. Nonetheless, with a discovery model, it may very well be hard to guarantee that preparation is running effectively, and now and again, the model outcomes will be invalid.

Thus, based on the above conversation, the field of brilliant urban communities remembers different difficulties for the space and gives researchers and specialists different exploration bearings to work.

7 Conclusion and Future Research Opportunities

Deep learning and machine learning techniques have recently facilitated model development in all aspects of smart city and urban development uncertainty prediction, planning, and analysis. The purpose of this chapter is to introduce the latest developments, classifications, application areas, and services of ML, DL, and IoT in the field of smart cities. This chapter provides a rigorous analysis of machine learning, deep learning, and the Internet of Things. It also clarifies the progress of the latest ML and DL applications in smart cities. Machine learning and deep learning classification methods are categorized by possible core algorithms associated with the field. Research and publication development trends in this area indicate that research in this area is expanding rapidly, and this trend is expected to increase in the near future. It highlights current challenges in deep learning in CS, machine learning, and smooth application of the Internet of Things and suggests ways to deviate from those challenges as an opportunity for future research. This chapter is intended for beginner researchers to use as a gateway to their field of study. Professional researchers can use it as a benchmark for further development in the field of study.

7.1 Future Research Opportunities

The joining of ML calculations for handling CS information is a promising region that must be created by creating ICT. Over the long run, less expensive, all the more impressive, and more effective biosensors will be fostered that produce better information. Understanding the capability of large information gathered from metropolitan IoT frameworks should be continually improved as new, more organized calculations can be made to produce. Significant data and help change conventional urban communities into exceptional smart urban areas.

AI instruments and stages can successfully deal with a wide range of datasets. It can likewise be introduced on hardware with restricted assets. This is a significant region that merits more work. Another system needs to be produced for examination that can dissect information streams continuously [19]. The reconciliation of semantic innovation for preparing Web-based media information is likewise a helpful application that can be utilized with SC. A more common and simpler approach to incorporate for residents interaction with brilliant gadgets is an advancing field. Utilization of regular language handling (NLP) and discourse acknowledgment calculations are more advantageous than enormous workstations, an exceptionally little device. By utilizing ML to zero in on the unstructured information present in EHR, doctors can make wonderful close judgments, pick the best medication, and lessen costs.

There are different patterns later on. Move to learn, for instance, is another learning worldview that learns past information and moves and uses that information to tackle new issues. The principal benefit of this kind of learning is that the learning period of this model requires negligible assets and less time contrasted with some DL models that utilize conventional learning types (regulated, solo, improved, and so forth). Numerous examinations have contemplated the utilization of DL models with move learning [24]. Later on, more applications are required to utilize move learning in different shrewd city use cases.

Microservice innovation is a variation of Web administration innovation that empowers the improvement of IoT applications utilizing a bunch of fine-grained, approximately coupled, reusable elements. Truth be told, it is not difficult to adjust or alter these substances to improve the adaptability, versatility, and reusability of your application [16]. In this manner, making a microservice-based structure for overseeing DL administrations empowers a coordinated design that can cycle and examine enormous information, offering amazing types of assistance for an assortment of IoT applications, consequently giving IoT large information, useful for investigation.

Today, cell phones are broadly viewed as modest, universal, and proficient information assortment stages. Moreover, they can be seen all over, catch the everyday exercises of end clients, and are upheld by incorporated correspondence sensors and finders. Utilizing this portable information for DL technique handling is viewed as a powerful answer for improving shrewd city administrations. As of now, research is being created up to the fifth era (5G) to give a multifunctional, adaptable, and adaptable stage that can tackle new issues with the negligible expense and force utilization [10]. 5G isn't just an expansion of fourth-era items, yet additionally another kind of remote organization with another engineering with highlights for high information rates, low idleness, and high dependability. The proposed network was created utilizing another remote access innovation that highlights more than 100 times quicker information transmission speeds than current 4G organizations. The advancement of remote organizations will get numerous progressions, location, speed, and knowledge, along these lines setting out new open doors for the improvement of the Internet of Things and smart urban areas [3]. In this manner, the utilization of 5G in the IoT foundation ought to be viewed as another open exploration.

To make everything brilliant, 6G remote organizations have been investigated to satisfy the high needs presented by quickly expanding versatile information traffic. An upheaval in the 6G plan will achieve new energy and practical correspondence innovations that can change human existence into a smart world just as a shrewd city. IoT information examination will likewise profit by new organizations, acquiring new abilities and eliminating a portion of the current impediments [13]. As of late, blockchain innovation has been drawn into consideration by analysts in different fields. A few specialists characterize a blockchain as "a bunch of squares containing one." A rundown of exchange records has been presented, like the customary public record "Blockchain" to help the execution of safety approaches. However, it is additionally utilized in numerous different regions and applications. Utilizing blockchain in IoT applications has been proposed to improve the security of IoT applications. On account of smart urban areas, blockchain embraces decentralized engineering to establish an ensured climate for applications [4]. Blockchain shrewd agreements are a promising and educational innovation that can be utilized to oversee measures between specialist co-ops and clients.

Billions of smart gadgets are introduced in shrewd city applications. As per reports, the quantity of IoT gadgets utilized in 2019 was about 26.66 billion, which is expected to surpass 41.6 billion by 2025. The terabytes of information created by these shrewd gadgets will surpass 79 and will arrive at 4ZB by 2050. Managing this enormous measure of information requires refreshing and overseeing principles and advances to guarantee consistency. Furthermore, various foundations should be more adaptable to rapidly order information and utilize just great variants [19]. During the time spent on information investigation, it is vital to recollect the conditions of the gathered information (counting transient and spatial angles). Barring this data can prompt bogus choices and expectations. Along these lines, precise examination of spatiotemporal information from IoT shrewd urban areas is vital, particularly in portable rush-hour gridlock situations where occasion area and gadget area are significant.

References

1. Abaker, I., Hashem, T., Chang, V., Anuar, N. B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., & Chiroma, H. (2016). The role of big data in smart city. *International Journal of Information Management, 36*, 748–758. https://www.sciencedirect.com/science/article/pii/S0268401216302778. Accessed 11 Feb 2020
2. Abbas, S., Khan, M. A., Falcon-Morales, L. E., Rehman, A., Saeed, Y., Zareei, M., … Mohamed, E. M. (2020). Modelling, simulation and optimization of power plant energy sustainability for IoT enabled smart cities empowered with deep extreme learning machines. *IEEE Access, 8*, 39982–39997.
3. Al-Sarem, M., Boulila, W., Al-Harby, M., Qadir, J., & Alsaeedi, A. (2019). Deep learning-based rumour detection on microblogging platforms: A systematic review. *IEEE Access, 7*, 152788–152812. https://ieeexplore.ieee.org/abstract/document/8871102/. Accessed 17 Apr 2020
4. Anagnostopoulos, T., Zaslavsky, A., Kolomvatsos, K., Medvedev, A., Amirian, P., Morley, J., & Hadjieftymiades, S. (2017). Challenges and opportunities of waste management in IoT-enabled smart cities: A survey. *IEEE Transactions on Sustainable Computing, 2*(3), 275–289.
5. Atitallah, S. B., Driss, M., Boulila, W., & Ghézala, H. B. (2020). Leveraging deep learning and IoT big data analytics to support the smart cities development: Review and future directions. *Computer Science Review, 38*, 100303.
6. Babar, M., Tariq, M. U., & Jan, M. A. (2020). Secure and resilient demand-side management engine using machine learning for IoT-enabled smart grid. *Sustainable Cities and Society, 62*, 102370.
7. Bakhshi, T., & Ahmed, M. (2018, October). IoT-enabled smart city waste management using machine learning analytics. In *2018 2nd international conference on energy conservation and efficiency (ICECE)* (pp. 66–71). IEEE.
8. Bhattacharya, S., Somayaji, S. R. K., Gadekallu, T. R., Alazab, M., & Maddikunta, P. K. R. (2020). A review on deep learning for future smart cities. *Internet Technology Letters*, 11, e187.
9. Boulila, W. (2019). A top-down approach for semantic segmentation of big remote sensing images. *Earth Science Informatics, 12*, 295–306. https://doi.org/10.1007/s12145-018-00376-7
10. Brynjolfsson, E., & Mcafee, A. (2017). The business of artificial intelligence. *Harvard Business Review*, 1–20. https://hbr.org/cover-story/2017/07/the-businessof-artificial-intelligence. Accessed 17 Apr 2020
11. Camero, A., Toutouh, J., Stolfi, D. H., & Alba, E. (2018, June). Evolutionary deep learning for car park occupancy prediction in smart cities. In *International conference on learning and intelligent optimization* (pp. 386–401). Springer.
12. M. Chen, S. Mao, Y. Zhang, V.C.M. Leung, 2014 Big data: Related technologies, challenges and future prospects. http://www.springer.com/series/10028. Accessed 2 Feb 2020.
13. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 354–377. https://www.sciencedirect.com/science/article/pii/S0031320317304120. Accessed 5 Mar 2020
14. Jan, M. A., He, X., Song, H., & Babar, M. (2021). Machine learning and big data analytics for IoT-enabled smart cities. *Mobile Networks and Applications, 26*(1), 156–158.
15. Kök, I., Şimşek, M. U., & Özdemir, S. (2017, December). A deep learning model for air quality prediction in smart cities. In *2017 IEEE international conference on big data (big data)* (pp. 1983–1990). IEEE.
16. Li, H., Ota, K., & Dong, M. (2018). Learning IoT in edge: Deep learning for the Internet of things with edge computing. *IEEE Network, 32*, 96–101. https://doi.org/10.1109/MNET.2018.1700202
17. Mahdavinejad, M. S., Rezvan, M., Barekatain, M., Adibi, P., Barnaghi, P., & Sheth, A. P. (2018). Machine learning for Internet of things data analysis: A survey. *Digital*

Communications and Networks, 4, 161–175. https://www.sciencedirect.com/science/article/pii/S235286481730247X. Accessed 1 Feb 2020
18. Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqa, A., & Yaqoob, I. (2017). Big IoT data analytics: Architecture, opportunities, and open research challenges. *IEEE Access, 5*, 5247–5261. https://doi.org/10.1109/ACCESS.2017.2689040
19. Mohammadi, G. M., & Al-Fuqaha, A. (2018). Enabling cognitive smart cities using big data and machine learning: Approaches and challenges. *IEEE Communications Magazine, 56*, 94–101. http://www.havenondemand.com. Accessed 2 Feb 2020
20. Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials, 20*, 2923–2960. http://arxiv.org/abs/1712.04301
21. Muhammad, A. N., Aseere, A. M., Chiroma, H., Shah, H., Gital, A. Y., & Hashem, I. A. T. (2020). Deep learning application in smart cities: Recent development, taxonomy, challenges and research prospects. *Neural Computing and Applications*, 18, 1–37.
22. Nosratabadi, S., Mosavi, A., Keivani, R., Ardabili, S., & Aram, F. (2019, September). State of the art survey of deep learning and machine learning models for smart cities and urban sustainability. In *International conference on global research and education* (pp. 228–238). Springer.
23. Obinikpo, A. A., & Kantarci, B. (2017). Big sensed data meets deep learning for smarter health care in smart cities. *Journal of Sensor and Actuator Networks, 6*(4), 26.
24. Qiu, J., Wu, Q., Ding, G., Xu, Y., & Feng, S. (2016). A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*. https://doi.org/10.1186/s13634-016-0355-x
25. Rahman, M. A., Asyhari, A. T., Leong, L. S., Satrya, G. B., Tao, M. H., & Zolkipli, M. F. (2020). Scalable machine learning-based intrusion detection system for IoT-enabled smart cities. *Sustainable Cities and Society, 61*, 102324.
26. Saadi, M., Noor, M. T., Imran, A., Toor, W. T., Mumtaz, S., & Wuttisittikulkij, L. (2020). IoT enabled quality of experience measurement for next-generation networks in smart cities. *Sustainable Cities and Society, 60*, 102266.
27. Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal, 3*. https://doi.org/10.1109/JIOT.2016.2579198
28. Singh, T., & Mishra, J. (2021). Learning with artificial intelligence systems: Application, challenges, and opportunities. In *Impact of AI technologies on teaching, learning, and research in higher education* (pp. 236–253). IGI Global.
29. Singh, T., Nayyar, A., & Solanki, A. (2020a). Multilingual opinion mining movie recommendation system using RNN. In *Proceedings of first international conference on computing, communications, and cyber-security (IC4S 2019)* (pp. 589–605). Springer.
30. Singh, S. K., Jeong, Y. S., & Park, J. H. (2020b). A deep learning-based IoT-oriented infrastructure for a secure smart city. *Sustainable Cities and Society, 60*, 102252.
31. Singh, T., Solanki, A., & Sharma, S. K. (2021a). Role of smart buildings in Smart City—Components, technology, indicators, challenges, future research opportunities. In *Digital cities roadmap: IoT-based architecture and sustainable buildings* (pp. 449–476).
32. Singh, S. P., Solanki, A., Singh, T., & Tayal, A. (2021b). Internet of intelligent things: Injection of intelligence into IoT devices. In *Artificial intelligence to solve pervasive internet of things issues* (pp. 85–102). Academic Press.
33. Solanki, A., & Singh, T. (2021). COVID-19 epidemic analysis and prediction using machine learning algorithms. In *Emerging technologies for battling Covid-19* (Vol. 324, pp. 57–78). Springer.
34. Statista, Smart city initiatives: global spending 2023, 2020., https://www.statista.com/. Accessed 18 Apr 2020.
35. United Nations, 68% of the world population projected to live in urban areas by 2050, 2018., https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. Accessed 6 Mar 2020.

36. Vinayakumar, R., Alazab, M., Srinivasan, S., Pham, Q. V., Padannayil, S. K., & Simran, K. (2020). A visualized botnet detection system based on deep learning for the Internet of things networks of smart cities. *IEEE Transactions on Industry Applications, 56*(4), 4436–4456.
37. Wang, X., Han, Y., Leung, V. C. M., Niyato, D., Yan, X., & Chen, X. (2020). Convergence of edge computing and deep learning: A comprehensive survey. *IEEE Communications Surveys & Tutorials, 22*, 869–904. https://doi.org/10.1109/COMST.2020.2970550
38. Yassine, A., Singh, S., Hossain, M. S., & Muhammad, G. (2019). IoT big data analytics for smart homes with fog and cloud computing. *Future Generation Computer Systems, 91*, 563–573. https://doi.org/10.1016/j.future.2018.08.040
39. Yu, W., Liang, F., He, X., Hatcher, W. G., Lu, C., Lin, J., & Yang, X. (2017). A survey on the edge computing for the Internet of Things. *IEEE Access, 6*, 6900–6919. https://ieeexplore.ieee.org/abstract/document/8123913/. Accessed 1 Feb 2020
40. Zhang, C., Patras, P., & Haddadi, H. (2018a). Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials*. http://arxiv.org/abs/1803.04311
41. Zhang, Q., Yang, L. T., Chen, Z., & Li, P. (2018b). A survey on deep learning for big data. *Information Fusion, 42*, 146–157. https://doi.org/10.1016/j.inffus.2017.10.006

Chapter 2
Understanding New Age of Intelligent Video Surveillance and Deeper Analysis on Deep Learning Techniques for Object Tracking

Preeti Nagrath, Narina Thakur, Rachna Jain, Dharmender Saini, Nitika Sharma, and Jude Hemanth

Abstract Surveillance is an imminent part of a smart city model. The persistent possibility of terrorist attacks at public and secured locations raises the need for powerful monitoring systems with subsystems for embedded object tracking. Object tracking is one of machine vision's basic challenges and has been actively researched for decades. Object tracking is a process to locate a moving object over time across a series of video frames. Object tracking powered with the Internet of Things (IoT) technology provides a broad range of applications such as smart camera surveillance, traffic video surveillance, event prediction and identification, motion detection, human-computer interaction, and perception of human behavior. Real-time visual tracking requires high-response time sensors, tracker speed performance, and large storage requirements. Researchers have ascertained and acknowledged that there is a significant change in the efficacy of drone-based surveillance systems towards object tracking with the inception of the deep learning technologies. Several tracking approaches and models have been proposed by researchers in the area of object tracking and have experienced major improvements with advancement in methods, but object tracking is still considered to be a hard problem to solve. This chapter explains state-of-the-art object tracking algorithms and presents views on current and future trends in object tracking and deep learning surveillance. It also provides an analytical discussion on multi-object tracking experiments based on various datasets available for surveillance and the corresponding results obtained from the research conducted in the near past. FairMOT, GNNMatch, MPNTrack,

P. Nagrath (✉) · N. Thakur · R. Jain · D. Saini · N. Sharma
Department of Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, New Delhi, India
e-mail: preeti.nagrath@bharatividyapeeth.edu; rachna.jain@bharatividyapeeth.edu

J. Hemanth
Department of Electronics and Communication Engineering, Karunya University, Coimbatore, Tamil Nadu, India

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*, Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_2

Lif T, GSDT, and Tracktor++ are among the methods investigated. For the MOT16 and MOT17 datasets, FairMOT generated accuracy of 74.9 and 73.7, respectively, whereas GSDT provided accuracy of 60.7 and 67.1 for the 2DMOT15 and MOT20 datasets. FairMOT is an efficient tracker among the models tested, while MPNTrack is significantly more stable and retains tracklet IDs intact across frames in a series. This concludes FairMOT being an efficient tracker and MPNTrack a stable one. It also discusses a case study on the application of IoT in multi-object tracking and future prospects in surveillance.

Keywords Multi-object tracking · Surveillance · Deep learning · Real-time tracking · smart city surveillance

1 Introduction

The idea of a smart city is focused on the smooth convergence of the current developments in information and communication technology (ICT). A "smart city" can be considered a city that meets the survival needs of its occupants by smart technologies, so that they are given the optimal atmosphere and resources to work on their efficiency and gain a better lifestyle [1]. Smart cities track people in the scope of surveillance by strategically located sensors across the urban landscape that gather data on several variables of urban life [2]. In order to extrapolate information on the issues faced by the community in areas such as crime prevention, electricity usage, traffic control, and waste mitigation, data is distributed, aggregated, and processed by the authorities from these sensors [3]. Drones can be used efficiently to operate a chunk of repetitive, yet important, tasks that take a significant amount of time and energy, so that people can concentrate on greater tasks having assured support from drones [4].

Some of the most important areas where drones can be deployed for surveillance purpose have been listed in Fig. 2.1. The applicability of drones in smart cities is nearly limitless, such as delivery of packages, smart transport, traffic control, power supply, metering of water, and other necessities [5]. In a fictional smart city served by a swarm of robustly networked drones, a road crash on a busy highway, long-distance traffic jams, and lack of prompt response from emergency services will all be avoided. With all the requisite information, the traffic monitoring drone senses the crash and warns the emergency services almost instantaneously. The drone ambulance is a sophisticated solution to deliver emergency aids in road accidents. Traffic can be diverted by the right path recommended by the drone traffic monitoring network, reducing congestion [6]. The surveillance drone will be rushed to the location, alerting and directing law enforcement officials at the same time. IoT has therefore empowered drone technology towards building the smart city providing efficient solutions through end device sensors to detect and respond itself. Efficient

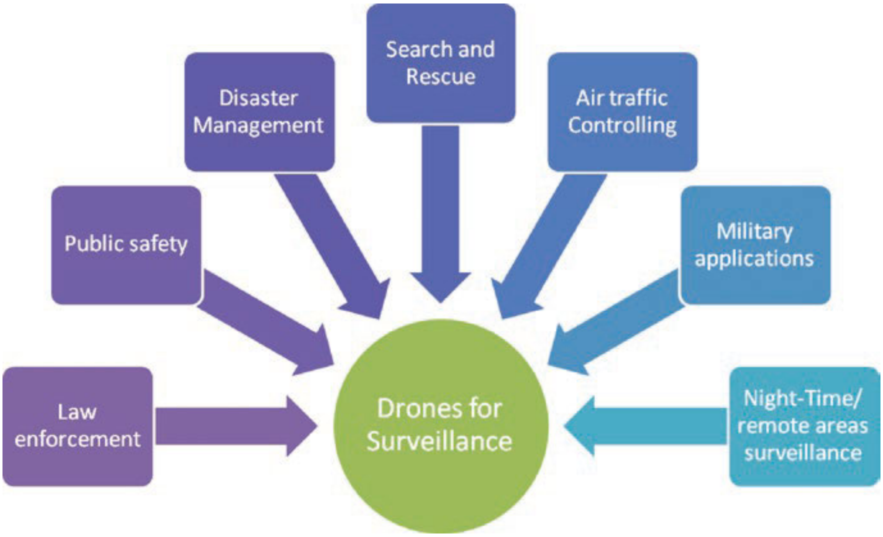


Fig. 2.1 Various areas for drone surveillance

drones, armed with daylight and/or thermal cameras and autonomous capability, can assist law enforcement agencies to detect crimes preemptively. Drones can also assist the police forces to remotely take into account aerial possibilities of potential attacks and crime scene analysis. A drone network and IoT can provide a reliable crime reduction platform.

The basic framework of drone-based surveillance system backed with IoT is shown in Fig. 2.2. The sensor cloud architecture provides low-power IoT devices such as drones with server-level computing capabilities. Computationally intensive activities are off-loaded to the cloud in a sensor-cloud system, while data collection tasks are conducted at the edge [7]. The Internet of Things presents a new model where computers on the edge need to be both cognitive and capable of communicating in real time. Smart cities deliver solutions to our toughest metropolitan issues and solve problems such as growing crime rates and traffic chaos. Visual monitoring through drones is one of the computer vision's most involved fields of study. In a wide variety of computer vision applications, object detection and tracking have played a major role. The fields of tracking, medical imaging, self-driving cars, traffic flow analysis, person counting, human-computer interactions, and public flow analyses have been explored thoroughly [8]. While tracking objects have been widely studied throughout recent decades due to a variety of real-life film issues, such as changes in poses, occlusion, changes in illumination, fast motion, and context uncertainty, tracking objects for surveillance remain a challenge. In object tracking, the aim is to identify the direction of the target over the frame sequences as it moves towards the plane of the image as indicated in sequence flow diagram shown in Fig. 2.3. The tracking systems first detect the objects. An object's monitoring takes place by locating all the objects found in subsequent frames. Then the

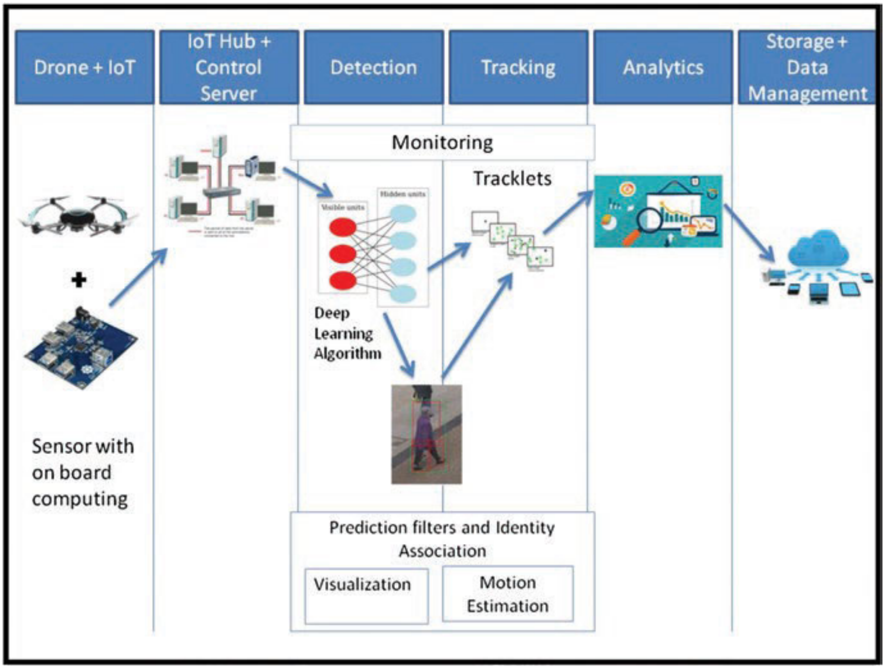


Fig. 2.2 IoT-equipped drone-based monitoring framework for smart city surveillance

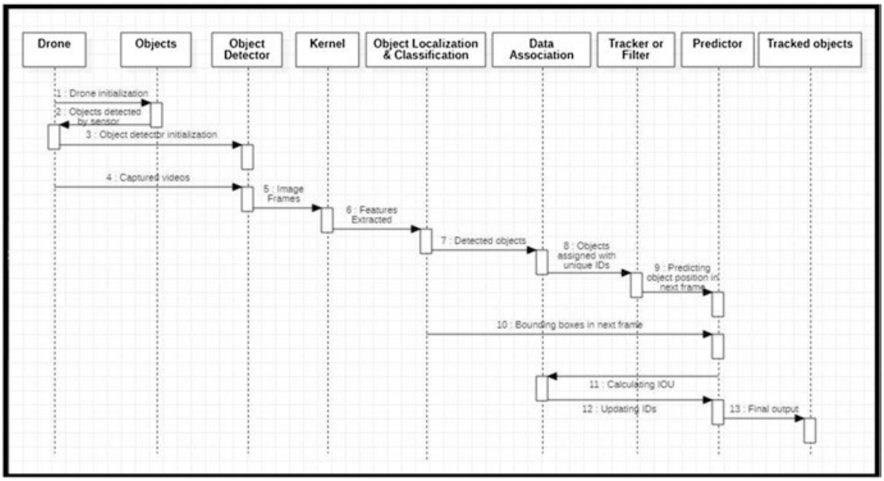


Fig. 2.3 Sequence flow diagram of object tracking algorithm through drone

objects observed are related in consecutive frames based on their characteristics and positions. The object detection module runs directly on image data and attempts to locate areas that correspond to objects detected such as humans. In other modules of object tracking system, such as monitoring, object recognition, or occlusion handling, the identification of regions likely to include the tracked object (e.g., by means of background segmentation methods) provides a source of interest for further study.

The researchers noticed the reliance of tracking by detection of the accuracy of the detection model [9]. Tracking efficiency is also assumed to be increased by optimally modeling convolutional neural network (CNN) models that can decrease the frame rate in real-time object tracking [10]. Tracking is split into identification, estimation, and interaction of objects between the frames to prevent this problem of reduced frame rate. Latest strategies use CNN for visual detection—a deep learning algorithm of convolution layer as the framework of the model. It is advantageous to train CNNs on large-scale data to fully utilize the potential of CNNs that covers a wide range of variation in objectives [11]. A single representation based on video sequences with completely similar features is extremely difficult to learn. Notice that several types of targets with various class names, transitioning forms, and images are used in individual sequences, and sequence-specific algorithm monitoring problems, such as alteration in lighting conditions, deformation, and occlusion. Also, it is harder because similar objects can be seen as the target in one frame and as background in another.

The chapter is aimed at understanding the object tracking algorithm highlighting region-based and feature-based tracking. There can be two different approaches of tracking depending on the mode through which data is obtained. The recorded data is utilized as off-line object tracking, whereas tracking performed on real-time data is called online object tracking. Based on the number of classes tracked in a video sequence, object tracking can be single-object tracking and multi-object tracking. Object tracking task establishes the state of object in the successive frames of a video.

The chapter is structured as follows. Section 2 gives a description of object tracking discussing various object tracking approaches and classification methods. It is followed by Sect. 3, which acquaints with object tracking in surveillance scenarios. Section 4 provides an analysis on experiments performed by various researchers and compares performances of the object tracking models presented. Section 5 puts forth promising future research directions and tasks, and the chapter concludes with Sect. 6.

2 Object Tracking

To track an object in a video stream, a typical method is to use an object detector, a classifier, and a motion estimator or tracker within this order. The object detector scans a video stream framework and chooses the targets that are to be evaluated by the classification system. Any candidate who assigns it is evaluated by the classifier

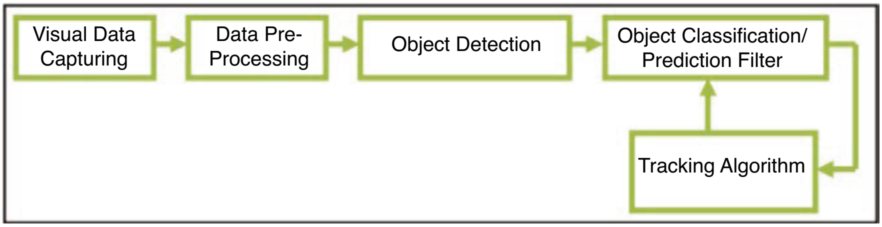


Fig. 2.4 Basic object tracking methodology

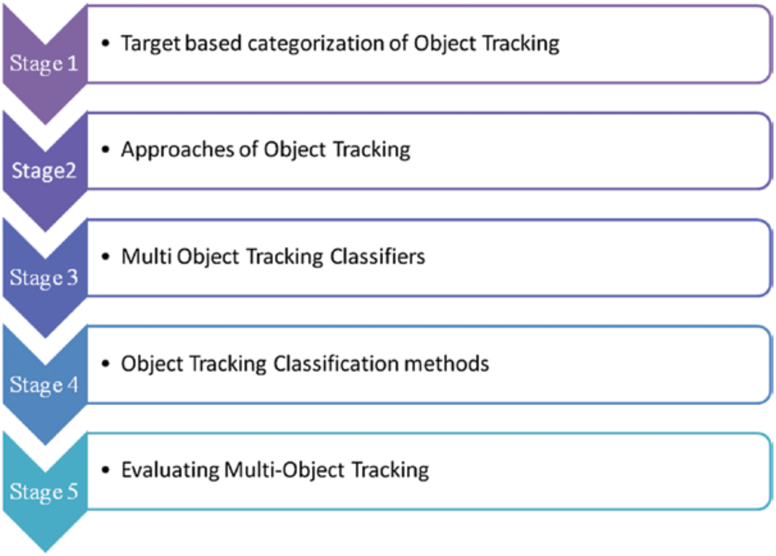


Fig. 2.5 A step-by-step guide to object tracking stages

to show the likeliness of the object to be sought. The target with the highest score is then locked and tracked across the field of view with the tracker [12].

Object tracking begins with object detection, which involves identifying and assigning bounding boxes to objects in an image. A unique ID is assigned to each object identified in an image and this unique ID is assigned by the algorithm in the subsequent frames wherever the same object is spotted. Thus, the algorithm tracks the new location of the same object in consecutive frames. Figure 2.4 depicts the object tracking movement.

It is vital to select the best suitable algorithm for our object tracking application, which can be very daunting to decide. To make the selection task easy, we have segregated the object tracking algorithm into five stages that clearly defines the categorization at each stage as shown in Figure 2.5. In the first stage, the target to be tracked must be selected based on which approaches to track the object are decided in stage 2. Stage 3 helps identify the classifiers suitable for the application and in the

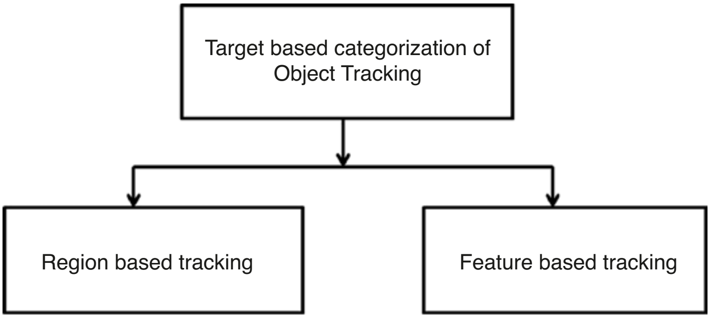


Fig. 2.6 Categorization of object tracking algorithm based on the target in the image

subsequent stage methods for the classification can be selected. Stage 5 measures the efficiency of the methods applied for tracking.

2.1 Stage 1: Target-Based Categorization of Object Tracking

Object tracking as mapped in Fig. 2.6 can be region based or feature based as explained further:

Region-Based Tracking
Region-based tracking algorithms track objects according to image regions identified by a previous segmentation of the moving objects. These methods assume that the foreground regions or blobs contain the object of interest. An example of region-based tracking is given by Wren et al. [13], where they used small blob features to track a person. Gaussian distribution of pixel values was used to identify different body parts. The log probability was used to assign the pixels to the corresponding part, including background pixels. The human was tracked by combining the movements of each small blob. First, it identifies a large blob for the whole person and then continues to find individual body parts.

Feature-Based Tracking
Feature-based tracking methods combine successive object detections for tracking. General local features are extracted, clustered for object classification, and then matched between images. Current methods can be classified into either causal or noncausal approaches. Noncausal approaches construct trajectories by finding the best association according to some global optimization criterion after all observations of a video are computed. Examples are the methods from Leibe et al. [14], Shirit et al. [15], and Wu and Nevatia [16]. Causal methods, in contrast, do not look into the future and only consider information from past frames. Examples for causal feature-based tracking are given by Okuma et al. [17] and Breitenstein et al. [18].

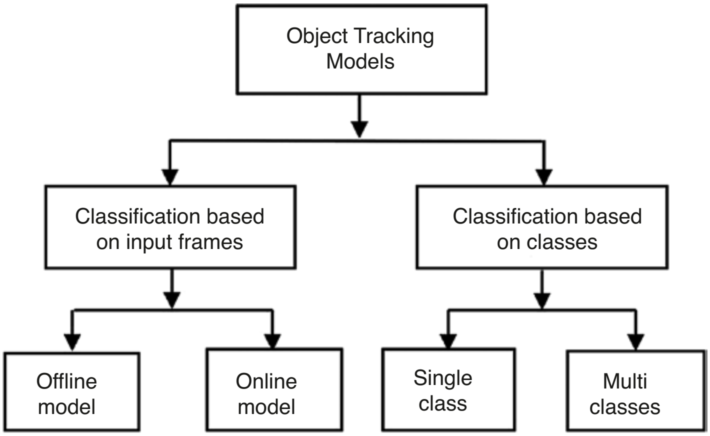


Fig. 2.7 Classification of object tracking models

2.2 Stage 2: Approaches of Object Tracking

The approaches of object tracking models can be classified as shown in Figure 2.7. Tracking techniques can be categorized as off-line and online learning.

- *Off-Line Object Tracking*

  It is the object tracking performed on pre-recorded video sequences where all the past and future frames are known already and can be accessed by the off-line models. Most of the off-line object tracking methods involve graph-based representation towards MOT as comprehensive optimization problem [19, 20] in order to minimize the overall loss function. Due to accessibility to large information, off-line models provide better performance. For example, Siam network-based trackers which are trained off-line are very expeditious.

- *Online Object Tracking*

  The object tracking performed on real-time videos or live video stream such as videos captured by surveillance cameras is the online object tracking. Online methods are more exigent as the algorithm needs to be very fast without any knowledge of future frames. The data association problem, which is determined by the similarity function applied on the detections in consecutive frames in online tracking, is solved either by deterministic methods (greedy association [21] or Hungarian algorithm [22]) or by probabilistic methods [23] as discussed in Sect. 2.4.1.

  Object tracking can also be classified as single class and multi-class.

- *Single-Class Object Tracking*: Single-class models may work for person tracking while multiple-class models are useful to track different classes such as car, truck, bicycle, and pedestrian in a single video frame [24]. The latter model requires classification of each object; however, it is a useful feature in the event of path intersection of different objects.

- *Multi-Class Object Tracking*: Due to the advancement of object detectors [25, 26] that provide accurate identification even in crowded scenes, tracking-by-detection methods have recently seen remarkable performance improvements. By integrating detections given by detectors, the tracking-by-detection methods typically create long trajectories of objects. They can be divided into batch methods and online methods. Batch methods typically use the entire frame detections together to connect scattered trajectories due to occlusion (i.e., tracklets). Short tracklets are created by linking the detections with a collection of detections for whole frames, and the tracklets are globally connected to create longer tracklets. This methodology is therefore very critical for the Global Association, and several approaches [27] have been suggested for the Global Association. However, under long-term occlusion, the efficiency of the batch methods is still limited due to the difficulties of separating various items. In addition, because of the iterative relations for creating globally optimized tuning, they typically need the detections for an entire sequence beforehand, and therefore require massive computation. Thus, adapting the batch methods to real-time applications is challenging. On the other hand, online methods [28–31] can be extended to real-time applications since, using online knowledge till the current frame, they sequentially create trajectories centered in frame-by-frame interaction. However, since unreliable (or even absent) detections of occluded objects in this technique are more difficult to manage, online approaches tend to create scattered trajectories and to migrate under occlusion.

An important component of the visual object tracking pipeline is its appearance model, which comprises two elements—statistical modeling, that is, computing the affinity between several observations, and visual representation, that is, using features to describe the visual characteristics of the object.

2.3 Stage 3: Multi-object Tracking Classifiers

The classifier plays an important role in the overall performance of the tracking system. A poor classifier will lead to bad detection accuracy and false locks. Modules for object classification are important to identify regions of interest given by the visual observation module. In visual surveillance, this module classifies the moving objects into humans, vehicles, and other moving objects. It is a classic pattern recognition problem. Depending on the preprocessing results of the observation layer, different techniques can be considered as shown in Fig. 2.8. However, three main categories of methods are distinguished. They are used individually or in combination.

- *Feature-Detector-Based Classification*

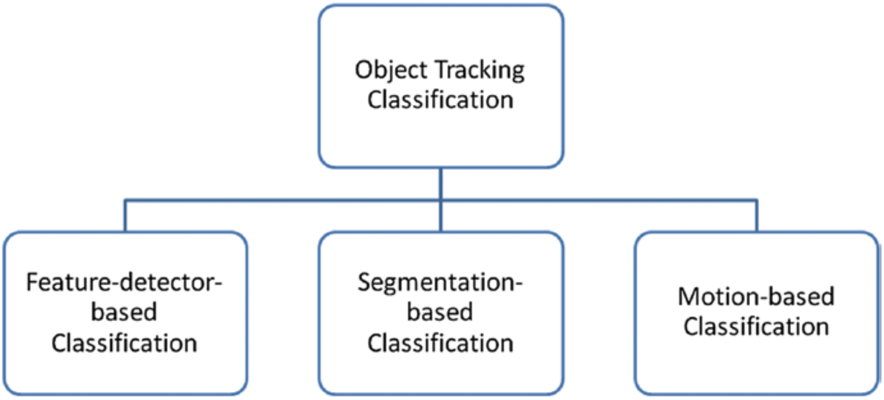


Fig. 2.8 Categorization of object tracking classification

Detected local features can be used to train object class detectors such as ISM by Leibe [14] for general feature detectors. Bo Wu and Ram Nevatia use specific human body part detectors for object detection and tracking [32]. Today, these approaches are generally not suitable for real-time operation. GPU-accelerated implementations will benefit from a huge performance boost in the near future, which will allow some detector-based tracking methods to become real time using high-end graphics hardware.

- *Segmentation-Based Classification*

Object classification based on image blobs is used by video surveillance and monitoring (VSAM) (Collins et al. [33] and Kuno et al. [34]). Different descriptors are used for classification of the object's shape, such as points, boxes, silhouettes, and blobs.

- *Motion-Based Classification*

Periodic motion patterns are used as a useful cue to distinguish nonrigid articulated human motion from motion of other rigid objects, such as vehicles. In [32], residual flow is used as the periodicity measurement.

2.4 Stage 4: Object Tracking Classification Methods

Object tracking classification methods have been mapped into Fig. 2.9, which are further discussed in the following sections. Tracking can be majorly classified into three categories, i.e., point tracking, kernel tracking, and silhouette tracking.

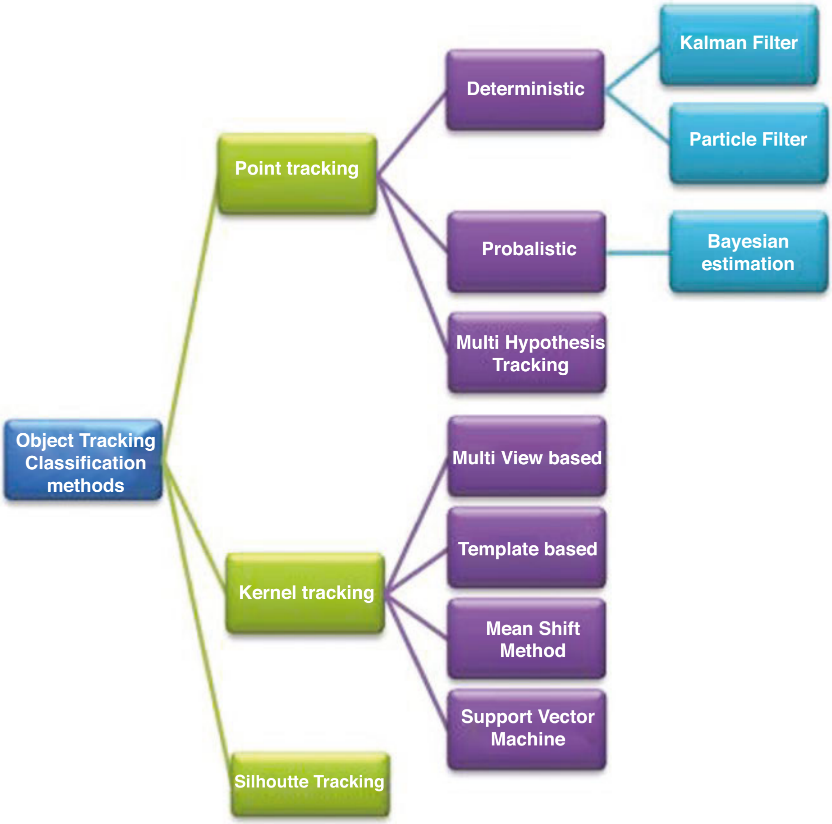


Fig. 2.9 Methods of object tracking classification

2.4.1 Point Tracking

The key feature points extracted by the kernel represent the object. The main function points that are derived from an object are extracted. The edge map's information is used to detect the main points (mostly corner points). The sequence is then observed to find these key points. The position of the tracked key points on the trajectory is compared to the segmented object marked in each frame to ensure that they are accurate [35].

- *Deterministic*

The neural networks (NNs) are commonly used as black box methods to identify and manage such nonlinear dynamic systems as part of the universal approximation property. Unknown nonlinear functions cannot, however, be accurately identified without the fulfillment of the PE condition, an essential condition for adaptive recognition and control groups and difficult to characterize and verify in real-life applications [36, 37]. In order to address this challenge, Wu et al. [38] proves the PE characteristics of RBFN and suggests the nonlinear dynamic system recognition method called deterministic learning theory.

Kalman Filter Anytime that the entity state is believed to be linear with Gaussian noise, the state of such a system can be calculated using the Kalman filter. The Kalman filter also has lower criteria for computation than the particle filter. Kalman filters are used to evaluate states in the state space format on the basis of linear dynamic systems [39]. The phase model describes the state's evolution from $k-1$ to k as follows:

$$x_k = Fx_{k-1} + Bu_{k-1} + w_{k-1} \tag{2.1}$$

where F and B are the state transition matrix applied to previous state vector x_{k-1} and control input matrix to control vector u_{k-1}, respectively, and w_{k-1} is the noise vector. The process model is matched with the measurement model that depicts the relation between the current state and the measurement at the current time step k as

$$z_k = Hx_k + v_k \tag{2.2}$$

Z_k is the measurement vector, x_k is the current state vector, v_k is the measurement noise vector, and H is the measurement matrix.

Particle Filter In real-life scenarios, though, the presumption that all state transformations are linear is very difficult. Therefore, the particle filter can be utilized for state estimation of the system due to its ability to represent random-state densities, and not just Gaussian [17]. When the filter becomes usable in a regular preview loop, measurements are processed. In the prediction process, both the deterministic and stochastic parts of a model are used to predict the probability density function (PDF) at the next instant. The latest calculation is used during the update phase to modify the PDF by adjusting the weight of each sample.

- *Probabilistic*

The methodology uses the double-stochastic matrix as a fundamental basis for the storage and handling of data with uncertain relation. A double-stochastic matrix is a fuzzy mapping that balances the probabilistic links between finite objects. Recursive operations on the matrices represent the inclusion of new knowledge in the scheme, so that uncertainty is maintained in the future. This prohibits the use of statistical decision-making methods that might lead to monitoring mistakes by trying unnecessarily to address ambiguities [40].

Bayesian Estimation Online tracking of multi-objects using just one moving camera is a major challenge; because of global camera motion, the assumptions of 2D traditional motion models (e.g., models of first or second order) no longer hold in the picture coordinate. Recently, the Bayesian estimation methods have been advanced, in which a generalized spatiotemporal Gaussian method is suggested in [16] to map a nonrigid and irregular object. Liu et al. [41, 42] also used Bayesian estimation using a random matrix framework for tracking the extended target from a network of multiple sensors. If the target has sudden motion or gets covered, creating the motion model of the target is a difficult task.

- *Multi-hypothesis Tracking (MHT)*

MHT is one of the first popular visual tracking algorithms. Originally proposed by Reid in 1979 [43], it establishes a tree of possible track assumptions, which provides a structured approach to the data connection problem for each candidate target. The chances are determined for each track and the most probable combination is chosen for each track. MHT is particularly suitable for using knowledge of a higher order, such as long-term motion and appearances, as the whole hypothesis may be used when calculating the probability.

2.4.2 Kernel Tracking

Kernel methods in machine learning are a class of pattern analysis algorithms whose most identified component is the support vector machine (SVM). Spatial masking with an isotropic kernel regularizes the histogram-based target representations of the function. The masking causes functions of spatially smooth similarity appropriate for optimization dependent on gradients. The issue of goal localization can then be modeled using the local maxima attraction basin. In object detection, the method effectively dealt with camera rotation, partial occlusions, noise, and target size variations.

- *Multi-view Based*

Multi-view trackers integrate information from various camera views in order to approximate the time development of objects in a restricted condition. In each view, the data to be combined can be represented by the features of an object (such as position, color, and silhouette).

- *Template Based*

From a reference file, a prototype can be considered, and an image of a scene can be considered as a source image. The goal is to create correspondence between the images from the reference and the source. The matching shows how exactly the picture is identical to the prototype.

- *Mean Shift Method*

Mean shift tracking is a statistical method to find local maxima in probability distributions where the correlation between the image and a shifted target region forms the probability distribution. Its maximum is assumed to locate the target position. Collins et al. [33] presented a method to generalize the traditional 2D mean shift algorithm by incorporating scale into a difference of Gaussian mean shift kernel. An algorithm that recursively moves the data point to the average of data points in its vicinity.

In d-D Euclidean space X, take a set S of *n* datasets x_i. Let $K(x)$ indicate a kernel function, which shows how much x contributes to the average estimate. Then the mean sample m at x is given with kernel K as

$$m(x) = \frac{\sum_{i=1}^{n} K(x - x_i)\, x_i}{\sum_{i=1}^{n} K(x - x_i)} \tag{2.3}$$

- *Support Vector Machine*

Support vector machines (SVMs) are efficient in classification tasks such as recognizing handwriting, faces, cars, etc. It is a complex algorithm and requires high computing time in training and classification of data. New techniques have been developed to make SVMs suitable for real-time applications. These techniques were studied and improved to ensure that real-time tracking systems can be quickly prototyped and developed.

2.4.3 Silhouette Tracking

Silhouette tracker is an effective algorithm and is best suitable for objects exhibiting a large variety of shapes, by reasonably initializing proper object masks and contours, which cannot be obtained easily. On the contrary, a basic bounding box includes too many unnecessary background items, whereas a manually designated mask might be able to provide a realistic silhouette.

Techniques such as joint probabilistic data association filters and multiple hypothesis tracking were used to solve problems of multi-object tracking. But these techniques are inefficient in assigning IDs to the objects. In MHT technique, probabilities are calculated for objects present in previous frames and in new frame. While tracking, the new position of the object is estimated from these calculated probabilities with the help of Kalman filter. Probabilities are also calculated in JPDA technique based on more measurements. These calculations give more information about the frame such as false target objects and density of unknown targets. This technique gives a correlation between the information obtained from present and past data.

2.5 Evaluation of Multi-object Tracking

To compute the training cost curve, the process is run several times with various detection score thresholds T. The common CLEAR MOT metrics [44] are computed over this curve. The final scores are calculated using the field under these curves, which take into account the tracker's output for all detector thresholds *T*. Intersection over union (IoU) of bounding boxes is the most commonly used metric to determine whether an entity and a prediction are linked or not, as it was developed in the MOT15 dataset presentation paper [45].

Intersection over union is a metric used to assess the accuracy of an object detector on a given dataset. To test an arbitrary object detector using intersection over union requires:

1. The ground-truth bounding boxes from the testing set that specify where in the image the object is
2. The bounding boxes predicted from the applied model

Intersection over union (IoU) then compares the predicted bounding box to the ground-truth bounding box. It compares the area of intersection of two bounding boxes to the overall area of the combined region. The following is the formula for measuring IoU:

$$\mathrm{IOU} = \frac{\text{Overlapping region}}{\text{Combined region}} \tag{2.4}$$

The interpretation of IoU is very easy. A score of “1” indicates that the predicted bounding box exactly fits the ground-truth bounding box. A score of “0” indicates that the expected and actual bounding boxes do not overlap in any way. The model performance for predicted bounding box is highly unlikely to be an exact primary bounding box. As a result, IoU can be used to determine how accurate is the object identified in the image. This allows one to consider whether or not the detected object is complete. The IoU is a suitable technique of comparing the results of a training model and the bounding box on the testing range.

The ID assigning and mapping of ground-truth bounding boxes and hypothesis are done using IoU threshold. If in the previous frame hypothesis and ground-truth object are matched, and in the subsequent frame the IoU is 0.5, then the ground-truth object and the hypothesis are matched in that frame, even though some other hypothesis subsists that has IoU greater than the previously matched objects, taking into account the continuity constraint. Using a 0.5 IoU threshold, the remaining objects are attempted to be matched with the remaining hypotheses after the preceding frames' matching has been completed.

False negatives (FN) are ground-truth bounding boxes that cannot be correlated with a hypothesis, whereas false positives (FP) are assumptions that cannot be associated with a real bounding box. Anytime a tracking of a ground-truth object is stopped and subsequently restored, a fragmentation is counted, while every time a tracking ground-truth object ID is improperly updated during the tracking period, an ID switch is counted (IDSW).

3 Object Tracking Algorithms

A lot of testing has been done to identify objects with good efficiency to reach the state of the art. These algorithms are intended to estimate a list of bounding boxes in the image of an input. In an algorithm, sequential video frames are analyzed and target movement between the frames is realized. There are typically two approaches to perform tracking: correlation filter-based approach [46] (ideal for short-term task) and deep learning approach (neural network-based trackers [47] suitable for long-term tasks). Cross-correlation is a technique used by correlation filter-based trackers to locate the target in a picture. Cross-correlation is to create an image-to-kernel (or “filter”) connection that corresponds to the target. It will calculate the dot product between the kernel and an area of the image in operation, and then slide the kernel to another section. This results in a new picture with maximum intensity on the targeted location. In deep learning-based tracking, CNNs are commonly used to acquire visual characteristics. Instead of employing standard CNN models, another popular approach is to train them as Siamese CNNs using contrastive loss functions to identify the optimal combination of features for object classification. A correlation filter is used to anticipate the position of the target in the following frames. The filter generates a response map of the object’s predicted location in the next frame based on the appearance attributes retrieved from a CNN. After integrating the IoU between prediction and detections, the predicted location is used to calculate a similarity score. There are a range of algorithms with vulnerabilities and strengths. In the choice of which algorithm to use, it is necessary to consider the intended use. Some of the most popular deep learning-based multi-object tracking algorithms are briefly discussed below:

- *Simple Online and Real-Time Tracking (SORT)*

  A basic algorithm built specifically for time-critical monitoring scenarios is simple online and real-time tracking (SORT) [48]. It mainly relies on analyzing a detection engine on an underlying target. In order to manage motion prediction and data association, respectively, it contains a combination of Kalman filter and Hungarian algorithm but lacks the appearance features in the association matrix for ease and speed, thereby increasing the amount of identity switches. It can link to any algorithm for object detection. The algorithm monitors several objects and combines the objects in each frame with those identified by using basic heuristics in previous frames. For instance, the intersection-over-union (IOU) metric between bounding boxes in neighboring frames is maximized by SORT.

- *Generic Object Tracking Using Regression Network (GOTURN)*

  By comparing pairs of cropped frames to thousands of video sequences, GOTURN [49] is trained. In the first frame (“previous frame”), the subject’s location is established, and the form is cut to a size double that of the object’s bounding box, centered around the object. The algorithm then predicts where the same point will appear in the next frame. For the second frame, the same double-binding box is used. The second frame is designed to predict the position of the linking box on a convolutional neural network (CNN).

- *Multi-Domain Network (MDNet)*

The VOT2015 competition was won by the Multi-Domain Network (MDNet) [50], a CNN architecture. MDNet's mission is to speed up preparation so that real-time reports can be generated. It separates the network into two sections. The first part is a structured feature extractor that learns to distinguish items from their backgrounds by training them over a series of training sets. The second element is trained to recognize objects in video frames using a specific training set. Thus, MDNet requires only the last few layers of CNN to be changed during preparation, reducing the computing time considerably.

- *SiamRPN*

SiamRPN [51] is the winner of the real-time VOT2018 challenge. It is a very interesting tracker in various applications thanks to its high performances and speed (it can run at 160 Fps GPU). The Siamese-key RPN's structure comprises two parts: the Siamese subnetwork and the proposal area. A fully convolution network with pair images, Siamese subnetwork executes extraction of features. The subnetwork region proposal is inspired by the Faster-RCNN's RPN method. The RPN's goal is to obtain proposals for more precise and appropriate bounding boxes.

- *FairMOT*

FairMOT [52] uses a very basic network configuration that consists of two homogeneous branches for target detection and re-ID function extraction, respectively. It removes the detection branch's disproportionate benefit, efficiently learns high-quality re-ID functionality, and achieves a good trade-off between detection and re-ID for improved MOT performance. The removal of anchors and the use of high-resolution feature maps further match re-ID features to object centers, improving tracking precision significantly.

- *GNNs for Simultaneous Detection and Tracking (GSDT)*

GSDT [53] is a joint MOT approach that simultaneously optimizes object detection and data association modules. Moreover, to model spatial-temporal object relations, GNNs are used to learn more discriminative features that benefit both detection and data association.

- *GNNMatch*

Each object is modeled as a tracklet in the GNNMatch technique, with possible relations between tracklets from previous frames and new detections at the current frame forming the graph's edges. A CNN extracts features from each tracklet's last case, and a GCNN [54] updates these features based on its connectivity by node interaction (tracklets). The MOT conditions of the bipartite matching problem are enforced using a Sinkhorn-based normalization.

- *Tracktor++*

The method Tracktor [55], which consists of two basic processing steps, performs multi-object tracking only using an object detector for a given frame t. The object detector's regression first aligns the item's new position at frame t to the current track bounding boxes b (t-1)k of frame t-1. The linked object classification scores s tk from the modified bounding box locations are then utilized to remove possibly occluded tracks. Second, a Dt of frame t detection set is generated by the object detector (or a defined collection of public detections). Finally, a new track is formed if the detection has no substantial intersection over union with any bounding box in the collection of current tracks.

- *Message Passing Network Track (MPNTrack)*

MPNTrack [56] is a fully differentiable pipeline that enables learning of both feature extraction and data association at the same time. The algorithm's core is a message passing network with a time-aware update stage that can capture the graph structure of the problem. This approach can explain and predict final solutions globally for a number of detections. As a result, MOT's learning is not limited to feature extraction; it also applies to the data association step.

- *Lif_T*

It is an extension of the issue of disjoint paths where more lifted edges are inserted to ensure path connectivity priors. The lifted edges aid in the prevention of ID switches and the reidentification of persons. Lif_T [57] depends on more rigorous visual interpolation and only applies spatial interpolation of static camera sequences.

4 Object Tracking for Surveillance

The well-known PETS dataset, primarily aimed at security applications, is one of the few exceptions. In addition, the standard method of presenting results so far requires the use of various subsets of available data, inconsistent model training, and various assessment scripts even for this commonly used benchmark. The new MOTChallenge benchmark makes it easier for a standard system to make the multi-target monitoring more relevant. Some of the standard datasets used for surveillance are discussed below.

PETS2009 The PETS2009 dataset consists of three elements of multi-viewed sequences of pedestrians walking outdoors. The sections are used to count people and measure density, track people, and evaluate flow with awareness of events. The calibration details are available for the eight distinct cameras. The data collection provides a variety of training images for background, city center, and daily data flow. The counting sequence of people is the most widely used sequence for pedestrian detection. The images are divided into three areas of interest: the complete image, the upper left street, and the lower right street. The standard series of 188 frames and 4307 individuals is extremely populated and thus very complicated. These works embrace details on the ground level and provide extra 3D monitoring. This competition aims (i) to use new or current crowd count and density evaluation technologies, ii) to track individuals in a crowd, and iii) to identify complex flows and mass incidents in a real-life environment. Multiple cameras and multiple actors were shot in the dataset scenarios.

MOT Dataset This is the most widely used dataset for multiple detections and tracing them. MOTChallenge is the most often used benchmark for the management of multi-objects. It also contains some of the main publicly accessible pedestrian tracking databases. For any dataset, both training and test sets, the bases for divisions and detections are available. The MOT20 dataset consists of eight sequences of which four are training sequences and the other four are testing sequences. Three separate scenes film the sequences. Many sequences are shot and spread on the train and test sets per scene. All sequences are captured in high resolution from an improved point of view, with a medium crowd density of 246 pedestrians per frame, ten times denser in contrast with the first benchmark.

UA-DETRAC Dataset for the identification and monitoring of multi-cars. UA-DETRAC is a demanding multi-object identification and multi-object monitoring benchmark in the real world. This dataset consists of 10 h of video footage from 24 different locations in Beijing and Tianjin, China. Videos are shot with the resolution 960 × 540 pixels at 25 frames per second (fps). Over 140,000 frames and 8250 manually annotated vehicles are used in the UA-DETRAC dataset resulting in a total of 1.21 million identified boundary boxes of objects.

Any other datasets that are available are DukeMTMC Multi-Person Dataset Tracking Multi-Camera Tracking Dataset, PoseTrack Multi-Person Pose Tracking Dataset, NVIDIA AI CITY Challenge including "Traffic Flow Analyzing," "Anomaly Detection," and "Multi-Sensor Vehicle Detection and Reidentification" (Multi-Object Tracking and Segmentation).

5 Current Status on Object Tracking

Object tracking performance depends strongly on how reliable the representation of the target object's appearance is. Therefore, obstacles are opposed by the assertion of goal appearance. In recent years, small object motion detection in congested image sequences has drawn much research interest, primarily due to its use in surveillance systems like sonar, optical sensors, and radar [58–63]. In several areas, including aerospace games, dim object detection occurs. The introduction of frame differentiation accompanied by thresholding [64] is an early approach to target detection. In the case of dim moving objects with high noise levels, a systematic protocol is necessary to establish whether or not an object exists. The recent advancements and extensive research have elaborated the field of object tracking. From multi-object tracking to visual tracking, there has been immense research in all areas leading to the origin and development of several tracking techniques. The aim of MOT, that is, multi-object tracking, is to approximate the states of several objects while maintaining their identity over time under changes of visibility and motion. Due to repeated occlusion by debris or other objects, related appearances of multiple objects, and so on, this issue is particularly difficult in a complex environment.

Most algorithms for monitoring collapse into either generative or biased approaches. Using generative models, generative approaches identify the target appearances and aim for the recognition sequences that better match the models. Various computing algorithms for generative object representation, including sparse representation [65, 66], estimate of density [67, 68], and incremental subspace learning [69], have been proposed. By contrast, discriminatory approaches are designed to create a paradigm that distinguishes the target entity from the context. These tracking algorithms generally train classification systems based on many learning instances, P-N learning, online boosts [70, 71], standardized SVMs [72, 73], and so on. Thanks to their heavy computing performance and favorable results, correlation filters have gained interest in recent years in the area of visual monitoring. For visual monitoring, inconsistency in the visibility of targets caused by bad lighting and weather conditions is extremely troublesome [74].

Multiple audiovisual surveillance is still a very important field of study in many areas. This extends from video monitoring by automatic indexing to insightful immersive environments. A robust person module can be used to help other techniques, including gesture recognizers, face markers or speaker markers, head position estimators, and scene analysis software, especially in the last example [75–77]. In the last few years, more and more strategies to meet the demands of unrestricted natural environments have been proposed and human trackers have been brought out of the lab environment into real-life scenarios.

Multi-target tracking (MTT) is another field of research [78]. It is a critical component of monitoring systems that seek to achieve a sequential estimation of target numbers and their states (positions, speeds, etc.). MTT's primary function is to classify the gathered findings into tracks that follow precise goals, a method known as data association [79]. The standard constraint is that each goal generates no more than one observation per scan. However, it is unknown which observations come from which targets, although there are also incorrect observations that do not derive from targets [80]. Several approaches have been developed to handle the problem of data association, including joint probabilistic data association (JPDA), global nearest neighbors (GNN), and multiple hypothesis tracking (MHT). MHT is widely recognized as the most popular approach for organizing the problem of data association in MTT systems.

Based on the type of tracking, there are a lot of techniques that have been developed and researched. Rauch and Firschein used a 3×3 window morphological operation to conduct a pre-screening, supplemented by a track assembler that combined image sequences for target detection. Barniv [64] was the first to use dynamic programming (DP) to detect and track low-observable objects [76]. Blostein [81] first suggested using multi-hypothesis tests to identify small moving objects of unknown location and velocity. In this step, a large number of candidate trajectories are organized into a branched structure [82, 83]. In each trajectory, the integral of the pixel values is consecutively compared with two thresholds. The main advantage of MHT-based algorithms is that they are not affected by the target velocity, target location, or SNR values in the image series [84, 85].

In a broad spectrum of computer vision implementations, CNNs have also shown their excellent representation capacity. A large dataset was used to train deep CNN to improve image classification by Krizhevsky et al. [86]. RCNN [87] was pre-trained on a large dataset and fine-tuned on the test dataset. Despite CNNs' enormous success, only a few monitoring algorithms based on CNN representations have been developed to date. Earlier CNN-based tracking algorithms [88] were trained off-line and could only identify and track predefined object classes. An online learning system was suggested by Li et al. [89] that was based on a collection of CNN layers. Online learning methods have low accuracies due to lack of training data [90, 91]. Off-line tracking network proposed in [92] focused on tracking particular classes. However, a class-specific network is required to track other object classes. This was achieved by Li et al. [93] that proposed an online method where the CNN incrementally learns new object classes while tracking, on observing new instances.

Apart from CNN, most algorithms have used generative and biased approaches to identify the target appearances. Bolme et al. [94] suggested a fast correlation tracker that runs at hundreds of images per second and has a minimal squared error (MOSSE) filter performance total. Henriques et al. [95] developed kernelized correlation filters (KCF) using circulant matrices to effectively integrate multichannel functions into a Fourier domain. In order to increase monitoring performance, many types of KCF tracker were subsequently tested. For example, DSST [96] learns to encode and scale filters independently, and MUSTer [97] uses a model-based psychological memory for short- and long-term memory storage. While these methods are satisfactory in restricted environments, they do have the inherent disadvantage of reverting to low-level manufactured functionality vulnerable in difficult situations, such as change in lighting condition and occlusion.

In MTT systems, MHT is widely acknowledged as the most common approach for organizing the issue of data association. MHT is a logical delayed decision-making approach that permits a definitive choice to be postponed until further data from the report becomes available. MHT structures are divided into two categories. Reid introduced the well-renowned hypothesis-oriented MHT (HOMHT) in his study [98]. HOMHT openly stores and propagates global hypotheses, and the difficulties of dealing with global hypotheses make implementation tough. The other approach, track-oriented MHT (TOMHT), is currently more popular. TOMHT reforms global hypotheses using freshly modified tracks for each scan instead of preserving previously established global hypotheses and usually retains a collection of possible tracks using the layout of the track tree [99].

6 Analysis and Comparison of Algorithms on the Basis of Previous Research Work

To maintain a reasonable comparison with all approaches, we use MOTChallenge detections. The benchmark includes a number of evaluation metrics. The most significant are multiple-object tracking accuracy (MOTA) and ID F1 score (IDF1), which measure two of the most important features of multiple-object tracking: object coverage and identity protection.

The comparison presented in Table 2.1 infers the performance of various deep learning–based multi-object trackers discussed in Sect. 3 of this chapter. It is observed that among all the trackers FairMOT has performed better as compared to its counterparts exhibiting maximum accuracy and F1 score on MOT16 and MOT17 challenges. However, GSDT has performed better in 2DMOT15 and MOT20 challenges. However, the other four trackers, i.e., GNNMatch, MPNTrack, Lif_T, and Tracktor++, have shown comparable and average performance. Among these, Lif_T still shows fair results. It can also be observed that MPNTrack algorithm, despite showing average performance, has least amount of ID switches among all. Therefore, it can be concluded that FairMOT is an efficient tracker whereas MPNTrack is much more stable and retains IDs of tracklets intact across the subsequent frames in a sequence.

7 Promising Future Research Directions and Tasks

Despite the recent development and advancements in object tracking, there is still room for research on several issues that are discussed in this section. For object detection, in order to enhance the accuracy of mapping of small objects under background clutter, the following features need to be updated by network architectures. First is the scale distribution and adaptation in the proposed networks. Typically, objects occur on various dimensions; this is more noticeable in the identification of faces and pedestrian detection. Size-invariant, multi-scale, or size-adaptive detectors are expected to be qualified to improve the robustness of scale shifts. More effective backbone architectures, such as ResNext, negative sample mining [100], and reverse connection [101], are all beneficial for scale-invariant detectors. For multi-scale detectors, both the multi-scale feature map FPN [102] and the generative adversarial network (GAN) [117], which narrows the discrepancies in interpretation between small objects and large objects with economical architecture, provide information into the generation of important pyramid features. It is useful to incorporate the information graph [], attention system [104], cascade network, and size distribution estimation [105] for size-adaptive detectors to detect objects adaptively. Second is the correlation of spatial distribution and contextual modeling. In object detection, spatial distribution plays a significant part. So, in order to obtain possible entity positions, area proposal generation and grid regression are taken. The

Table 2.1 Comparison analysis of performance of deep learning multi-object tracking algorithms: (a) FairMOT, (b) GSDT, (c) GNNMatch, (d) MPNTrack, (e) Lif_T, and (f) Tracktor++ on the state-of-the-art MOTChallenge, i.e., 2DMOT15, MOT16, MOT17, and MOT20 datasets

	Dataset	MOTA ↑	IDF1 ↑	MT	ML	IDS ↓		Dataset	MOTA ↑	IDF1 ↑	MT	ML	IDS ↓
FairMOT	2DMOT15	60.6	**64.7**	47.60%	11.00%	591	**GSDT**	2DMOT15	**60.7**	64.6	47.00%	10.50%	477
	MOT16	**74.9**	**72.8**	44.70%	15.90%	1074		MOT16	66.7	69.2	38.60%	19.00%	959
	MOT17	**73.7**	**72.3**	43.20%	17.30%	3303		MOT17	66.2	68.7	40.80%	18.30%	3318
	MOT20	61.8	67.3	68.80%	7.60%	5243		MOT20	**67.1**	**67.5**	53.10%	13.20%	3133
	(a)							**(b)**					
GNNMatch	2DMOT15	46.7	43.2	21.80%	28.20%	820	**MPNTrack**	2DMOT15	51.5	58.6	31.20%	25.90%	**375**
	MOT16	57.2	55	22.90%	34.00%	559		MOT16	58.6	61.7	27.30%	34.00%	**354**
	MOT17	57.3	56.3	24.40%	33.40%	1911		MOT17	58.8	61.7	28.80%	33.50%	**1185**
	MOT20	54.5	49	32.80%	25.50%	2038		MOT20	57.6	59.1	38.20%	22.50%	**1210**
	(c)							**(d)**					
Lif_T	2DMOT15	52.5	60	33.80%	25.80%	730	**Tracktor++**	2DMOT15	44.1	46.7	18.00%	26.20%	1318
	MOT16	61.3	64.7	27.00%	34.00%	389		MOT16	54.4	52.5	19.00%	36.90%	682
	MOT17	60.5	65.6	27.00%	33.60%	1189		MOT17	53.5	52.3	19.50%	36.60%	2072
	MOT20	–	–	–	–	–		MOT20	52.6	52.7	29.40%	26.70%	1648
	(e)							**(f)**					

similarities between various concepts and groups of objects are neglected. Besides, the position-sensitive score maps in R-FCN leave the global structure data. Subset selection [106] and sequential reasoning tasks [107] for potential alternatives solve these problems. Masking salient sections and combining them in a combined learning manner with the global system are also important [108]. Third is the multitask mutual optimization. Multitask mutual optimization has also been explored by several researchers because of the similarities between separate tasks inside and outside target detection [109, 110]. The functionality of various object recognition subtasks, for instance superpixel semantic segmentation in salient object detection, should be considered and multitask optimization applied to other applications, such as instance segmentation [111], multi-object tracking [112], and multi-person pose estimation. In addition, knowledge from various modalities, such as text [113], thermal details [114], and photographs [115], may be integrated to achieve a more discriminating network, given the particular task.

Multi-object tracking, which is a subtask of object tracking and detection, has also been a topic of extensive research, and despite being under research for around a decade, there are a number of research opportunities in the given field, for instance, the incorporation of multiple cameras in MOT. It is clear that the monitoring of multiple objects will benefit from multi-camera settings [116]. Different cameras have two types of setups. The first one is that the same scene is captured by many cameras, giving different views. In this environment, however, one main issue is how to combine the data collected from multiple cameras. Second is that a nonoverlapping multi-camera network results in a new scene filmed by each camera. In any case, a reidentification question becomes the data correlation across several cameras. Another future work in MOT is the development of multiple three-dimensional object tracking. Much of the latest methods concentrate on 2D multiple-object detection that is on the location of the subject, except with multiple cameras. For high-level computer vision tasks, 3D tracking [118], which may offer more precise location, scale estimate, and reliable occlusion management, is theoretically more useful. 3D monitoring, however, involves camera configuration or has other difficulties to address in calculating camera poses and scene structure. 3D model architecture, meanwhile, is another problem exclusive to 2D MOT.

With the advancement of deep learning and its incorporation in multi-object tracking, various research studies have integrated both the areas to develop strong frameworks, and this has emerged as an active area of research. Models focused on deep learning have appeared as an incredibly effective system for coping with various forms of vision issues, including image classification, object detection, and monitoring of single objects. The good observation model provided by the deep learning model for target detection will dramatically improve tracking efficiency for the MOT problem [119, 120]. More research efforts are required to formulate and model the goal association problem using deep convolutional neural networks, while the first attempt was made very recently to use sequential neural networks for online MOT. It is still in its beginnings to use deep learning to lead the association algorithm and to specifically track it. An additional research is required in this direction to understand whether deep algorithms can still be helpful in this phase. Some previous experiments were conducted [121, 122] to examine overcrowded situations, which displayed that most items are small and completely hidden in this sort of situation and are therefore very difficult to trace. The study of scene comprehension outcomes will provide descriptive details and scene layout, which is very beneficial if it is properly combined into a MOT algorithm for the tracking problem. Changes to the existing identification and monitoring algorithms will potentially involve several new situations, because individuals might emerge in odd locations and actions that are not available in the existing multi-object tracking datasets.

8 Case Study

The International Data Corporation (IDC) says that 135 billion dollars will be spent globally to enable smart city initiatives by 2021. Drones are known to increase protection as an assist. In the near future, drones will be used for security and surveillance. Drones can make a difference by constructing our smart cities so that we can live without regular restrictions. Drone aerial surveys with state-of-the-art high-resolution cameras and post-processing techniques lowered the time, resources, and effort needed to conduct an exploratory investigation. Drones are being used to construct and track transmission lines, power plants, and power grids. In addition, these can also be used in different areas of urban planning and administration, such as road building and suburban settlements. Drones have undergone exponential transition from hobbyist toys to sophisticated IoT applications by exploiting the current IoT uprising. In addition, the introduction of 5G technologies could improve drones' ability to respond to instructions in real time, which allows direct feedback. Their capability and efficiency are anticipated to be improved. Drones improve production and performance while decreasing workload and expense. This is an indispensable contribution to global operations. The use of intelligent drone technologies would enable farmers to adopt precision farming techniques. Drones will be able to track crops thoroughly, which will contribute to high yields. Data on the condition of the crop, water volume, and soil type on a farm can be obtained by farmers using drone sensors. They will also recognize rodents, weeds, diseases, and nourishing deficiencies quickly enough, through deep learning. Following study, the drones will make recommendations and strategies for farmers concerning optimum productivity environments. This would contribute to greater productivity in agriculture and hence higher returns. Inspection and repair of manufacturing installations usually present many problems as well as health threats. Parts like boilers, furnaces, mines, power supplies, and trenches are difficult for regulators to enter and present health risks to the inspecting personnel. Yet servicing will be accelerated and become simpler by inspectors learning how to use smart drones.

The drones capture data from sensors and cameras and then send it to the operator in real time. Smart drones can also be used for the inspection and repair of wind turbines in the future. They travel to the turbine blades, conduct checks, and do small repairs, including deicing, painting, and cleaning where required. This

Fig. 2.10 Realms of drone technology utilization

eliminates the interference of human operators, leading to lower costs and fewer safety risks. Smart drones can also improve how supplies are delivered. Where other modes of transportation are not possible, drones would be useful. This eliminates the challenges of providing emergency relief and equipment in crises and disasters. A pilot project has also started in Rwanda with medical supplies provided by drones to remote hospitals. It is also predicted that drones will change emergency management. In addition to visual searches and reviews, drones will also collaborate and establish emergency shelters for the survivors. Advances of technologies like 3D printing and use of additive construction technology would make that possible. The application areas are shown in Fig. 2.10 where drone technology can be deployed in a smart city.

Drones can also be able to assist firefighters identify the precise location of a fire and find wounded individuals. Smart drones will be used by police to track aggressive activity and unleash tear gas or pepper spray to disperse crowds. Construction firms are now growing investment in drone use. According to Goldman Sachs' forecasts, building firms will invest $11 billion (€9.9 billion) on drones by 2020. Using drones, architects can see from the safety of their workplaces what is taking place on-site. Drones will also continue to provide an up-to-date view of and track the development of a potential structure. This reduces the time and effort needed for frequent visits to the site, especially for customers. The use of drones will also improve the monitoring of roofs, mining activities, and services, which in turn will minimize injuries and therefore increase protection at any construction site. Drones will enter locations that are normally inaccessible and transmit real-time data in the process. Drone consumers will continue to explore new applications through technical advancements. However, there are challenges such as the life of the battery and flight time.

9 Conclusion

IoT-enabled drones can detect, locate, and track a moving human target automatically. The framework focuses on an object recognition machine learning task, which is its computer-intensive kernel. The ability to conduct essential computer vision activities, such as target recognition, photographic imaging, and package delivery, is an indispensable step towards modern and smart applications. This chapter is intended at understanding intelligent surveillance methods through deep learning–based multi-object tracking. MOT consists of two stages in the context of the identification monitoring paradigm: To first locate the alleged position of all objects in the film, the object detector is added to each frame of a video set. Then false-positive detections are eliminated in the data association process, while true detections are paired with the corresponding identities forming trajectories. The chapter presented object tracking approaches and discussed various object detection classifiers used in tracking by the detection method. Some basic state-of-the-art deep learning methods used for object tracking have been briefly discussed. The chapter also lists the standard datasets used particularly for object tracking in surveillance and discusses current trends in the domain. A comparative study is tabulated on different algorithms performed on state-of-the-art MOTChallenge datasets. The algorithms studied include FairMOT, GNNMatch, MPNTrack, Lif_T, GSDT, and Tracktor++. FairMOT produced accuracy of 74.9 and 73.7 for MMOT16 and MOT17, respectively, while GSDT has turned out to perform well for 2DMOT15 and MOT20 datasets with accuracy of 60.7 and 67.1, respectively. Among the models examined, it has been observed that FairMOT is an efficient tracker, but MPNTrack is considerably more stable and keeps tracklet IDs intact across frames in a series. Future prospects of object tracking have been presented. A case study is presented discussing deployment of drone technology powered by IoT in different realms in a smart city.

References

1. Memos, V. A., Psannis, K. E., Ishibashi, Y., Kim, B.-G., & Gupta, B. B. (2018). An efficient algorithm for media-based surveillance system (EAMSuS) in IoT smart city framework. *Future Generation Computer Systems, 83*, 619–628.
2. Monahan, T. (2018). The image of the smart city: surveillance protocols and social inequality. In *Handbook of cultural Security*. Edward Elgar Publishing.

3. Ali, Z., Chaudhry, S. A., Ramzan, M. S., & Al-Turjman, F. (2020). Securing smart city surveillance: a lightweight authentication mechanism for unmanned vehicles. *IEEE Access, 8*, 43711–43724.
4. Vattapparamban, E., Güvenç, İ., Yurekli, A. İ., Akkaya, K., & Uluağaç, S. (2016). Drones for smart cities: Issues in cybersecurity, privacy, and public safety. In *2016 international wireless communications and mobile computing conference (IWCMC)* (pp. 216–221). IEEE.
5. Alsamhi, S. H., Ma, O., Ansari, M. S., & Almalki, F. A. (2019). Survey on collaborative smart drones and internet of things for improving smartness of smart cities. *IEEE Access, 7*, 128125–128152.
6. Chow, J. Y. J. (2016). Dynamic UAV-based traffic monitoring under uncertainty as a stochastic arc-inventory routing policy. *International Journal of Transportation Science and Technology, 5*(3), 167–185.
7. Koubâa, A., Qureshi, B., Sriti, M. F., Javed, Y., & Tovar, E. (2017). *A service-oriented Cloud-based management system for the Internet-of-Drones*. In 2017 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) (pp. 329–335). IEEE.
8. Fujimura, K., & Nanda, H. *Visual tracking using depth data*. U.S. Patent 7,590,262, issued September 15, 2009.
9. Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. S. (2016). *Fully-convolutional siamese networks for object tracking*. In European conference on computer vision (pp. 850–865). Springer.
10. Zhang, P., Zhuo, T., Huang, W., Chen, K., & Kankanhalli, M. (2017). Online object tracking based on CNN with spatial-temporal saliency guided sampling. *Neurocomputing, 257*, 115–127.
11. Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., & Yu, N. (2017). Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism. In *Proceedings of the IEEE international conference on computer vision* (pp. 4836–4845).
12. Chahyati, D., Fanany, M. I., & Arymurthy, A. M. (2017). Tracking people by detection using CNN features. *Procedia Computer Science, 124*, 167–172.
13. Wren, C. R., Azarbayejani, A., Darrell, T., & Pentland, A. P. (1997). Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*(7), 780–785.
14. Leibe, B., Schindler, K., Cornelis, N., & Van Gool, L. (2008). Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(10), 1683–1698.
15. Shitrit, H. B., Berclaz, J., Fleuret, F., & Fua, P. (2013). Multi-commodity network flow for tracking multiple people. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(8), 1614–1627.
16. Wu, B., & Nevatia, R. (2007). Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *International Journal of Computer Vision, 75*(2), 247–266.
17. Okuma, K., Taleghani, A., De Freitas, N., Little, J. J., & Lowe, D. G. (2004). *A boosted particle filter: Multitarget detection and tracking*. In European conference on computer vision (pp. 28–39). Springer.
18. Breitenstein, M. D., Kuettel, D., Weise, T., Van Gool, L., & Pfister, H. (2008). *Real-time face pose estimation from single range images*. In 2008 IEEE conference on computer vision and pattern recognition (pp. 1–8). IEEE.
19. Wei, Y., Sun, J., Tang, X., & Shum, H. Y. (2007, October). *Interactive offline tracking for color objects*. In 2007 IEEE 11th international conference on computer vision (pp. 1–8). IEEE.
20. Luiten, J., Zulfikar, I. E., & Leibe, B. (2020). *Unovost: Unsupervised offline video object segmentation and tracking*. In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 2000–2009).

21. Singh, G., Rajan, S., & Majumdar, S. (2017, April). *A greedy data association technique for multiple object tracking*. In 2017 IEEE third international conference on multimedia big data (BigMM) (pp. 177–184). IEEE.
22. Jonker, R., & Volgenant, T. (1986). Improving the Hungarian assignment algorithm. *Operations Research Letters, 5*(4), 171–175.
23. Serby, D., Meier, E. K., & Van Gool, L. (2004, August). *Probabilistic object tracking using multiple features*. In Proceedings of the 17th international conference on pattern recognition. ICPR 2004 (Vol. 2, pp. 184–187). IEEE.
24. Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *Acm Computing Surveys (CSUR)*, *38*(4), 13–es.
25. Lee, B., Erdenee, E., Jin, S., Nam, M. Y., Jung, Y. G., & Rhee, P. K. (2016, October). *Multi-class multi-object tracking using changing point detection*. In European conference on computer vision (pp. 68–83). Springer.
26. Bose, B., Wang, X., & Grimson, E. (2007, June). *Multi-class object tracking algorithm that handles fragmentation and grouping*. In 2007 IEEE conference on computer vision and pattern recognition (pp. 1–8). IEEE.
27. Zhang, L., Li, Y., & Nevatia, R. (2008, June). *Global data association for multi-object tracking using network flows*. In 2008 IEEE conference on computer vision and pattern recognition (pp. 1–8). IEEE.
28. Jepson, A. D., Fleet, D. J., & El-Maraghi, T. F. (2003). Robust online appearance models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25*(10), 1296–1311.
29. Ross, D. A., Lim, J., Lin, R.-S., & Yang, M.-H. (2008). Incremental learning for robust visual tracking. *IJCV, 77*(1–3), 125–141.
30. Grabner, H., Grabner, M., & Bischof, H. (2006). *Real-time tracking via on-line boosting*. In BMVC.
31. Grabner, H., Leistner, C., & Bischof, H. (2008). *Semi-supervised on-line boosting for robust tracking*. In ECCV.
32. Wu, B., & Nevatia, R. (2007). Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *IJCV, 75*(2), 247–266.
33. Collins, R. T., Lipton, A. J., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., & Wixson, L. (2000). A system for video surveillance and monitoring. *VSAM Final Report, 2000*(1–68), 1.
34. Kuno, Y., Watanabe, T., Shimosakoda, Y., & Nakagawa, S. (1996, August). *Automated detection of human for visual surveillance system*. In Proceedings of 13th international conference on pattern recognition (Vol. 3, pp. 865–869). IEEE.
35. Falconer, G. (1977). *Target tracking with the hough transform*. In 1977 11th Asilomar Conference on Circuits, Systems and Computers, Conference Record, Pacific Grove, CA, USA, pp. 249–252. https://doi.org/10.1109/ACSSC.1977.748926.
36. Zulkifley, M. A., Rawlinson, D., & Moran, B. (2012). Robust observation detection for single object tracking: deterministic and probabilistic patch-based approaches. *Sensors, 12*(11), 15638–15670.
37. Wang, H., Suter, D., Schindler, K., & Shen, C. (2007). Adaptive object tracking based on an effective appearance filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(9), 1661–1667.
38. Wu, W., Wang, C., & Yuan, C. (2019). Deterministic learning from sampling data. *Neurocomputing, 358*, 456–466.
39. Weng, S. K., Kuo, C. M., & Tu, S. K. (2006). Video object tracking using adaptive Kalman filter. *Journal of Visual Communication and Image Representation, 17*(6), 1190–1208.
40. Witte, F. P., & Lucas, D. (1978). *Probabilistic tracking in a multitarget environment*. In 1978 IEEE conference on decision and control including the 17th symposium on adaptive processes, San Diego, CA, USA, pp. 1212–1216. https://doi.org/10.1109/CDC.1978.268126.

41. Liu, J., Reich, J., & Zhao, F. (2003). Collaborative in-network processing for target tracking. *EURASIP Journal on Advances in Signal Processing, 2003*(4), 1–14.
42. Liu, J., & Guo, G. (2019). A random matrix approach for extended target tracking using distributed measurements. *IEEE Sensors Journal*, 1–10.
43. Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control, 24*(6), 843–854.
44. Bernardin, K., & Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing, 2008*, 1–10.
45. Leal-Taixé, L., Milan, A., Reid, I., Roth, S., & Schindler, K. (2015). Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv preprint arXiv,*1504.01942.
46. Li, Y., & Zhu, J. (2014, September). *A scale adaptive kernel correlation filter tracker with feature integration*. In European conference on computer vision (pp. 254–265). Springer.
47. Wang, N., & Yeung, D. Y. (2013). Learning a deep compact image representation for visual tracking. *Advances in Neural Information Processing systems*.
48. Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016, September. *Simple online and realtime tracking*. In *2016* IEEE international conference on image processing (ICIP) (pp. 3464–3468). IEEE.
49. Held, D., Thrun, S., & Savarese, S. (2016, October). *Learning to track at 100 fps with deep regression networks*. In European conference on computer vision (pp. 749–765). Springer.
50. Jung, I., Son, J., Baek, M., & Han, B. (2018). Real-time mdnet. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 83–98).
51. Li, B., Yan, J., Wu, W., Zhu, Z., & Hu, X. (2018). *High performance visual tracking with siamese region proposal network*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 8971–8980).
52. Zhang, Y., Wang, C., Wang, X., Zeng, W., & Liu, W. (2020). FairMOT: On the fairness of detection and re-identification in multiple object tracking. *arXiv preprint arXiv*, 2004.01888.
53. Wang, Y., Kitani, K., & Weng, X. (2020). Joint object detection and multi-object tracking with graph neural networks. *arXiv preprint arXiv*, 2006.13164, 5.
54. Papakis, I., Sarkar, A., & Karpatne, A. (2020). GCNNMatch: Graph convolutional neural networks for multi-object tracking via Sinkhorn normalization. *arXiv preprint arXiv*, 2010.00067.
55. Bergmann, P., Meinhardt, T., & Leal-Taixe, L. (2019). *Tracking without bells and whistles*. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 941–951).
56. Brasó, G., & Leal-Taixé, L. (2020). *Learning a neural solver for multiple object tracking*. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6247–6257).
57. Hornakova, A., Henschel, R., Rosenhahn, B., & Swoboda, P. (2020, November). *Lifted disjoint paths with application in multiple object tracking*. In International conference on machine learning (pp. 4364–4375). PMLR.
58. Zhang, F., Chengfang, L., & Lina, S. (2005). Detecting and tracking dim moving point target in IR image sequence. *Infrared Physics & Technology, 46*(4), 323–328.
59. Liu, D., Jianqi, Z., & Weike, D. (2007). Temporal profile based small moving target detection algorithm in infrared image sequences. *International Journal of Infrared and Millimeter Waves, 28*(5), 373–381.
60. Nemati, A., & Kumar, M. (2014, June). *Modeling and control of a single axis tilting quad copter*. In American Control Conference (ACC), Portland, OR, pp. 3077–3082.
61. Tang, J., Xin, G., & Gang, J. (2013). *Dim and weak target detection technology based on multi-characteristic fusion*. In Proceedings of the 26th conference of spacecraft TT&C technology,Beijing, China, pp. 271–277.
62. New, W. L., Tan, M. J., Meng, H. E., & Venkateswarlu, R. (1999, July). *New method for detection of dim point targets in infrared images*. In SPIE's International symposium on optical science, engineering, and instrumentation, Denver, CO, pp. 141–150.

63. Deshpande, S. D., Meng, H. E., Venkateswarlu, R., & Chan, P. (1999, July). *Max-mean and max-median filters for detection of small-targets*. In SPIE's International symposium on optical science, engineering, and instrumentation, Denver, CO, pp. 74–83.
64. Barniv, Y. (1985). Dynamic programming solution for detecting dim moving targets. *IEEE Transactions on Aerospace and Electronic Systems*, *21*(1), 144–156 (Current Version 2007).
65. Mei, X., & Ling, H. (2009). *Robust visual tracking using l1 minimization*. In ICCV.
66. Zhang, T., Ghanem, B., Liu, S., & Ahuja, N.. (2012). *Robust visual tracking via multi-task sparse learning*. In CVPR.
67. Han, B., Comaniciu, D., Zhu, Y., & Davis, L. (2008). Sequential kernel density approximation and its application to real-time visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(7), 1186–1197.
68. Hare, S., Saffari, A., & Torr, P. H. (2011). *Struck: Structured output tracking with kernels*. In ICCV.
69. Li, X., Hu, W., Zhang, Z., Zhang, X., & Luo, G. (2007, October). *Robust visual tracking based on incremental tensor subspace learning*. In 2007 IEEE 11th international conference on computer vision (pp. 1–8). IEEE.
70. Kalal, Z., Mikolajczyk, K., & Matas, J. (2011). Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34*(7), 1409–1422.
71. Aftab, W., Hostettler, R., De Freitas, A., Arvaneh, M., & Mihaylova, L. (March 2019). Spatio-temporal gaussian process models for extended and group object tracking with irregular shapes. *IEEE Transactions on Vehicular Technology, 68*(3), 2137–2151.
72. Zhang, S., Sui, Y., Yu, X., Zhao, S., & Zhang, L. (2015). Hybrid support vector machines for robust object tracking. *Pattern Recognition, 48*(8), 2474–2488.
73. Hadzagic, M., Michalska, H., & Lefebvre, E. (2005). Track-before detect methods in tracking low-observable targets: A survey. *Sensors & Transducers Magazine, 54*(1), 374–380.
74. Baris, C. (2006). *Dim target detection in infrared imagery*. PhD thesis, Middle East Technical University.
75. Wald, A., & Wolfowitz, J. (1948). Optimum character of the sequential probability ratio test. *Annals of Mathematical Statistics, 19*(3), 326–339.
76. Tzannes, A. P., & Brooks, D. H. (2002). Detecting small moving objects using temporal hypothesis testing. *IEEE Transactions on Aerospace and Electronic Systems, 38*(2), 570–586.
77. Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection*. In CVPR, pp 886–893.
78. Andriyenko, A., Roth, S., & Schindler, K. (2014). Continuous energy minimization for multi-target tracking. *IEEE TPAMI, 35*(1).
79. Brendel, W., Amer, M., & Todorovic, S. (2011). *Multi object tracking as maximum weight independent set*. In CVPR, pp. 1273–1280.
80. Yang, B., & Nevatia, R. (2012). *An online learned CRF model for multi-target tracking*. In CVPR, pp. 2034–2041.
81. Blostein, S., & Huang, S. (1991). Detecting small moving objects in image sequences using sequential hypothesis testing. *IEEE Transactions on Signal Processing, 39*(7), 1611–1629.
82. Breitenstein, M. D., Reichlin, F., Leibe, B., Koller-Meier, E., & Gool, L. J. V. (2011). Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE TPAMI, 33*(9), 1820–1833.
83. Poiesi, F., Mazzon, R., & Cavallaro, A. (2013). Multi-target tracking on confidence maps: An application to people tracking. *CVIU, 117*(10), 1257–1272.
84. Shu, G., Dehghan, A., Oreifej, O., Hand, E., & Shah, M. (2012). *Part-based multiple-person tracking with partial occlusion handling*. In CVPR, pp. 1815–1821.
85. Song, X., Cui, J., Zha, H., & Zhao, H. (2008). *Vision-based multiple interacting targets tracking via on-line supervised learning*. In ECCV, pp. 642–655.
86. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks*. In NIPS.

87. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). *Rich feature hierarchies for accurate object detection and semantic segmentation*. In CVPR.
88. Fan, J., Xu, W., Wu, Y., & Gong, Y. (2010). Human tracking using convolutional neural networks. *IEEE Transactions on Neural Networks, 21*(10), 1610–1623.
89. Li, H., Li, Y., & Porikli, F. (2014). *DeepTrack: Learning discriminative feature representations by convolutional neural networks for visual tracking*. In BMVC.
90. Wang, N., Li, S., Gupta, A., & Yeung, D.-Y.. (2015). Transferring rich feature hierarchies for robust visual tracking. *arXiv preprint arXiv*, 1501.04587.
91. Hong, S., You, T., Kwak, S., & Han, B. (2015). *Online tracking by learning discriminative saliency maps with convolutional neural networks*. In ICML.
92. Fan, J., Xu, W., Wu, Y., & Gong, Y. (2010). Human tracking using convolutional neural networks. *Neural Networks, 21*, 1610–1623.
93. Li, H., Li, Y., & Porikli, F. (2014). *Deeptrack: Learning discriminative feature representations by convolutional neural networks for visual tracking*. In BMVC.
94. Bolme, D., Beveridge, J., Draper, B., & Lui, Y. (2010). *Visual object tracking using adaptive correlation filters*. In Proceedings of the IEEE computer society conference on computer vision and pattern recognition (pp. 2544–2550). https://doi.org/10.1109/CVPR.2010.5539960.
95. Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2015). High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 37*(3), 583–596.
96. Danelljan, M., Häger, G., & Khan, F. (2014). *Accurate scale estimation for robust visual tracking*. In British machine vision conference, pp. 1–11.
97. Z. Hong, Chen, Z., Wang, C., Mei, X., Prokhorov, D., & Tao, D. (2015). *MUlti-Store Tracker (MUSTer): A cognitive psychology inspired approach to object tracking.* In 2015 IEEE conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, pp. 749–758. https://doi.org/10.1109/CVPR.2015.7298675.
98. Reid, D. B. (1979). An algorithm for tracking multiple target. *IEEE Transactions on Automatic Control, 24*(6), 843–854.
99. Chen, L., Wainwright, M. J., Cetin, M., & Willsky, A. S. (2006). Data association based on optimization in graphical models with application to sensor networks. *Mathematical and Computer Modelling, 43*(9–10), 1114–1135.
100. Liu, T., Sun, J., Zheng, N., Tang, X., & Shum, H. -Y. (2007). *Learning to detect a salient object*. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–8.
101. Hosang, J., Benenson, R., Dollar, P., & Schiele, B. (2016). What makes for effective detection proposals? *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(4), 814–830.
102. Alexe, B., Deselaers T., Ferrari, V. (2010). *What is an object?* In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, pp. 73–80.
103. Cheng, H. D., Jiang, X. H., Sun, Y., & Wang, J. L. (2001). Color image segmentation: Advances and prospects. *Pattern Recognition, 34*(12), 2259–2281.
104. Wolfe, J. M., Cave, K. R., & Franzel, S. L. (1989). Guided search: An alternative to the feature integration model for visual search. *Journal of Experimental Psychology: Human Perception and Performance, 15*(3), 419–433.
105. Koch, C., & Ullman, S. (1987). Shifts in selective visual attention: Towards the underlying neural circuitry. In: Vaina, L. M. (Ed.), *Matters of intelligence. Synthese library* (Studies in epistemology, logic, methodology, and philosophy of science, Vol. 188, pp. 115–141). Springer.
106. Parkhurst, D., Law, K., & Niebur, E. (2002). Modeling the role of salience in the allocation of overt visual attention. *Vision Research, 42*(1), 107–123.
107. Liu, T., Yuan, Z. J., Sun, J., Wang, J. D., Zheng, N. N., Tang, X. O., & Shum, H.-Y. (2011). Learning to detect a salient object. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(2), 353–367.

108. Achanta, R., Estrada, F., Wils, P., & Susstrunk, S. (2008). Salient region detection and segmentation. In Gasteratos, A., Vincze, M., Tsotsos, J. K. (Eds.), *Computer vision systems* (Lecture notes in computer science, Vol. 5008, pp. 66–75). Springer.
109. Ma, Y.-F., & Zhang, H.-J. (2003). *Contrast-based image attention analysis by using fuzzy growing*. In Proceedings of the 11th ACM international conference on multimedia, pp. 374–381.
110. Hua, G., Liu, Z. C., Zhang, Z. Y., & Wu, Y. (2006). Iterative local-global energy minimization for automatic extraction of objects of interest. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*(10), 1701–1706.
111. Ko, B. C., & Nam, J.-Y. (2006). *Automatic object-of-interest segmentation from natural images*. In Proceedings of the 18th international conference on pattern recognition, pp. 45–48.
112. Allili, M. S., & Ziou, D. (2007). *Object of interest segmentation and tracking by using feature selection and active contours*. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–8.
113. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.
114. Long, J., Shelhamer, E., Darrell, T. (2015). *Fully convolutional networks for semantic segmentation*. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440.
115. Xie, S., Girshick, R. B., Dollar, P., Tu, Z., & He, K. (2017). *Aggregated residual transformations for deep neural networks.* In CVPR.
116. Kong, T., Sun, F., Yao, A., Liu, H., Lv, M., & Chen, Y. (2017). *Ron: Reverse connection with objectness prior networks for object detection.* In CVPR.
117. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., & Bengio, Y. (2014). *Generative adversarial nets.* In NIPS.
118. Liu, J., Zhang, S., Wang, S., & Metaxas, D. N. (2016). Multispectral deep neural networks for pedestrian detection. *arXiv,* 1611.02644.
119. Li, Y., He, K., Sun, J. et al. (2016). *R-fcn: Object detection via region-based fully convolutional networks.* In NIPS, pp. 379–387.
120. Ristani, E., Solera, F., Zou, R., Cucchiara, R., & Tomasi, C. (2016). *Performance measures and a data set for multi-target, multi-camera tracking*. In Proceedings of the European Conference Computer Vision. Workshops.
121. Su, C., Zhang, S., Xing, J., Gao, W., & Tian, Q. (2016). *Deep attributes driven multi-camera person re-identification*. In Proceedings of the European conference computer vision, pp. 475–491.
122. Zhou, B., Wang, X., & Tang, X. (2012). *Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents*. In Proceedings of the IEEE computer society conference computer vision pattern recognition, pp. 2871–2878.

Chapter 3
Tech to TakeCare: IoT-Based Smart Solution for Real-Time Supervision

Srishti Sharma and Virendra Pratap Singh

Abstract In spite of the ubiquitous evolution of the Internet and the continuous improvements in technology, a miniscule bit has been done to simplify the lives of the key caregivers of infants, senior citizens, and the physically disabled. Recovery from an accident or mishap in golden-ager is sluggish. A simple fall can be life threatening. Likewise, a protracted wet bed for infants may lead to ailments. Young mothers and custodians of the elderly and the incapacitated often have to lay out their whole lives for everyday monitoring errands. Through this work, we propose a dependable, real-time smart supervision system for basic supervision tasks of infants, senior citizens, and the physically disabled. With the aid of the Internet of Things and ThingSpeak cloud, we recommend to construct a solution that records and tracks numerous diverse vitals of babies, elderly, and the incapacitated like blood pressure, body temperature, and heart rate. Furthermore, we also proposition to utilize sound and moisture sensors to amass a regular record of the activities of infants, elderly, and the incapacitated, thus tracking their routine and communicating weekly analysis of the same to the custodians, aiding them in the regulation of healthy lifestyle patterns. Moreover, crucial warnings would be issued and communicated to the custodian's mobile from anywhere in alarming situations. This study is particularly advantageous for the employed class for supervision of their babies and aged while alleviating uncertainties of the mounting competitive employment situation and upholding monetary stability.

Keywords IoT · ThingSpeak · Smart supervision · Sensors · Old · Infants

S. Sharma (✉) · V. P. Singh
Department of Computer Science and Engineering, The NorthCap University, Gurugram, India
e-mail: srishti@ncuindia.edu; virendra17csu207@ncuindia.edu

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*, Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_3

1 Introduction

The field of computer science is progressing at an unfathomable pace and has changed very rapidly in the past decade or so, with the introduction of many new and emerging technologies. One such technology is the Internet of Things which has truly revolutionized the way the world works and the way we intermingle with the physical world around us.

The Internet of Things is a massive grid of components, all connected to one another via the Internet. Internet of Things (IoT) is structured on amassing some statistics from the devices linked to the Internet and relocating this data to a central system. This collected data will be used to deliver a personalized experience for the users by making the devices communicate with one another. There do not exist any standard definitions for the Internet of Things. According to Gartner [1], "IoT is an interconnected network of tangible devices that make use of embedded technologies for transmitting and transferring data amongst their interior states or the exterior milieu." According to the IERC [2], IoT is "an active universal system structure comprised of self-configuring proficiencies founded on typical interoperable communication conventions where tangible and cybernetic 'things' have characteristics, physical qualities, and virtual characters and utilize smart boundaries, effortlessly unified into an information net." The Internet of Things provides a framework for individuals and equipment to be linked Anytime, Anywhere, with Anything and Anybody, by means of Any route/net and Any technology [3]. Linking the physical world using IoT makes life easier and well organized.

The Internet of Things has taken over our lives like a storm! Smart home gadgets like Amazon Alexa, Google Home or smartwatch, smart glasses, smart TVs, smart health monitoring systems, and many more are all brilliant examples of Internet of Things applications. These smart devices are continuously moving data and information to other devices linked to the Internet in real time and between them, to provide an all-inclusive involvement across platforms. IoT devices consist of some components like sensors for getting signals for investigation or actuators for monitoring or both of these [4]. These devices are linked to each other using the Internet and cloud technology for interplay and transfer to analogous devices and people. For example: In a smart home, there is a beautiful interplay of multiple smart devices like smart locks, TV, lights, and air conditioners. All these devices have the capability to transfer data in real time, amongst themselves and to the user(s).

Owing to the decreasing technology prices and easy accessibility in the last decade, there has been a rapid proliferation of this technology around the globe. One of the major contributing factors towards the sudden IoT boom is the rapid surge in smartphone and tablet users across the world. The mobile revolution, coupled with declining sensor prices and cloud-based storage technologies, is pushing for the seamless interoperability of several devices with one another and data exchange thereof. Today, billions of IoT devices around the globe are linked to one another and generate enormous amounts of insightful data having untapped possibilities. To generate insights from these vast amounts of data, artificial intelligence and machine learning can be applied to an extensive array of material gathered using IoT, including video, motionless images, speech, sensors, etc.

IoT allows for mechanization in almost all industries and permits the assemblage of big data. Touted as the backbone that would lead to the fourth Industrial Revolution, IoT has applications in almost all domains we can think of, like in the manufacturing, logistics, and retail sectors; the energy and utilities sectors; intelligent transportation systems; environment monitoring systems; healthcare and home management and monitoring systems; and many more.

Healthcare forms an integral part of everyone's life. Regrettably, the continuously aging populace and the sudden and unprecedented increase in deadly and enduring illnesses including the Covid-19 pandemic outbreak are leading to a substantial strain on caregivers, doctors, accompanying healthcare professionals, and healthcare systems across the globe. Manifestly, a resolution is essential to diminish the burden on primary caregivers of infants and the elderly even as we continue to deliver first-rate attention to these vulnerable persons. In this situation, the Internet of Things is broadly acknowledged as a probable and budding answer to assuage these burdens through the use of monitoring equipment like weight scale, temperature monitor, and patients' vitals recording gadgets for recording and monitoring glucose, blood pressure, heart rate, etc. that may be linked to the Internet and transfer this data from the physical space into the cybernetic space. Therefore, IoT in healthcare has been the emphasis of considerable current studies and research [5–9] as it has the capability to transform healthcare distribution around the globe.

Through this work, we outline a system that uses IoT to streamline and untangle the lives of the prime custodians of babies, senior citizens, and differently abled people. They need continuous monitoring as recovery from a simple illness or a mishap like a fall also takes a lot of time or may even become life threatening for elderly or infants. Correspondingly, a continuous wet bed for babies can cause illness. Custodians of senior citizens and differently abled have to spend a major amount of time in ensuring their well-being. In this work, we propose a real-time smart supervision, reliable wearable device for overseeing babies, differently abled people, and senior citizens. Furthermore, we come up with the solution of using moisture sensor, accelerometer, and sound sensor to keep a day-to-day track record of the happenings with infants, physically disabled people, and the elderly, thus recording their daily routines, conveying weekly analysis of the same to primary caretakers, helping them to make their dependents follow a healthy lifestyle, and raising alerts in case of any variations.

A schematic outline of the proposed system is shown in Fig. 3.1 that brings out the main idea of the proposed system model. The proposed system model is based on recording the reading of some of the vitals of a person like respiratory rate and pulse and transferring the same to a central node and to a machine learning–enabled ThingSpeak cloud system. If some deviation from normal is detected, then an alert will be sounded to the caregiver(s), doctor, or nurses.

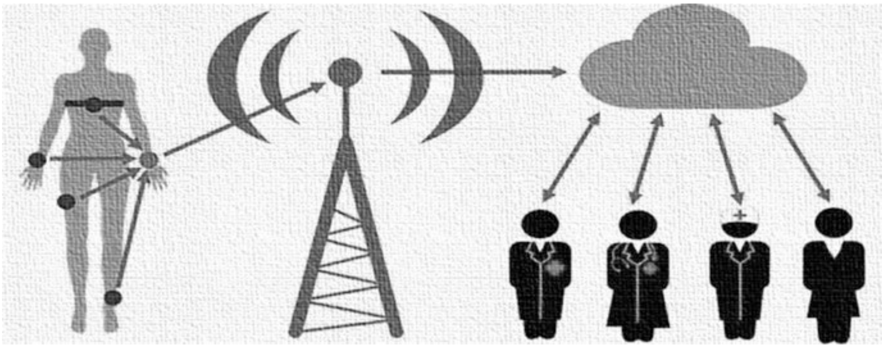


Fig. 3.1 Schematic outline of the proposed model

2 Related Work

Numerous researchers have investigated the potential role of IoT in healthcare. A detailed survey is carried out by authors in [10], with emphasis on existing IoT systems, probable frameworks, and outstanding complications and difficulties. The authors throw light on each of these topics one by one. In [11], the authors talk about data acquisition, accumulation, and maintenance and scrutiny. In [12], the authors carry out a comparative analysis of the different types of sensors with respect to their data transfer capabilities in particular. However, they do not propose any model or framework for usage. In [13], the researchers investigate sensors and the issues involved with the acquisition, storing, and management of humongous amounts of data.

In the arena of healthcare, a lot of IoT studies are focused towards the upkeep and maintenance of patients with certain definitive ailments like diabetes [7] and Parkinson's disease [8]. Such research is carried out with the sole aim of monitoring patients to detect whether they are serious, need attention or medical aid, or need to be transferred to the hospital [9]. In [14], the authors outline a model for discovery of heart attacks by means of off-the-rack apparatuses and a specialized antenna. They utilize an ECG sensor to record heart activity which is then handled by a microcontroller. Similarly, the researchers in [15] also propose a system for ECG monitoring. Authors in [6] propose a scheme that uses IoT for elderly care and monitoring to ensure that the elderly live in the familiar and cozy environment of their homes while at the same time allowing for timely medical intervention if needed. The authors recognized and acknowledged the possible role of machine learning in identifying deviations from routine and raising alarm signals. In [16], ambient assisted living (AAL) is proposed to aid the day-to-day actions of aged individuals self-reliantly. Researchers in [15, 17] outline frameworks for data procurement, handling, and dispersal using IoT for health monitoring and management. In [18], an IoT-enabled framework is outlined that allows for remedial health meetings amid patients in villages and backward areas, healthcare professionals, and consultants in Nigeria.

In [19], the authors propose a m-IoT arrangement for continuous and instantaneous recording and management of sugar levels of individuals. In this scheme, data from patient sensors is made available to reliable healthcare providers using the Internet. Similar research for blood glucose-level monitoring and management is outlined by authors in [20, 21]. Furthermore, a common IoT-based health statistics procurement system which may also be utilized for glucose-level monitoring is proposed in [22].

Monitoring of an individual's blood pressure (BP) is also an essential task as unmonitored and elevated BP levels may lead to strokes or coronary heart disease. IoT-based 24-h BP care is carried out in [23]. An encouraging setting in which BP should be continuously measured is shown by displaying the transportation assembly amongst a well-being post and the well-being center in [24]. In [25], the researchers report how the Withings BP equipment functions are contingent on the linking to an Apple portable computation device. A gadget for amassment of blood pressure statistics and broadcasting these via an IoT system is outlined in [26]. The proposed system consists of a BP kit along with a transportation unit. In [27], the authors depict an IoT-enabled place-intelligent unit for carry-on BP-level recording and management.

A body temperature and recording scheme constructed over a home gateway is depicted in [28]. In this scheme, body temperatures are transferred using infrared radiation. In [29] also, the authors illustrate a similar temperature recording and management system that also makes use of the Internet of Things. In this scheme also, the major components are the RFID unit and the unit for temperature recording. In [30], the authors present an IoT-enabled low-power and affordable pulse oximeter for distant patient supervision. Similarly, a unified pulse oximeter for telemedicine submissions is defined in [31].

Based on the exhaustive literature review, it can be summarized that most of the IoT-based healthcare systems are specific to a particular healthcare application like BP monitoring, ECG monitoring, and temperature recording. Further, for an effective IoT-based healthcare system, it is of utmost importance to have good-quality sensors and cloud storage having the ability to store large bulks of sensor data being transmitted continuously [11, 13, 32].

Authors in [33] conceptualized the design of a smart cradle with a real-time baby monitoring system over IoT, planned and fabricated to monitor a baby's vitals, like crying condition, humidity, and ambient temperature. In [34], the authors proposed a system for IoT-based monitoring of parameters of infants and neonatal as delayed detection of any diseases in infants may become critical and lead to even deaths. An IoT-based system for scrutiny of infant activities with video enhancement and instant notifications for better baby monitoring with a real-time database in the cloud using accurate sensors for parental supervision is designed in [35]. In [36], the authors proposed an intelligent system for real-time remote baby cradle supervision based on the "Raspberry Pi 3 B +" card, a pi camera, a sound, and temperature sensors. In addition, the authors used a convolutional neural network to identify and interpret the baby status in its cradle and increase the efficiency of the system.

Through this work, we propose a novel IoT-based healthcare system for end-to-end infant and elderly, care, and a general archetypal which can be translated and functionally translated to all IoT-enabled healthcare frameworks. The major contributions of this work, which make it novel, are:

1. An end-to-end model for distant supervision of both infant and elderly healthcare, which was earlier missing in the literature.
2. The integration of secure cloud architecture and machine learning in the framework is an additional feature that helps monitor the vitals of babies and the elderly and raises alarm in case of any abrupt changes, thereby providing protection against and saving from potential life-threatening situations. The medical resource management layer is an intermediary supplementary supporting layer that is responsible for the management and organization of medical records in an efficient manner and implementation of ML to derive insights from the same and the smart medical service layer is connected to specialized health services like hospitals, emergency and trauma centers, and medication supply chain.
3. This work can be especially beneficial for the working class for remote supervision of both their infants and elderly while sitting in the comfort of their workplace and hence helping them focus on their work without worrying about menial supervision tasks.

3 Proposed System Architecture

IoT is the most preferred technology when coming to establishing inter-device communication. The proposed wearable system is organized using various sensors which are lightweight and consume low power providing insights on physiological as well as outer parameters. The device makes the use of wireless IoT system conventions expediting the procedure of delivering data over cloud servers located in distinct areas and fostering mobility. For achieving efficiency this device has been bifurcated into two segments, namely the baby care system and the elderly care system both working on a pivotal microcontroller, the Raspberry Pi-3. This microcontroller is available with an external storage capacity, its own operating system, and programmable abilities to communicate with both the software and hardware.

3.1 Baby Care System

The baby care system manifests four tasks, namely live temperature control and detection; switching on the fan and music system when required; spotting wet diapers; and bringing and reporting unusual activities coming into notice. The

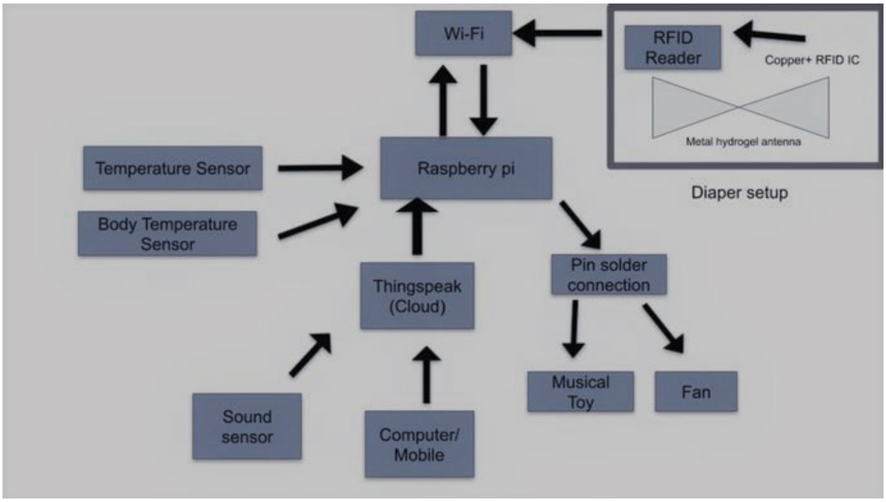


Fig. 3.2 Baby care system architecture

complete system design is laid out in Fig. 3.2. This device gathers data with the use of different sensors. The temperature of the room and infant is recorded via a temperature sensor and body-sensing IR sensor, respectively, which is subsequently transmitted to the microcontroller for processing further and in the process also raising required alerts to custodians via the services of SMS gateways.

The guardian's mobile phone controls the fan and music devices. Musical toy (device) acts as a device to divert the baby's attention while crying and induce sleep. VNC, an application available for Raspberry Pi, connects the user's mobile phone with a microcontroller for fetching the data into a personal device. The central commanding units of the system include the microcontroller and the personal mobile device, thereby acting as mini-warehouses and processing sensory input data received, and then yielding a meaning in the form of output. Furthermore, ThingSpeak cloud collects the information, passing it via the MQTT server that triggers the interconnected actuators, regulating the fan and music device. The data from the microcontroller is sent to the ThingSpeak cloud every 15 s.

Underlying sound sensors notice baby cries and help in the detection of unusual events, thus alerting guardians to oversee.

3.2 Elderly Care System

The features of elderly care system's design consider all prerequisites that intersect daily in the lives of infants, the elderly, and differently abled. High portability of elderly system ensures one solution to multiple applications: routine insulin check, stress-level monitoring, wet diaper alerts, and notifying for any fall from a place.

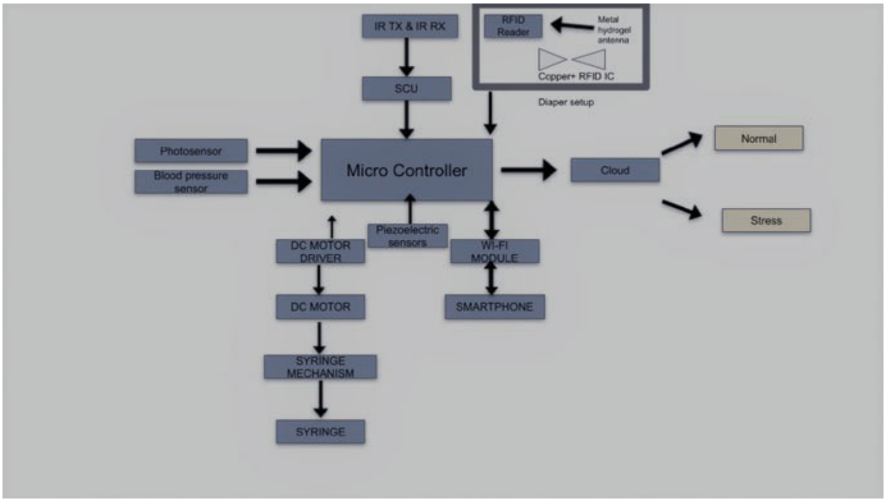


Fig. 3.3 Elderly care system architecture

The use of inbuilt Wi-Fi modules enhances a smoother communication between the cloud and devices. The use of MQTT cloud provides all the data in a CSV file, helping this system become smarter and "learn." Learning here means adjusting interpretation and performance of the system, and setting up the parameters in convenience to the guardian's needs and inputs. The device records all the feedback through the input. The brief organization of this device is given in Fig. 3.3.

The system covers the following features:

3.2.1 Insulin Detection

This setup consists of two sensors, infrared transmitter and infrared receiver. The concentration of glucose in the bloodstream exhibits a correlation with the intensity value of the wavelength parameter of the given radiation. After that the infrared sensor is connected to the SCU. This unit is used to communicate with the microcontroller. Post-examination by the microcontroller, which is a Raspberry Pi in the given case, would examine the glucose values and send the value to the smartphone. The microcontroller operates an infusion pump through a smartphone and a DC motor. The gears are made up of brass, which ensures higher durability and can resist wear and tear. There are steel spindles, which increase the movement and inscribe momentum in this mechanism.

3.2.2 Fall Detection

Falls in the elderly pose a major threat to the well-being of their body. Detection of falls can be ensured by using two piezoelectric sensors. This sensor works on the basic principle of emitting an electric field when a stress (mechanical) is exerted on the device. The application of this sensor as opposed to its conventional use of having it on the floor is through a simple observation; the phenomenon of elderly fall takes place when the subject falls with their two hands.

3.2.3 Stress Detection

This application takes into account the data from two sensors, namely, sensors that detect heart rate and sensors that detect blood pressure. The readings of the two sensors are interpreted in conjunction for stress detection. If both seem to be elevated and higher than normal, then a person can be concluded to be stressed. The blood pressure and heart rate sensors are further sent through a machine learning code in the Raspberry Pi microcontroller. This data is further processed, including morphological, band-pass, and median filters. The features are extracted through discrete wavelet transform. The classification of stress is determined through a deep neural network including the layers of:

(a) Support vector machine
(b) Decision trees
(c) Bayesian network
(d) Artificial neural network

Decision scores based on these parameters yield the stress of the person on the basis of heart rate and blood pressure.

3.2.4 Diaper Wetting Detection

For diaper wetting detection, the diaper is embedded with an RFID sensor, which is within a 1-m read range. Along with an RFID sensor, a superabsorbent polymer is under a cellulose-derived material. A combination of this significantly reduces the leakage. This is further protected through water-resistant layers. This sensor setup factors into the:

(a) Signal strength
(b) Battery activation through urine

3.3 Description of the Components Used

The proposed system architecture, as described in Sects. 3.1 and 3.2, uses the following four components, namely, microcontrollers, sensors, Wi-Fi/Internet, and mobile phones. As everyone is familiar with Wi-Fi/Internet and mobile devices, we describe in this section the microcontroller and the sensors used.

3.3.1 Microcontroller

A pivotal part of any IoT device is a microcontroller. Acting as both the nervous system and brain, it supports input/output programmable pins. In the proposed architecture, we have used the Raspberry Pi microcontroller. Owing to an inbuilt processor, storage space on a portable SD card, it can act as a computer in a small board. Connection with multiple devices via HID ports makes it easier to operate with external mouse and keyboards. Figure 3.4 shows a Raspberry Pi microcontroller.

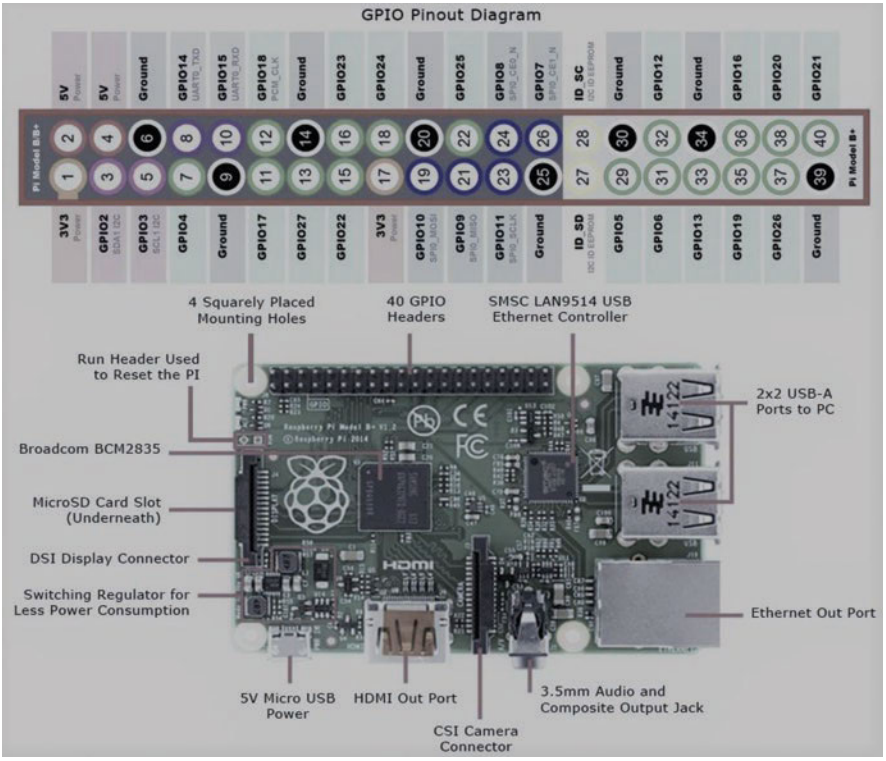


Fig. 3.4 The Raspberry Pi microcontroller

3.3.2 Sensors

A number of different sensors are used in the proposed framework, each of which is illustrated and explained as follows:

1. DS18B20 temperature sensor module: We are using this sensor, as shown in Fig. 3.5, in our system to detect the room temperature where the baby has been kept. This sensor has a precision of up to 0.0625 °C.
2. MLX90614 ESF noncontact human body infrared temperature measurement module: This infrared sensor, as shown in Fig. 3.6, is used for the detection of noncontact thermal temperatures of the human body and is vital in obtaining movement detection.
3. Grove sound sensor: This high-sensitivity microphone-based sensor, shown in Fig. 3.7, detects any sound activity in the surroundings. It is based on the LM386 amplifier.
4. Hydrogel-based RFID tags: The RFID reader senses the battery activation of the electrodes present in the diaper setup as elicited in Figs. 3.2 and 3.3. The RFID sensors were connected via the Internet, which shot a notification given the activation. Hydrogel-based RFID tags are as shown in Fig. 3.8.
5. Infrared transmitter and receiver: Infrared transmitter and infrared receiver used in insulin detection are shown in Fig. 3.9.
6. Piezoelectric sensor: This sensor detects tactical changes and converts the senses to electrical signals. It can be used for fall detection. It is represented in Fig. 3.10.
7. Pulsometer: Fig. 3.11 shows a pulsatometer that measures the heart rate through volumetric changes in the blood flow. This sensor detects light absorption by hemoglobin, which indicates the use of optical technology in detecting the heart rate of the subject.
8. Sensors that detect blood pressure: These sensors output mean artificial pressure and consist of a complex circuit exhibiting the operations of a no-pass filter, high-pass filter, and non-inverting amplifier. These sensors output the value of blood pressure, which includes the equation in which voltage is a parameter involved, maximum pressure, and minimum pressure according to the biological values. Blood pressure monitoring sensor is shown in Fig. 3.12.

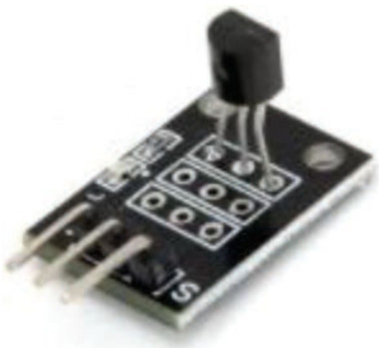


Fig. 3.5 The DS18B20 temperature sensor

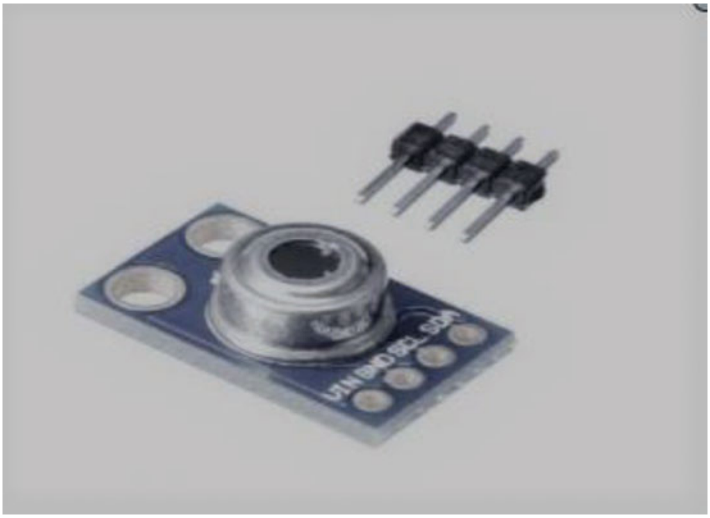


Fig. 3.6 The MLX90614 ESF temperature sensor

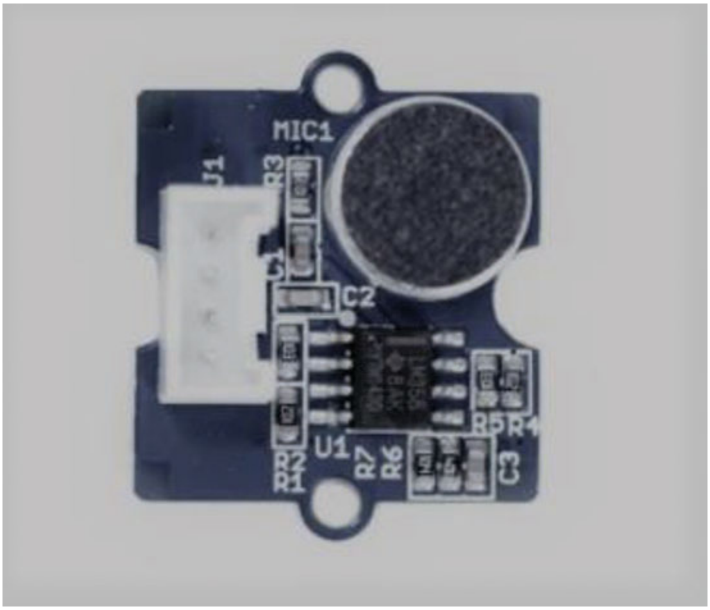


Fig. 3.7 The grove sound sensor

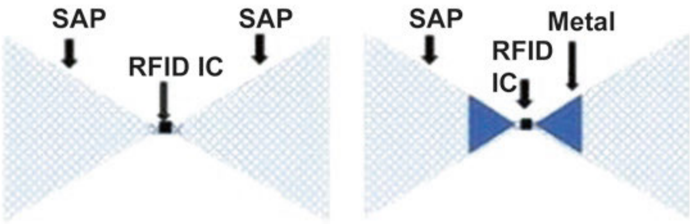
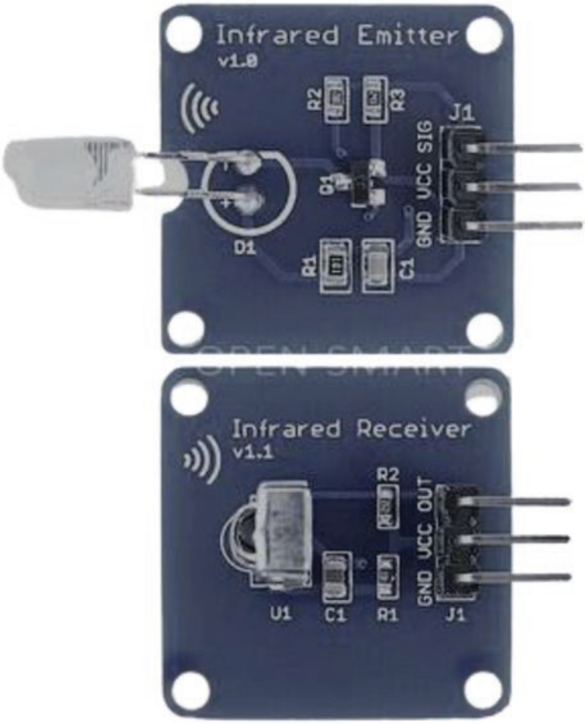


Fig. 3.8 Hydrogel-based RFID tags: Sensor design illustration used as reference for sensor fabrication. (L): Only Hydrogel based tags, (R): hydrogel-metal hybrid tags

Fig. 3.9 Infrared transmitter and receiver



3.4 *Secure Cloud Storage Architecture and Machine Learning*

Health statistics amassed from the users need to be warehoused securely for constant and sustained use. Medics profit immensely on or after being aware of an affected patient's past tendencies, ailments, and medical records, and supervised machine learning (ML) algorithms do not work well unless they have large volumes of annotated data for training. As the data volumes may be supremely high for IoT-enabled healthcare architectures, cloud storage seems to be the sole feasible

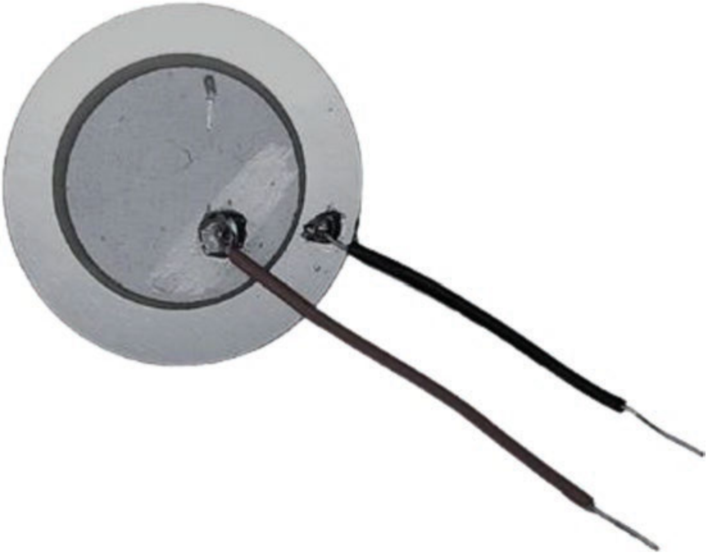


Fig. 3.10 Piezoelectric sensor

Fig. 3.11 Pulsatometer

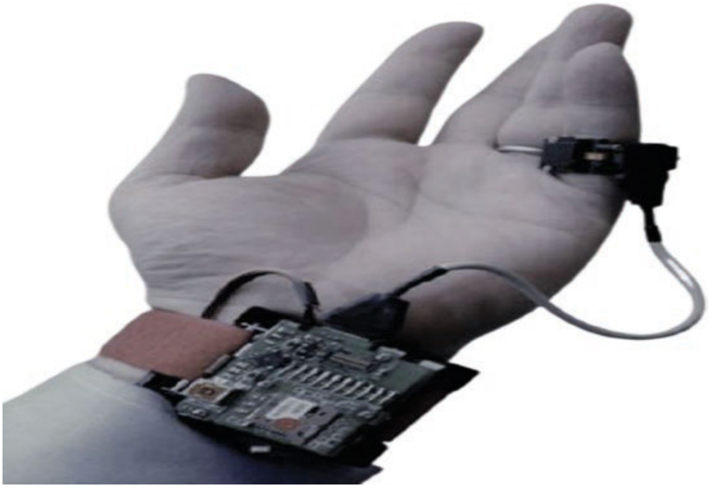


Fig. 3.12 Blood pressure monitoring sensor

technique for stowage of the observed healthcare statistics of patients year after year. Nevertheless, ensuring availability for healthcare specialists without trading off on data safety is a crucial concern [37, 38] which needs to be talked about and taken into consideration by investigators propositioning emerging IoT-enabled healthcare systems. Furthermore, the use of machine learning algorithms has been continually stressed in literature as an untapped avenue for refining the quality of the existing healthcare architecture [6–8]. ML provides a yet untapped possibility of recognition and spotting of previously unknown tendencies and drifts in health statistics data, delivery of appropriate action, tactics, diagnostics, and recommendations to healthcare specialists which are explicit to distinct patients. Therefore, there needs to be a lot of emphasis on cloud storage architectures, intended to provision the execution of ML algorithms on big data.

Numerous types of data handling are possible using cloud-based frameworks; however, the utmost pertinent and applicable amongst all of them are computational off-loading and machine learning. Computational off-loading encompasses the use of the cloud to accomplish intricate information dispensation outside the scope of low-resource habiliment equipment. By transferring raw or somewhat handled sensor statistics to the cloud, the computational capabilities of numerous machines may be exploited for dispensation. Using cloud-based dynamic computational setting as opposed to processing on the stand-alone mobile devices offers countless rewards; further intricate processes may be implemented, outcomes may be attained faster, and mobile battery life may be elongated as a consequence of lesser processing occurring internally.

Complicated sensor nodes should profit significantly by the use of computational off-loading. For instance, ECGs have a typical outline, and diverse nonconformities to this shape may be an indication of a serious condition in a user that needs attention and immediate medical aid. A small, low-powered sensor node does not have the capacity to transmit ECG readings rapidly and uses ML algorithms to regulate users' health.

Machine learning when applied to user health statistics can aid in attaining meaningful evidence, finding formerly unidentified associations amongst indicators and ailments, defining likely diagnosis founded on the diagnosis of past patients, and developing appropriate action plans tailored for individual patients based on what worked for analogous patients in the past, etc. ML helps users receive the most appropriate care by eliminating the risk of human errors.

In the proposed framework, we can integrate any of the popular supervised ML algorithms like support vector machines, naïve Bayes classifier, and decision trees to spot nonconformities or deviations from trends using annotated data from past medical records for training. The system can send urgent notifications to the caretakers' mobile device or to emergency centers from anywhere in such alarming circumstances. The architecture of the proposed model is as outlined in Fig. 3.13.

As shown in Fig. 3.13, at the lowest level is the sensor data-collecting layer that comprises different sensors for data acquisition and transmitters for data communication and multicast. The medical resource management layer is an intermediary supplementary supporting layer that is responsible for the management and organization of medical records in an efficient manner and implementation of ML to derive insights from the same. In this layer, cloud computing and computational off-loading are available to health and life science providers, thereby comprising an efficient means for information analytics, safety, and privacy.

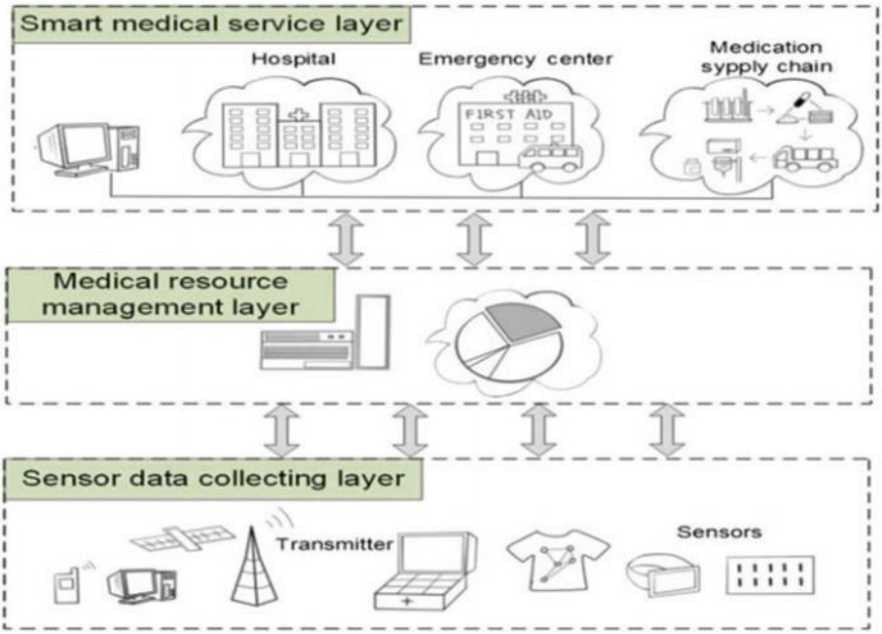


Fig. 3.13 Network architecture

Lastly, there exists a smart medical service layer which is straightaway connected to specialized health services like hospitals, emergency and trauma centers, and medication supply chains. They can exploit this three-layered architecture to view a patient's past medical records and make meaningful analysis and conclusions from this and prescribe the course of further analytics as was also shown in [39].

4 Conclusions and Future Scope

The world will see 35 billion IoT device installations by the end of 2021 and numerous wearable health devices. In this study, we proposed the solution for the infant age group of less than a year and elderly age group of 60+. In the future, we plan to bring more features by thoroughly examining young children between the age group of 1–9 and elders 60+ age group. We plan to offer consolidated and more combined solutions, in order to serve a wider demographic. This can include features for middle-aged people, including detection of diabetes using RFID. Further, we plan to carry out a more detailed resolution on disease-based wearables.

Another major future research direction also involves the integration of deep neural networks to detect deviation from trend and to improve the overall efficiency of the proposed system. Health reports generated using deep learning models can facilitate important decision-making to improve the current condition of lifestyle diseases in the target demographic. In insulin detection, the user receives input on a smartphone device from the microcontroller; this decision can further be attuned by the deep learning models and the process can be fully automated. Long-term application of this can result in noticing behavioral patterns of infants and elders and thus providing better recommendations for their daily routine.

References

1. Gartner, IT Glossary, Internet of Things. http://www.gartner.com/it-glossary/internetof-things/
2. IERC – European Research Cluster on the Internet of Things. *Internet of Things – Pan European research and innovation vision*, October, 2011.
3. ITU Internet Reports, The Internet of Things, November 2005. http://www.itu.int/osg/spu/publications/internetofthings/InternetofThings_summary.pdf
4. Adori, L., Iera, A., & Morabito, G. (2010, May). The Internet of Things: A survey. *ScienceDirect: Computer Networks, xx*(Article in Press), 1–19. https://doi.org/10.1016/j.comnet.2010.05.010
5. Gope, P., & Hwang, T. (2016, March). BSN-care: A secure IoT-based modern healthcare system using body sensor network. *IEEE Sensors Journal, 16*, 1368–1376.
6. Zhu, N., et al. (2015, July/August). Bridging e-health and the Internet of Things: The SPHERE project. *IEEE Intelligent Systems, 30*(4), 39–46.
7. Chang, S.-H., Chiang, R.-D., Wu, S.-J., & Chang, W.-T. (2016, May/June). A context-aware interactive M-health system for diabetics. *IT Professional, 18*(3), 14–22.

8. Pasluosta, C. F., Gassner, H., Winkler, J., Klucken, J., & Eskofier, B. M. (2015, November). An emerging era in the management of Parkinson's disease: Wearable technologies and the Internet of Things. *IEEE Journal of Biomedical and Health Informatics, 19*(6), 1873–1881.
9. Fan, Y. J., Yin, Y. H., Xu, L. D., Zeng, Y., & Wu, F. (2014, May). IoT-based smart rehabilitation system. *IEEE Transactions on Industrial Informatics, 10*(2), 1568–1577.
10. Islam, S. M. R., Kwak, D., Kabir, H., Hossain, M., & Kwak, K.-S. (2015). The Internet of Things for health care: A comprehensive survey. *IEEE Access, 3*, 678–708.
11. Dimitrov, D. V. (2016, July). Medical Internet of Things and big data in healthcare. *Healthcare Informatics Research, 22*(3), 156–163. [Online] Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4981575/.
12. Poon, C. C. Y., Lo, B. P. L., Yuce, M. R., Alomainy, A., & Hao, Y. (2015). Body sensor networks: In the era of big data and beyond. *IEEE Reviews in Biomedical Engineering, 8*, 4–16.
13. Yin, Y., Zeng, Y., Chen, X., & Fan, Y. (2016, March). The Internet of Things in healthcare: An overview. *Journal of Industrial Information Integration, 1*, 3–13. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2452414X16000066.
14. Wolgast, G., Ehrenborg, C., Israelsson, A., Helander, J., Johansson, E., & Manefjord, H. (2016, October). Wireless body area network for heart attack detection [education corner]. *IEEE Antennas and Propagation Magazine, 58*(5), 84–92.
15. Agarwal, R., & Sonkusale, S. (2011, October). Input-feature correlated asynchronous analog to information converter for ECG monitoring. *IEEE Transactions on Biomedical Circuits and Systems, 5*(5), 459–468.
16. Dohr, A., Modre-Osprian, R., Drobics, M., Hayn, D., & Schreier, G. (2010). The Internet of Things for ambient assisted living. In *Proceedings of the 7th international conference on information technology: New generation*, pp. 804–809.
17. Wang, L., Yang, G.-Z., Huang, J., Zhang, J., Yu, L., Nie, Z., et al. (2010, April). A wireless biomedical signal interface system-on-chip for body sensor networks. *IEEE Transactions on Biomedical Circuits and Systems, 4*(2), 112–117.
18. Adewale, O. S. (2004, June). An internet-based telemedicine system in Nigeria. *International Journal of Information Management, 24*(3), 221–234.
19. Istepanian, R. S. H., Hu, S., Philip, N. Y., & Sungoor, A. (2011, August/September). The potential of Internet of m-health Things 'm-IoT' for non-invasive glucose level sensing. In *Proceedings of the IEEE annual international conference of the engineering in medicine and biology society (EMBC)*, pp. 5264–5266.
20. Guan, Z. J. (2013, March 27). Somatic data blood glucose collection transmission device for Internet of Things. Chinese Patent 202 838 653 U.
21. Wei, L., Heng, Y., & Lin, W. Y. (2012, 7 March). Things based wireless data transmission of blood glucose measuring instruments. Chinese Patent 202 154 684 U.
22. Lijun, Z. (2013, June 5). Multi-parameter medical acquisition detector based on Internet of Things. Chinese Patent 202 960 774 U.
23. Dohr, A., Modre-Opsrian, R., Drobics, M., Hayn, D., & Schreier, G. (2010, April). The Internet of Things for ambient assisted living. In *Proceedings 7th international conference on information technology, new generation (ITNG)*, pp. 804–809.
24. Puustjarvi, J., & Puustjarvi, L. (2011, May). Automating remote monitoring and information therapy: An opportunity to practice telemedicine in developing countries. In *Proceedings of the IST-Africa conference* (pp. 1–9).
25. Tarouco, L. M. R., et al. (2012, June). Internet of Things in healthcare: Interoperatibility and security issues. In *Proceedings of the IEEE international conference on communication (ICC)*, pp. 6121–6125.
26. Guan, Z. J. (2013, March 27). Internet-of-Things human body data blood pressure collecting and transmitting device. Chinese Patent 202 821 362 U.
27. Xin, T. J., Min, B., & Jie, J. (2013, April 17). Carry-on blood pressure/pulse rate/blood oxygen monitoring location intelligent terminal based on Internet of Things. Chinese Patent 202 875 315 U.

28. Jian, Z., Zhanli, W., & Zhuang, M. (2012, December 5). Temperature measurement system and method based on home gateway. Chinese Patent 102 811 185 A.
29. In, Z. L. (2014, February 12). Patient body temperature monitoring system and device based on Internet of Things. Chinese Patent 103 577 688 A.
30. Larson, E. C., Goel, M., Boriello, G., Heltshe, S., Rosenfeld, M., & Patel, S. N. (2012, September). SpiroSmart: Using a microphone to measure lung function on a mobile phone. In *Proceedings of the ACM international conference on ubiquitous computing*, pp. 280–289.
31. Larson, E. C., Goel, M., Redfield, M., Boriello, G., Rosenfeld, M., & Patel, S. N. (2013, January). Tracking lung function on any phone. In *Proceedings of the ACM symposium computing for development*, Art. ID 29.
32. Xu, B., Xu, L. D., Cai, H., Xie, C., Hu, J., & Bu, F. (2014, May). Ubiquitous data accessing method in IoT-based information system for emergency medical services. *IEEE Transactions on Industrial Informatics, 10*(2), 1578–1586.
33. Jabbar, W. A., Shang, H. K., Hamid, S. N. I. S., Almohammedi, A. A., Ramli, R. M., & Ali, M. A. H. (2019). IoT-BBMS: Internet of Things-based baby monitoring system for smart cradle. *IEEE Access, 7*, 93791–93805. https://doi.org/10.1109/ACCESS.2019.2928481
34. Nivetha, B., & Sriram Manish Kumar, E. (2020). Iot based nicu baby healthcare monitoring system. *Materials Today: Proceedings, 33*(Part 7), 4837–4841, ISSN 2214-7853. https://doi.org/10.1016/j.matpr.2020.08.393
35. Rekha, P., & Suganya, K. (2020, March). Smart baby monitoring cradle system using IOT. *International Journal of Innovative Research in Science, Engineering and Technology, 9*(3), 408–414.
36. Cheggou, R., Mohand, S. S. H., Annad, O.. & Khoumeri, E. H. (2020). *An intelligent baby monitoring system based on raspberry PI, IoT sensors and convolutional neural network*. 2020 IEEE 21st international conference on information reuse and integration for data science (IRI), pp. 365–371. https://doi.org/10.1109/IRI49571.2020.00059.
37. Olaronke, I., & Oluwaseun, O. (2016, December). Big data in healthcare: Prospects, challenges and resolutions. In *Proceedings of the Future Technologies Conference (FTC)*, pp. 1152–1157.
38. Zhou, J., Cao, Z., Dong, X., & Vasilakos, A. V. (2017, January). Security and privacy for cloud-based IoT: Challenges. *IEEE Communications Magazine, 55*(1), 26–33.
39. Natarajan, K., Prasath, B., & Kokila, P. (2016, March). Smart health care system using Internet of Things. *Journal of Network Communications and Emerging Technologies (JNCET), 6*(3), 37–42..

Chapter 4
IoT in Healthcare: *A 360-Degree View*

Rishika Mehta, Kavita Khanna, and Jyoti Sahni

Abstract The proliferation of the Internet of Things (IoT) has generated immense possibilities in the healthcare domain. IoT has the potential to transform the healthcare sector by changing its current focus from curative approach to ensuring complete wellness of an individual. However, this domain is still in infancy, and a number of aspects must be examined before its full potential can be realized. In order to assist the researchers, this chapter provides a 360-degree view of IoT around the healthcare domain. The chapter also presents a distant view as well as in-depth study of the five-layered IoT architecture with reference to the healthcare domain; compares a number of communication protocols for IoT-based applications on the basis of their data rate, range coverage, energy requirements, cost, and other notable parameters; discusses the role of cloud as well as fog for processing a large amount of data collected through an IoT ecosystem and their associated security aspects; discusses the applicability of IoT in preventive and curative healthcare with respect to various operational areas including disease monitoring, age-based monitoring, physical abnormality monitoring, and profile-based monitoring; and presents open research questions, as well as future research directions in relation to the use of IoT in healthcare.

Keywords Internet of Things · Healthcare · Analytics · Fog computing

R. Mehta (✉)
Department of CSE & IT, The NorthCap University, Gurugram, India

K. Khanna
Campus Director, Delhi Skill and Entrepreneurship University, New Delhi, India
e-mail: kavita.khanna@dseu.ac.in

J. Sahni
School of Engineering and Computer Science, Victoria University of Wellington, Wellington, New Zealand
e-mail: jyoti.sahni@ecs.vuw.ac.nz

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*, Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_4

1 Introduction

Traditional devices, when provided with the capability of computing and connectivity, give rise to the Internet of Things (IoT). IoT has the highest potential among all the cutting-edge upcoming technologies in transforming the way different organizations work. The regular devices, when connected to the Internet, become smart by being able to send or receive information or both. Smart devices communicate with each other to automate almost everything around us ranging from the recording of minute activities to accomplishing complex tasks. The curious researchers can find the most important aspects of IoT in [1–3].

One of the major areas where IoT promises to deliver value-based services is healthcare [4]. IoT has the power to revolutionize the healthcare sector by changing its current focus from a curative approach to ensuring complete wellness and well-being of an individual. Through advancements in data sensing and communication technology, it is now feasible to capture an individual's data and monitor it remotely. This facilitates comprehensive patient care by enabling quick diagnosis of sensitive complications and taking prompt actions for improvising the patient's existing vital signs or medical conditions.

It must, however, be noted that healthcare services can never be generalized. Each one of us has different body type with contrasting habits and varied medical histories. Two persons suffering from the same ailment might need to be cured with different prescriptions and treatments. In addition to individual differences, there may be variations due to age, profession, physical condition, and medical history such as chronic ailment. IoT in healthcare can cater to all these variations by taking insights from the recording of vital parameters of patients and initiating timely care through remote monitoring. This makes patients self-sufficient and at the same time offers support to doctors for taking quick actions. This chapter gives a detailed view of IoT in healthcare with the objective to present a comprehensive study on the underlying architecture and application scenarios, and lists the associated challenges. The rest of the chapter is organized as follows. Section 2 discusses the general scenario for the application of IoT in healthcare. Section 3 presents a layered IoT healthcare architecture entailing various health-specific sensors, technologies, and protocols that enable the development of IoT-based healthcare solutions. Section 4 identifies different healthcare application areas and discusses related work in these application areas, associated challenges, and the role IoT can play in yielding a prospective solution and finally, Section 5 concludes the work.

2 IoT Healthcare Architecture: A 1000-Feet View

IoT in healthcare offers a number of applications ranging from remote monitoring of patients to asset tracking in hospitals. It has the ability to keep individuals informed about their health vitals while also assisting physicians in improving the

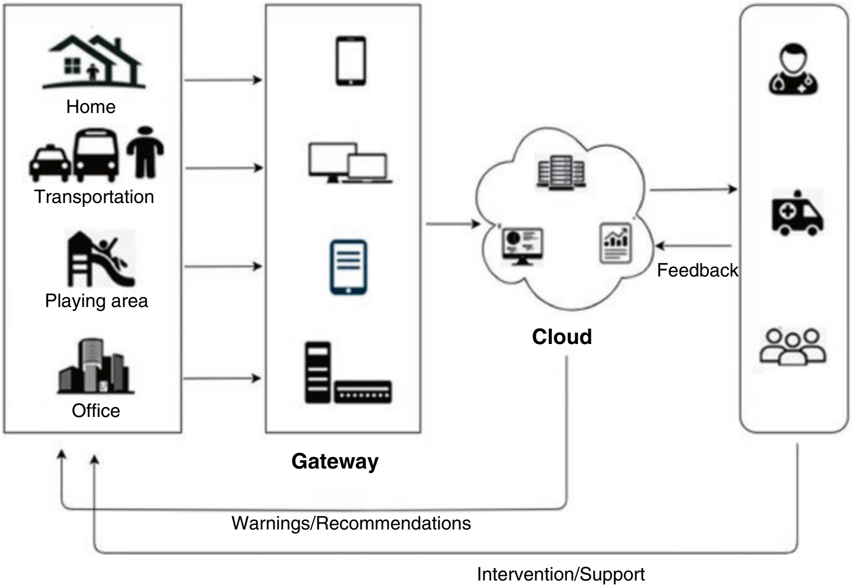


Fig. 4.1 A general IoT healthcare scenario

services they provide. Figure 4.1 explains how IoT can transform the traditional healthcare sector to enable on-the-go healthcare services.

The underlying idea is that a patient need not be in a hospital to get his or her health checkup done; instead, he or she is free to travel anywhere. The patient's health data, read through the sensors placed on or around his or her body, is analyzed on the other end of the associated IoT network, i.e., fog devices or cloud. For in-depth analysis and storage, the acquired data is sent to the cloud via gateway, e.g., smartphone. After the analysis, the doctor can send appropriate instructions to the patient and other family members in the loop. A detailed IoT healthcare architecture is discussed in the next section.

3 IoT Healthcare Architecture: A Closer View

The implementation of IoT-based solutions for various health-related concerns requires a scalable and secure IoT architecture in place. This section discusses healthcare-specific IoT architecture in detail. The architecture is divided into five layers: sensory layer, communication and connectivity layer, gateway layer, analytics layer, and user application layer as shown in Fig. 4.2. This architecture is a practical design of how an IoT healthcare framework should look like when designing a real-time healthcare solution. It is critical to acquire raw data correctly, but it is equally critical to mine the collected data for finding correlations. The first layer

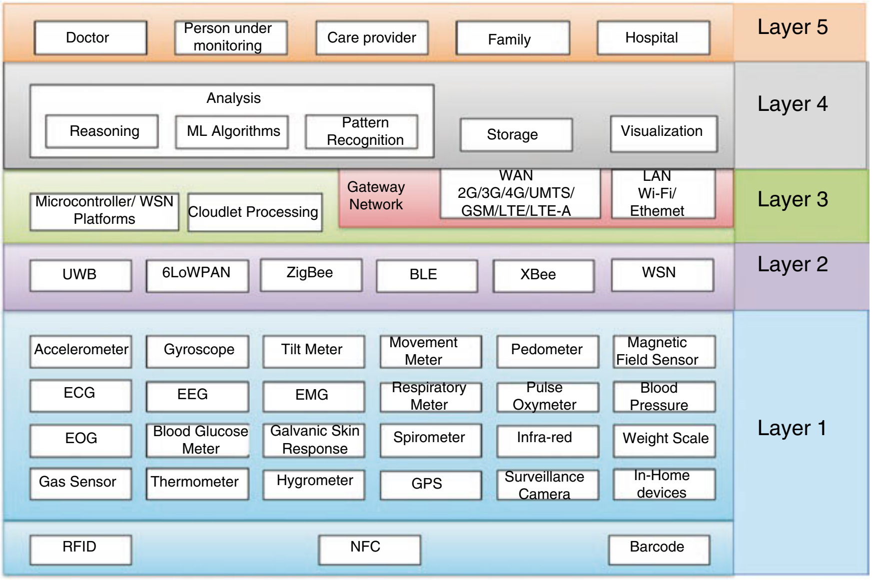


Fig. 4.2 IoT healthcare-layered architecture

consists of various sensors and tags for capturing the health information and tracking the objects, respectively. As the data is captured, it is sent for processing in real time. Data processing may involve a number of tasks. Based on the latency sensitivity of a task, it can be processed at the gateway layer or it can be sent to the cloud for further processing and subsequent storage. The processed results are transmitted to the end users, namely patients, their relatives, and doctors, as applicable.

The objective is to maintain the desired latency of the different tasks with the minimum possible cost in terms of power consumption, bandwidth, and other computational resources. The following section discusses each of the layers in detail.

3.1 Sensory Layer

The bottom-most layer is known as sensory/perception or device layer. This tier, being imminent to the user, deals with the acquisition of required data from the person under monitoring [5–7]. It may employ varied sensors based on the parameters that need to be monitored including motion recognition sensors for measuring physical activities, biosensors (wearable or implantable) for capturing physiological vitals, environment sensors, and location sensors [8, 9]. We have presented below a tabulation of a variety of sensors applicable to the healthcare domain.

The sensors use low power, show lesser data rate connectivity, and have limited or no ability to process data. Tags in this layer are used for identification and tracking of the objects (Table 4.1).

Table 4.1 Sensors and their description

Sensors	Description
Motion recognition sensors	
Accelerometer	Senses linear movement
Gyro meter	Measures angular velocity
Tiltmeter	Senses tilting
Movement meter	Records body movement
Pedometer	Measures step count
Magnetic field sensor	Senses linear and angular position
Biosensors	
Electrocardiogram (ECG)	Measures electrical activity of heart
Electroencephalogram (EEG)	Captures electrical activity of brain
Electromyography (EMG)	Measures electrical activity of muscles
Respiratory meter	Measures lung activity
Pulse oximeter	Measures dissolved oxygen in blood
Blood pressure meter	Calculates blood pressure
Blood glucose meter	Measures dissolved glucose in blood
Galvanic skin response	Records surface temperature of the skin
Electrooculography (EOG)	Captures eye movement
Spirometer	Measures in-and-out movement of air in the lungs
Infrared	Measures blood flow
Weight scale	Measures body weight
Environment sensors	
Gas sensor	Monitors air quality
Thermometer	Measures in/out temperatures
Hygrometer	Measures in/out humidity
Location sensors	
GPS	Used for location tracking
In-home sensors	Used for in-home assistance

Radio-frequency identification (RFID), barcodes, and near-field communication (NFC) form an integral part of tags in the bottom-most layer of IoT healthcare architectural framework. All the gathered information is then processed by the upper layers for appropriate decision-making about the patient's health [10].

3.2 Connectivity and Communication Layer

This layer is responsible for transmitting the captured data from the sensory layer to the higher layers through a number of communication protocols. These protocols connect various devices and servers together forming a large IoT ecosystem. A wide variety of protocols may be employed in different scenarios. Prominent parameters that differentiate these protocols include supported data rate, range coverage, energy requirements, cost, reliability, and throughput. Table 4.2 compares the various dominant communication protocols specific to IoT systems based on the above parameters [11–23] and presents their scale of adoption in the market till date.

When implementing an IoT solution, an important decision involves choosing the right communication protocols based on the optimum range of different parameters. A good option would be one that has a faster data rate, a larger range of coverage, lower energy and cost requirements, and more reliability and throughput.

A detailed discussion of a number of communication protocols is presented below. Bluetooth (IEEE 802.15.1), although widely acceptable and commonly used in smartphones, laptops, etc. [24], is not suitable for IoT devices. To cater to the low energy needs of IoT-based devices, a newer version—Bluetooth Low Energy (BLE) or Bluetooth Smart—has been created. BLE modifies the original framework to achieve higher energy efficiency by reducing the cost and overall power consumption while also ensuring a communication range of about tens of meters. It decreases the data rate and lowers the power usage of traditional Bluetooth by 95%. Its scalability and flexibility to adapt to nearly all market use cases form the foundation of IoT communication.

Another protocol applicable in IoT is IEEE 802.15.4 which is a low-cost, low-speed, and low-power WPAN protocol [25]. It has its applications in ZigBee and 6LoWPAN (IPv6 over low-power wireless personal area network) protocols. Topologies like Star, P2P, Mesh, and Cluster Tree Network can be formed in IEEE

Table 4.2 Comparison of various communication protocols

Communication protocols	Data rate	Range	Energy requirements	Cost	Reliability	Throughput	Adoption
BLE	1 Mbps	Moderate (up to 100 m)	Low	Low	High	Moderate	Moderate
ZigBee	250 kbps	Moderate (up to 100 m)	Low	Low	High	High	Moderate
LoRaWAN	50 kbps	High (15 km)	Low	Moderate	High	Low	Moderate
Cellular (4G, 4G LTE)	12 Mbps	High (up to miles)	High	High	High	High	Moderate
NB-IoT	Up to 200 kbps	High	Low	Low	High	Moderate	Increasing
LTE-M	1 Mbps	High	Moderate	Moderate	High	High	Upcoming
Wi-Fi	54 Mbps	Moderate (50 m)	Moderate	Low	Moderate	High	High
Wi-Fi HaLow	347 Mbps	High Up to 1 km	Low	Moderate	High	High	Upcoming
Wi-Fi HEW	11 Gbps	Moderate	Moderate	Moderate	High	High	Upcoming

802.15.4 [26]. 6LoWPAN supports IPv6 packets over IEEE 802.15.4 wireless personal area network by fragmenting IPv6 datagram to fit into much smaller IEEE 802.15.4 frame size. It efficiently connects battery-operated devices to the Internet [27]. It is well suited to form a wireless sensor network and therefore works well in isolated network environments [28]. ZigBee uses a layered architecture for its protocol. It uses IEEE 802.15.4 for its physical and MAC layers. It adds network and application layers to create a four-layer architecture [29]. ZigBee devices allow batteries to be used up to years on primary cells without any chargers. The ZigBee alliance created a universal language, Dotdot, for smart devices enabling them to work together on any network and talk to each other seamlessly at the application layer [30].

Yet another protocol, long-range wide-area network (LoRaWAN) standard wirelessly connects low-powered battery-operated devices to the Internet. It provides long-range transmission with low power consumption. It ensures two-way communication and provides end-to-end security. LoRaWAN provides a single hop distance between the end device and the gateway. It forms star topology unlike most other protocols which use mesh network architecture. In a mesh network, the individual nodes forward the information of other nodes which increases the communication distance which might drain the network capacity and battery life due to irrelevant information passing [31]. Star network architecture, however, makes much more sense for saving the battery life in order to achieve the goal of long-range connectivity [32].

IoT devices need reliability, security, and constant connectivity. Standard Wi-Fi may not be the best choice for many IoT applications; however, some applications like home automation and building environments utilize the installed Wi-Fi. Wi-Fi has limitations both in range coverage and power utilization. To address these shortcomings, two new IoT IEEE specifications have been published: Wi-Fi HaLow (IEEE 802.11ah) and HEW (IEEE 802.11ax) [16] which are suitable for IoT systems.

Wi-Fi HaLow (IEEE 802.11ah) was launched to cope with limited range and higher power consumption issues of standard Wi-Fi. This newer published specification promises to offer better range coverage while ensuring low power consumption. High-efficiency wireless standard HEW (IEEE 802.11ax), on the other hand, ensures higher throughput and data rate while also reducing the latency. It helps the clients to retain their wake time making them power thrifty and capable of avoiding collisions simultaneously [16].

The next protocol is cellular technology, which was designed to offer higher range and bandwidth at the cost of power consumption. It works fine for devices that can be charged often like the smartphones but is not apt for IoT sensors and devices that need to last several months to years on battery [12]. To address this challenge, carriers are moving towards newer cellular technologies: NB-IoT and LTE-M, specifically meant for IoT. However, this can be implemented near cell towers to ensure good connectivity. These technologies ensure low cost, low data rate, and low power consumption paving the way for a plethora of currently cost-restricted IoT applications [33, 34].

As can be seen from the comparison offered above, each communication protocol has its own set of benefits and limitations. An IoT application may require different protocols at different levels of communication. Therefore, there cannot be a single communication protocol to suit all IoT-based applications. Based on the application requirements, appropriate communication protocols may be used for communication at different levels.

3.3 Gateway

This layer is the backbone of IoT architecture. Microcontroller or wireless sensor network (WSN) platforms should be in place for processing the data obtained from the previous layer. Examples of some of the recent WSN platforms are Telos, Tmote sky, Pluto, and SunSpot and the most used ones are Arduino [35] and Raspberry Pi [36]. Cloudlet processing can be used at this layer to generate insights to cater to the needs of time-critical applications. Gateway, which acts as an entry point to another network, transfers the processed data at this layer to higher layers by employing 2G, 3G, 4G, or Wi-Fi connection.

A desktop computer can be used as a cloudlet which can run time-critical tasks and provide local storage. A cloudlet has been implemented in [37], where a home gateway iMedBox is connected to Biopatch (wearable sensor transferring real-time data to iMedBox), iMedPack (ensures medicine compliance), and sensors. The collected information is explicated, stored on iMedBox, and then forwarded to health IoT cloud for detailed analysis. This gateway is capable of analyzing, storing, and displaying the collected information. For in-depth analysis and subsequent storage, the processed data from this layer is sent to the cloud analytics layer.

3.4 Storage and Analytics Layer (SAL)

This layer is responsible for storing and analyzing voluminous data obtained from previous layers. The SAL layer operates through remote locations and involves the use of cloud and fog computing. The high-capacity devices in the cloud allow for the analysis of heterogeneous data and storing of patients' medical histories. Various techniques including reasoning [38], machine learning algorithms [39], and pattern recognition techniques [40] are employed for data analysis. The insights generated in this layer are stored for the long term. This helps to lessen the burden of voluminous data on medical experts and enables them to comprehend the data effectively and effortlessly. The major functionalities of long-term storage, processing, in-depth data analytics, and security are discussed in detail below.

Storage and Processing The data collected by biosensors and various other sensors is transferred to the cloud for long-term storage. Along with collected data, the medical history of the patients is also stored in the cloud data centers. But when it comes to delay sensitive real-time applications, cloud alone is unsuitable due to its high latency, location unawareness, and limited mobility support [41]. Fog, an extension of cloud, improves the quality of service by ensuring low latency and by providing security. It positions the gathered data at the network edge, i.e., one hop away from the user [42]. A clear distinction between cloud and fog on various attributes is given in Table 4.3. Since sensors have limited storage and processing capabilities, cloud provides a solution for various IoT applications with its enormous storage and computing power. But due to its high latency, downtime due to network interruptions, and security issues, vulnerable data needs to be located near the user.

For addressing these challenges, fog was introduced. Fog ensures lower latency, reliable connection, power efficiency, and higher security [43].

As stated in Table 4.3, cloud is centralized whereas fog architecture is distributed. Cloud consists of numerous data centers located far away from the client nodes. Fog has thousands of tiny nodes located close to the client devices. Fog acts as an intermediary between sensing devices and data centers. Without fog, the cloud interacts directly with the client devices, which is time-consuming. Fog aims for short-term analysis to provide instant response, whereas cloud targets long-term and in-depth analysis which is much more powerful than fog in terms of storage and processing power. Cloud may collapse without Internet while fog uses various standards and protocols to ensure lower risk of failure. The demand for ever-increasing connected devices can be satisfied with fog computing as it uses local resources

Table 4.3 Cloud Computing vs. fog computing

Requirements	Cloud computing	Fog computing
Response time	High	Low
Network delay	High	Low
Reduction in the amount of data sent to cloud	No	Yes
Distance between user and storage site	Multiple hops	One hop
Connectivity	Internet	Various protocols and standards
Functioning site	Within Internet	At the edge of local network
Security	Lower	Higher
Data analysis	Long-term	Short-term
Location awareness	No	Yes
Geo-distribution	Centralized	Distributed
Bandwidth consumption	High	Low
Network congestion	High	Low
Probability of attack during transmission	High	Low
Processing capability	High	Limited
Scalability	High	Limited

instead of remote ones, thereby enhancing the performance and minimizing the bandwidth issues [44, 45].

Analytics Due to the healthcare sector demanding effective deployment in real-time setting, much emphasis is now being laid on the analysis and presentation of valuable information from the collected sensor data. A variety of healthcare solutions have been proposed in recent years from neonatal monitoring to elderly care including mapping of different environmental conditions. However, adoption of these systems in the physical world largely depends on the effective analysis of collected heterogeneous data. Raw sensor data needs to be prepared before performing data analysis such as removal of motion artifacts, noise components [14, 46], and interpolation of missing values. After the data preparation, intelligent decision-making techniques need to be exploited in real time to offer various healthcare patient-centric services [47]. The cleaned-up sensor data along with patients' medical history is scrutinized for pattern-based classification. The detected patterns are passed to the recommendation system for probabilistic disease identification so that preventive or curative measures could be initiated on time. The recommendation system acts as a reference model for doctors in order to support complete diagnosis of the health condition.

Pattern-based classification is the process of identifying patterns out of the incoming sensor data and determining abnormal deviations which do not attune to the conventional nature of the data. The obtained noticeable unusual patterns in the corporeal data are of utmost importance in the curative realm. These patterns help physicians to take proactive decisions in a very short duration, specifically in remote monitoring [48]. Abnormalities are detected based on the classification techniques to segregate the data set into regular and irregular classes. For example: SVM [49], Markov models [50, 51], and decision tree [52] are the most commonly used techniques for abnormality detection.

A recommendation system eases the physicians by suggesting probabilistic symptoms or diseases out of a large pool of patients' physiological data. These systems exploit active patients' medical history and patterns identified in the previous step to match with similar cases in the healthcare database by applying reasoning and subject-matter knowledge for making predictions like human experts [53]. Collaborative filtering is the most used technique for predicting the diseases that a patient may contract in the future by finding out the similarities among patient's data and patterns from the database [54]. For massive amounts of continuously streaming data with very large dimensionality flowing at a very high rate, storing and processing become very expensive. Also, since data keeps on changing quickly, perceiving it globally gets very difficult [55]. Spark-based reasoning system is also utilized in making recommendations for medical complications, diagnostics, and alternative prescriptions, all of which are approved by medical experts [56]. These suggestions help physicians to provide better treatment services for time-critical medical conditions, improve diagnosis based on the real-time in-depth analysis and pattern matching, and provide them with alternative curative measures to treat patients. The real-time recommendations

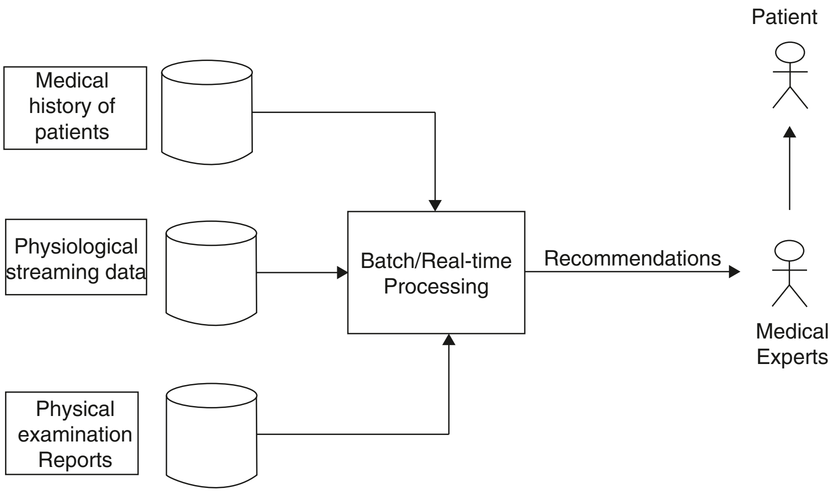


Fig. 4.3 Conceptual view of medical recommendation system

help patients to better understand the fluctuations in their vitals and take preventive measures, thereby avoiding any further complications in their health. Figure 4.3 shows the conceptual view of the recommendation system where a patient's medical history, streaming vital data, and physical reports are processed in parallel for generating recommendations and sending them to the medical professionals.

Security Security and privacy of patients' physiological data are essential for wide adoption of IoT in the healthcare domain. The major security risks are identified in three areas: the first one is the physical layer, where sensor nodes and devices can be tampered with. The second one is the gateway layer, where the gateway responsible for sending the data from the sensors to the cloud and instructions to the sensor nodes may be attacked during data transmission. The last one is the cloud, where the data center, repository of all the acquired and medical data, may be attacked causing irrevocable damage to the system. Typical security and prevention mechanisms including cryptography, secure protocol solutions, and confidentiality support cannot be used due to the dynamic nature of the network, strict resource constraints, and system framework of IoT-based healthcare systems [57]. To deal with potential threats, several solutions have been proposed. Malasri and Wang proposed a secure architecture based on key exchange protocol, symmetric encryption/decryption, and authentication scheme ensuring security, authenticity, and integrity of the sensor network [58]. Moosavi presented end-to-end secure architecture incorporating smart gateways ensuring authentication and authorization [59]. Despite the different efforts being made, security and privacy concerns still remain one of the major hurdles towards adoption of IoT-based solutions [60].

3.5 User Application Layer and Visualization

The results from CAL are transmitted to end users who can be the person under monitoring, relatives, or doctors as applicable. The analysis results of sensory information need to be presented in such a manner that multiple physiological parameters can be correlated to get interesting subsets out of a large pool of data. The physicians need multiple visual representations of the analyzed sensory data for real-time computational analysis for getting insights [61]. Several visualization frameworks have been proposed in the literature. For instance, Sung W. Park et al. developed a visualization framework for continuously streaming data by interpolating scattered data using re-factorization [62]. Daniel classified visualization methods aiding involvement of the human mind for visual data mining of voluminous data sets from the databases based on the data type, technique of visualization employed, and interaction methodologies for users [41]. More attention needs to be given to visualization of streaming data in healthcare for transforming the operational working of the healthcare sector.

4 Healthcare Operational Areas and Role of IoT

Healthcare is one of the prominent domains where IoT can play a significant role. Since healthcare is a generic term for all health-related concerns, it is necessary to identify many operational areas of healthcare in order to apply IoT to various health and well-being related sectors. This section explores these areas and their associated challenges and discusses related work in the literature.

4.1 Disease Monitoring

The Internet of Things is being widely utilized to control the growth of epidemics and monitor chronic diseases. An epidemic is the fast spread of an infectious disease that affects a large number of people in a short period of time. An epidemic spread can be prevented only if the infection is detected and controlled in its infancy which requires capturing near-real-time data followed by its timely analysis. To limit the epidemic spread, Sareen et al. [63] proposed a scheme for identifying Zika virus infected users by using Naïve Bayesian Network. It prevents the outbreak by employing position-based evaluation using Google Maps. Sareen et al. [64] also proposed a cost-effective model based on IoT and cloud computing for preventing the Ebola virus epidemic spread. This model employs RFID and wearable sensors to capture real-time data and identify the infected and uninfected individuals. This model employs a cloud-only approach which is not suitable for real-time scenarios. To reduce the latency and cost incurred in only using cloud computing, fog computing as an intermediate layer between the cloud and the users has been used in [63].

Based on proximal social contacts, Zhaoyang Zhang et al. [65] proposed a cluster-based epidemic control scheme using a smartphone-based body area network which divides the large population into varied clusters. A number of recent advances in this discipline have focused on the definition of models to control the disease at its onset. For example, Sood and Mahajan [66] discussed a novel methodology to control Chikungunya spread at its onset, while Sandhu et al. [67] employed a random decision tree to predict H1N1 infection rate. Most of the recent research works also focus on vector-borne epidemics such as cholera, Ebola, dengue, Kala-Azar, Japanese Encephalitis AES, Japanese Encephalitis JS, and Zika [68, 69].

In India, a large portion of old-age people suffer from chronic diseases. A. Raji et al. [70] proposed an IoT-based monitoring system for identifying the health status of elderly patients with chronic diseases. In some chronic disease cases of aged people staying at home, self-monitoring also proves beneficial. For this, Al-Taee et al. [71] designed an IoT-based mobile health platform for diabetes management of elderly people through which they can access their real-time health data and get feedback if the personalized medical plan is not adhered to. However, more work remains to be done in terms of generating automated health summaries and providing real-time proactive medical recommendations to patients, and this will continue to be a research challenge. Such a system would be especially beneficial for encouraging patients to stick to their treatment schedules, which would aid in the treatment of severe chronic conditions.

4.2 Age-Based Monitoring

IoT can provide a hassle-free experience for a parent to a newborn by providing a data-driven solution that lets parents know the toddler's body temperature, sleep levels, heart rate, breathing rate, and movement levels. Not only this, certain more parameters like wake-up time, sleeping habits, and optimum bedtime habits can also be predicted. A number of smart sensors have been developed to help parents know what their child needs the most [72]. Ashish. B. proposed a cost-friendly design for monitoring premature newborns in temperature-controlled incubators so as to provide immediate care to prevent critical situations [73]. Research in this area is in its nascent stage; very few articles based on IoT-based systems for toddler care are available in the literature. Another challenge is that most of the sensors are wearable and hence may cause inconvenience to the newborns. The use of passive sensors remains a challenge in infant care.

Many children get severe diseases at an early age. Predominant reasons include low birth weight, malnutrition, poor hygiene, and unsafe water and food intake. Childhood obesity, skin infections, pneumonia, and asthma are the most common diseases among children of India [74, 75]. One of the main reasons for childhood obesity is obsessive reliance on junk food due to unhealthy eating habits. IoT can help children to make better food choices by monitoring their calorie intake on a daily basis and showing the ill effects of opting unhealthy and sedentary lifestyles.

Physiological parameters like body temperature and heart rate give a clear description of a child's health, which can easily be measured through IoT-enabled sensor devices. A number of IoT-based research articles contributing to health awareness among children are available in the literature. M. Vazquez-Briseno et al. [76] proposed a mobile health platform to enhance children's health awareness by monitoring their food intake with the help of IoT-based intelligent tags. Fatty children have more chances of developing diabetes, heart diseases, and different types of cancer compared to others. Junaed Siddiquee et al. [77] discussed IoT-based smart asthma attack prediction system concept which helps asthma patients to avoid triggering situations such as the presence of allergens, dry air, and food additives causing breathing trouble by generating warning alerts. Other common childhood illnesses including tuberculosis, malaria, dengue, polio, and pneumonia demand the attention of researchers.

A leap from young-adult to the middle-adult group without any disease is difficult to achieve with the present unhealthy lifestyle. Cardiovascular diseases are most common among adults, which may lead to heart attacks in severe cases. Immediate help is needed for a patient of cardiovascular disease which if not provided may prove to be fatal. Unfortunately, most people do not know whether they suffer from cardiovascular disease until some serious symptoms are recognized. To deal with this, Shao-Jie Hsu et al. [78] developed an IoT- and cloud computing-based early warning system for patients with Coronary Aartery Disease. This system alerts the patients to reduce the activity level when they reach the intensity threshold. Parag Chatterjee et al. [79] discussed the importance of decision support systems for patients with cardiometabolic diseases which help healthcare professionals to deeply analyze the patient's data to tackle the ailment with a better approach. Recommendation tools, however, cannot be general and they are required to be tailored to specific diseases. Designing and integrating recommendation systems that cater to a large set of health related issues remain a challenge (Table 4.4).

Elderly people need the utmost care and support from the healthcare system as even minor injury might affect their health severely. The existing healthcare system needs the added power of IoT which can help elderly people to age in place by maintaining their independence with minimal involvement of care providers by offering its services through remote monitoring. Aging comes along with a lot of medications. In old age, people need to maintain their regular periodic medicine intake, which if not done correctly may cause some secondary complications [81]. To avoid these complications, Zanjal and Talmale [82] proposed an IoT-based medicine reminder system that reminds patients to take medicines on time and helps doctors to monitor medicine intake of the patients remotely. Another interesting application for elderly people is ambient assisted living (AAL). It helps seniors to perform their daily chores independently and yet lead a safe lifestyle. A. Dohr et al. [83] realized AAL through Keep In Touch (KIT) and closed-loop healthcare. KIT collates the patient health data which is then forwarded to the service center. Patients, service centers, and physicians form a closed loop. Physicians access the patients' data through the service center and give suggestions or make changes in the treatment if needed.

Table 4.4 Contributions in different operational areas in healthcare

Study	Sensors	Approach used
Intelligent system for prevention of Zika virus outbreak [63]	Mosquito sensors	Remote monitoring of infected users and Google map-based risk management
Cloud framework to control Ebola virus spread [64]	WBAN sensors	Infection level evaluation using J48 decision tree, proximal infected candidate identification by RFID, and monitoring of current state of spread using temporal network analysis
Epidemic control using smartphone-based body area network [65]	Body sensors	Epidemic control using social contact information and graph partitioning for division of large population
Monitoring and controlling pandemic influenza spread [67]	Not provided	Identification and categorization of infected patients in five groups based on symptoms and calculation of infection spread by patient using Outbreak Role Index
Mobile platform for diabetes management [71]	Multiple medical sensors including blood glucose monitoring sensor, blood pressure monitoring sensor, pulse rate monitor, weight scale	Real-time interaction with physician and personalized feedback to patient
Smart incubator for premature neonate monitoring [73]	LM35 temperature sensor	Early identification of critical events such as alert generation if incubator temperature rises above normal range
Asthma attack prediction system [77]	Fitbit, PS2 pollen sensor, humidity sensor, food allergen detector	Patient activity tracking using GPS and smartphone-based alert generation for abnormal situations
Mobile healthcare system for wheelchair users [80]	Wireless heart rate sensor, ECG sensor, pressure-detecting cushion, accelerometer sensor	Remote monitoring of wheelchair-bound users and fall detection using pressure and accelerometer sensor

Alzheimer's is another most common medical condition diagnosed in elderly individuals; it worsens memory and hinders brain actions. It starts with trouble in remembering newly learned information and leads to forgetting close members of the family. Muhammad Wasim Raad et al. [84] proposed ECG sensor-based model to monitor patients with Alzheimer's staying at home. At-home care products developed specifically for the elderly are available online; a few of them include wireless bed alarms for preventing falls, cordless motion sensor and nurse call system for detecting the patient's motion, and wireless bed pad and pager for detecting the motion of aged people [85–87].

4.3 Physical Abnormality Monitoring

IoT is unveiling various opportunities for people with physical abnormalities, overwhelming them with advanced and appealing prospects and helping them to create a better life for themselves. Crosswalk, an app created to help pedestrians with difficulty walking, works by signaling the traffic lights to extend the time for safe crossover [88]. Lin Yang et al. [80] proposed a mobile-based healthcare system for wheelchair users staying at home. The system monitors heart rate, records ECG, detects fall using a pressure cushion, and monitors surroundings for detecting emergency situations. However, this system is restricted only within the home area and does not consider the person's mobility out of the home environment. Extension of IoT systems from fixed to mobile systems particularly for the disabled remains a challenge. Laura Lenz et al. [89] discussed the use of IoT and big data as impactful technologies that can help learning-disabled students and also stated how monitoring can help therapists to better understand the behavior of learning-disabled students in the long run. Mari Carmen Domingo [90] proposed an IoT architecture for different environments of home, school, and shopping areas for impaired people and analyzed how people with vision, listening, and physical disabilities can benefit from the power of IoT. The problems such as customization of the IoT system for the disabled, encouraging disabled people to grow social contacts, self-management, standardization, scalability, and cooperation among connected things, remain to be solved as challenges for future research.

4.4 Profile-Based Monitoring

Various life and health-challenging professions can benefit from IoT which can help to automate certain tasks and send appropriate signals for informed and timely decision-making. For example, IoT can be of help in monitoring and maintaining the health of professionals in various profiles including athletics, diving, defense, mining, trekking, and so on.

Athletes On the one side, IoT helps athletes in monitoring their nutrition intake, exercise levels, and sleep quality; on the other side, it also helps coaches to track the effort and recovery level of the athletes. It alerts coaches in real time to the athletes' improper activities or internal injuries, allowing the sportsmen to avert major future issues. Castillejo created a sports GUI program that looks after the athlete's health by recommending various activities. This software has been thoroughly tested on real devices in the gymnasium [91].

Divers Divers risk their lives to save the lives of others. If they have a problem, the Internet of Things (IoT) may be able to help them. Sensor-enabled IoT devices can track their travels and alert them if they are in danger. Sharknet and Aeris have created a device that can track divers' movements and send out alarms by sending divers' position and depth if they get into trouble [92].

Defense IoT can be a useful tool for providing healthcare to soldiers stationed in remote regions. Military bases have a well-defined perimeter within which military health services are provided, both in terms of routine checkups and on demand. The wearables can alert soldiers and, if necessary, notify a medical staff at the military base hospital to provide timely medical assistance. A framework designed for use on the battlefield allows tight integration of military personnel, weapons, and complete warfare [93].

Use of IoT in these fields is in its early years, and a number of opportunities exist for improving the healthcare services for a wide range of domains in various dimensions, including age, profession, and demography.

5 Conclusion

Timely identification of an ailment or medical complication can fasten the recovery process of the patients. IoT has the power to administer this urgency for providing affordable and accessible care by notably revamping the traditional health-monitoring system. This chapter presents a systematic study of IoT in the healthcare domain. It discusses the IoT architecture including the various sensors and technologies related to data communication, storage, analytics, and visualization that have enabled an unbounded expansion of IoT-based healthcare services. This work describes different operational areas of healthcare including disease monitoring, age-based monitoring, physical abnormality monitoring, and work profile-based monitoring. The chapter also discusses the related work and research gaps in each of these areas, thus providing a comprehensive survey to till-date research in the field which is expected to be useful for researchers who are new to the field, medical professionals, as well as engineers.

References

1. Evans, D. (2011). The internet of things: How the next evolution of the internet is changing everything. *CISCO White Paper, 1*(2011), 1–11.
2. Mehta, R., Sahni, J., & Khanna, K. (2018). Internet of things: Vision, applications and challenges. *Procedia Computer Science, 132*, 1263–1269.
3. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems, 29*(7), 1645–1660.
4. Bhatt, C., Dey, N., & Ashour, A. S. (Eds.). (2017). *Internet of things and big data technologies for next generation healthcare*. Springer.

5. Gope, P., & Hwang, T. (2015). BSN-care: A secure IoT-based modern healthcare system using body sensor network. *IEEE Sensors Journal, 16*(5), 1368–1376.
6. Yeh, K. H. (2016). A secure IoT-based healthcare system with body sensor networks. *IEEE Access, 4*, 10288–10299.
7. Wu, T., Wu, F., Redoute, J. M., & Yuce, M. R. (2017). An autonomous wireless body area network implementation towards IoT connected healthcare applications. *IEEE Access, 5*, 11413–11422.
8. Ray, P. P. (2014). Internet of things based physical activity monitoring (PAMIoT): An architectural framework to monitor human physical activity. *Proceeding of IEEE CALCON, Kolkata*, pp. 32–34.
9. Ray, P. P. (2014, November). Home health hub internet of things (H 3 IoT): An architectural framework for monitoring health of elderly people. In *2014 international conference on science engineering and management research (ICSEMR)* (pp. 1–3). IEEE.
10. Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer Networks, 54*(15), 2787–2805.
11. Gomez, C., Oller, J., & Paradells, J. (2012). Overview and evaluation of Bluetooth low energy: An emerging low-power wireless technology. *Sensors, 12*(9), 11734–11753.
12. *Top 7 IoT communication protocols.* Retrieved from https://breadware.com/blog/iot-communication-protocols/. Last Accessed on 30 May 2021.
13. *Top 15 standard IoT protocols that you must know about.* Retrieved from https://www.ubuntupit.com/top-15-standard-iot-protocols-that-you-must-know-about/. Last Accessed on 30 May 2021.
14. Tabish, R., Mnaouer, A. B., Touati, F., & Ghaleb, A. M. (2013, November). A comparative analysis of BLE and 6LoWPAN for U-HealthCare applications. In *2013 7th IEEE GCC conference and exhibition (GCC)* (pp. 286–291). IEEE.
15. *Mobile IoT (M-IoT) applications for the NB-IoT and LTE-M evolution of beacons*. Retrieved from: http://bewhere.com/nb-iot-and-lte-m/. Last Accessed on 30 May 2021.
16. *WiFi's evolving role in IoT.* Retrieved from: https://www.networkworld.com/article/3196191/wifi-s-evolving-role-in-iot.html. Last Accessed on 30 May 2021.
17. Wi-Fi HaLow. Retrieved from: https://www.wi-fi.org/discover-wi-fi/wi-fi-halow. Last Accessed on 30 May 2021.
18. Sun, W., Choi, M., & Choi, S. (2013). IEEE 802.11 ah: A long range 802.11 WLAN at sub 1 GHz. *Journal of ICT Standardization, 1*(1), 83–108. Wi-Fi HaLow. Retrieved from: http://www.methods2business.com/wifi-halow-technology. Last Accessed on 30 May 2021
19. *D-Link, Asus tout 802.11ax Wi-Fi routers, but you'll have to wait until later in 2018.* Retrieved from: https://www.zdnet.com/article/d-link-asus-tout-802-11ax-wi-fi-routers-but-youll-have-to-wait-until-later-in-2018/. Last Accessed on 30 May 2021.
20. *Wi-Fi alliance introduces Wi-Fi 6.* Retrieved from: https://www.wi-fi.org/news-events/newsroom/wi-fi-alliance-introduces-wi-fi-6. Last Accessed on 30 May 2021.
21. *A look at IEEE 802.11ax-2019, the new Wi-Fi standard for HEW (high-efficiency Wi-Fi).* Retrieved from: https://www.allaboutcircuits.com/news/IEEE-802.11ax-2019-new-Wi-Fi-standard-hew-high-efficiency-WiFi/. Last Accessed on 30 May 2021.
22. Olsson, J. (2014). 6LoWPAN demystified. *Texas Instruments, 13*, 1–13.
23. Bhagwat, P. (2001). Bluetooth: Technology for short-range wireless apps. *IEEE Internet Computing, 5*(3), 96–103.
24. *Next-generation 802.11ax Wi-Fi: Dense, fast, delayed*. Retrieved from: https://www.zdnet.com/article/next-generation-802-11ax-wi-fi-dense-fast-delayed/. Last Accessed on 30 May 2021.
25. Park, C., & Rappaport, T. S. (2007). Short-range wireless communications for next-generation networks: UWB, 60 GHz millimeter-wave WPAN, and ZigBee. *IEEE Wireless Communications, 14*(4), 70–78.
26. Singh, C. K., Kumar, A., & Ameer, P. M. (2008). Performance evaluation of an IEEE 802.15.4 sensor network with a star topology. *Wireless Networks, 14*(4), 543–568.

27. Kushalnagar, N., Montenegro, G., & Schumacher, C. (2007). IPv6 over low-power wireless personal area networks (6LoWPANs): Overview, assumptions, problem statement, and goals. *RFC, 4919*, 1–11.
28. ZigBee Alliance. Retrieved from: https://www.zigbee.org/zigbee-for-developers. Last Accessed on 30 May 2021.
29. Elahi, A., & Gschwender, A. (2009). *ZigBee wireless sensor and control network*. Pearson Education.
30. *Dotdot.* Retrieved from: https://www.zigbee.org/zigbee-for-developers/dotdot/. Last Accessed on 30 May 2021.
31. *LoRa mesh networking with simple Arduino-based modules.* Retrieved from: https://nootropicdesign.com/projectlab/2018/10/20/lora-mesh-networking/. Last Accessed on 30 May 2021.
32. *The power of LoRaWAN for sensor networks.* Retrieved from: https://www.technolution.eu/en/about-us/publications/430-the-power-of-lorawan-for-sensor-networks.html. Last Accessed on 30 May 2021.
33. *Differences between Nb-IoT and LTE-M.* Retrieved from: https://accent-systems.com/blog/differences-nb-iot-lte-m/. Last Accessed on 30 May 2021.
34. *Cellular IoT explained – NB-IoT vs. LTE-M vs. 5G and more.* Retrieved from: https://www.leverege.com/blogpost/cellular-iot-explained-nb-iot-vs-lte-m. Last Accessed on 30 May 2021.
35. *Arduino.* Retrieved from: https://www.arduino.cc/. Last Accessed on 30 May 2021.
36. Raspberry Pi. Retrieved from: https://www.raspberrypi.org/. Last Accessed on 30 May 2021.
37. Yang, G., Xie, L., Mantysalo, M., Zhou, X., Pang, Z., Da Xu, L., Kao-Walter, S., Chen, Q., & Zheng, L. R. (2014). A health-IoT platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE Transactions on Industrial Informatics, 10*(4), 2180–2191.
38. Lohani, B., Indic, P., & Shirvaikar, M. (2018, May). Extraction of vital signs using real time video analysis for neonatal monitoring. In *Real-time image and video processing 2018* (Vol. 10670, p. 1067005). International Society for Optics and Photonics.
39. Mahfouz, M. R., To, G., Gaylord, M. S., & Lorch, V. (2015). Neonatal health care monitoring system. U.S. Patent Application No. 14/526,003.
40. Ransing, R. S., & Rajput, M. (2015, January). Smart home for elderly care, based on wireless sensor network. In *2015* IEEE international conference on nascent technologies in the engineering field (ICNTE) (pp. 1-5). IEEE.
41. Keim, D. A. (2002). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics, 8*(1), 1–8.
42. *Cloud computing vs Fog computing.* Retrieved from: https://www.educba.com/cloud-computing-vs-fog-computing/. Last Accessed on 30 May 2021.
43. *Fog computing vs cloud computing business growth and disruption.* Retrieved from: https://blog.tyronesystems.com/fog-computing-vs-cloud-computing-business-growth-and-disruption. Last Accessed on 30 May 2021.
44. *Fog computing vs Cloud computing for IoT projects.* Retrieved from: https://www.sam--solutions.com/blog/fog-computing-vs-cloud-computing-for-iot-projects/. Last Accessed on 30 May 2021.
45. Saharan, K. P., & Kumar, A. (2015). Fog in comparison to cloud: A survey. *International Journal of Computer Applications, 122*(3), 10–12.
46. Poungponsri, S., & Yu, X. H. (2013). An adaptive filtering approach for electrocardiogram (ECG) signal noise reduction using neural networks. *Neurocomputing, 117*, 206–213.
47. Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., & Kwak, K. S. (2015). The internet of things for health care: A comprehensive survey. *IEEE Access, 3*, 678–708.
48. Azimi, I., Rahmani, A. M., Liljeberg, P., & Tenhunen, H. (2017). Internet of things for remote elderly monitoring: A study from user-centered perspective. *Journal of Ambient Intelligence and Humanized Computing, 8*(2), 273–289.

49. Venkatesan, C., Karthigaikumar, P., Paul, A., Satheeskumaran, S., & Kumar, R. (2018). ECG signal preprocessing and SVM classifier-based abnormality detection in remote healthcare applications. *IEEE Access, 6*, 9767–9773.
50. Forkan, A. R. M., Khalil, I., Tari, Z., Foufou, S., & Bouras, A. (2015). A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living. *Pattern Recognition, 48*(3), 628–641.
51. Khan, F. A., Haldar, N. A. H., Ali, A., Iftikhar, M., Zia, T. A., & Zomaya, A. Y. (2017). A continuous change detection mechanism to identify anomalies in ECG signals for WBAN-based healthcare environments. *IEEE Access, 5*, 13531–13544.
52. Salem, O., Guerassimov, A., Mehaoua, A., Marcus, A., & Furht, B. (2013, June). Sensor fault and patient anomaly detection and classification in medical wireless sensor networks. In *2013 IEEE international conference on communications (ICC)* (pp. 4373–4378). IEEE.
53. Bedi, P., Kaur, H., & Marwaha, S. (2007, January). Trust based recommender system for semantic web. *IJCAI, 7*, 2677–2682.
54. Davis, D. A., Chawla, N. V., Christakis, N. A., & Barabasi, A. L. (2010). Time to CARE: A collaborative engine for practical disease prediction. *Data Mining and Knowledge Discovery, 20*(3), 388–415.
55. Toshniwal, D. (2013, February). Clustering techniques for streaming data-a survey. In *2013 IEEE 3rd international advance computing conference (IACC)* (pp. 951–956). IEEE.
56. Archenaa, J., & Anita, E. M. (2017). Health recommender system using big data analytics. *Journal of Management Science and Business Intelligence, 2*(2), 17–23.
57. Le, X. H., Khalid, M., Sankar, R., & Lee, S. (2011). An efficient mutual authentication and access control scheme for wireless sensor networks in healthcare. *Journal of Networks, 6*(3), 355.
58. Malasri, K., & Wang, L. (2009). Design and implementation of a secure wireless mote-based medical sensor network. *Sensors, 9*(8), 6273–6297.
59. Moosavi, S. R., Gia, T. N., Nigussie, E., Rahmani, A. M., Virtanen, S., Tenhunen, H., & Isoaho, J. (2016). End-to-end security scheme for mobility enabled healthcare internet of things. *Future Generation Computer Systems, 64*, 108–124.
60. *Security and privacy concerns for IoT.* Retrieved from: https://www.iotworldtoday.com/2016/04/20/top-10-reasons-people-aren-t-embracing-iot/. Last Accessed on 30 May 2021.
61. Roberts, J. C. (2007, July). State of the art: Coordinated & multiple views in exploratory visualization. In *Fifth international conference on coordinated and multiple views in exploratory visualization (CMV 2007)* (pp. 61–71). IEEE.
62. Park, S., Linsen, L., Kreylos, O., Owens, J. D., & Hamann, B. (2005, November). A framework for real-time volume visualization of streaming scattered data. In *Proceedings of tenth international fall workshop on vision, modeling, and visualization* (pp. 225–232). IEEE.
63. Sareen, S., Gupta, S. K., & Sood, S. K. (2017). An intelligent and secure system for predicting and preventing Zika virus outbreak using Fog computing. *Enterprise Information Systems, 11*(9), 1436–1456.
64. Sareen, S., Sood, S. K., & Gupta, S. K. (2018). IoT-based cloud framework to control Ebola virus outbreak. *Journal of Ambient Intelligence and Humanized Computing, 9*(3), 459–476.
65. Zhang, Z., Wang, H., Wang, C., & Fang, H. (2014). Cluster-based epidemic control through smartphone-based body area networks. *IEEE Transactions on Parallel and Distributed Systems, 26*(3), 681–690.
66. Sood, S. K., & Mahajan, I. (2017). Wearable IoT sensor based healthcare system for identifying and controlling chikungunya virus. *Computers in Industry, 91*, 33–44.
67. Sandhu, R., Gill, H. K., & Sood, S. K. (2016). Smart monitoring and controlling of Pandemic Influenza A (H1N1) using social network analysis and cloud computing. *Journal of Computational Science, 12*, 11–22.
68. Hassan, N. H., Salwana, E., Drus, S. M., Maarop, N., Samy, G. N., & Ahmad, N. A. (2018). Proposed conceptual Iot-based patient monitoring sensor for predicting and controlling dengue. *International Journal of Grid and Distributed Computing, 11*(4), 127–134.

69. Kapoor, R., Sidhu, J. S., & Chander, S. (2018, October). IoT based national healthcare framework for vector-borne diseases in India perspective: A feasibility study. In *2018 international conference on advances in computing, communication control and networking (ICACCCN)* (pp. 228–235). IEEE.
70. Raji, A., Jeyasheeli, P. G., & Jenitha, T. (2016, January). IoT based classification of vital signs data for chronic disease monitoring. In *2016 IEEE 10th international conference on intelligent systems and control (ISCO)* (pp. 1–5). IEEE.
71. Al-Taee, M. A., Al-Nuaimy, W., Al-Ataby, A., Muhsin, Z. J., & Abood, S. N. (2015, November). Mobile health platform for diabetes management based on the internet-of-things. In *2015 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT)* (pp. 1–5). IEEE.
72. *13 smart products for the internet of toddlers.* Retrieved from: https://community.arm.com/iot/b/blog/posts/13-smart-products-for-the-internet-of-toddlers. Last Accessed on 30 May 2021.
73. Ashish, B. (2017, February). Temperature monitored IoT based smart incubator. In *2017 IEEE international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC)* (pp. 497–501). IEEE.
74. *Top 5 lifestyle diseases among children in India.* Retrieved from: https://www.parentcircle.com/article/top-5-lifestyle-diseases-among-children-in-india/. Last Accessed on 30 May 2021.
75. *Pediatric Infectious diseases in India.* Retrieved from: https://internationalreportingproject.org/stories/view/pediatric-infectious-diseases-in-india. Last Accessed on 30 May 2021.
76. Yang, Y., Lee, T., Lee, Y., Choi, J., Park, E., & Lim, H. (2017, June). Implementation of infants risk detection sensing system using IoT. *AIP Conference Proceedings, 1836*(1), 020073. AIP Publishing.
77. Siddiquee, J., Roy, A., Datta, A., Sarkar, P., Saha, S., & Biswas, S. S. (2016, October). Smart asthma attack prediction system using internet of things. In *2016 IEEE 7th annual information technology, electronics and mobile communication conference (IEMCON)* (pp. 1–4). IEEE.
78. Hsu, S. J., Lin, S. S., Pai, T. W., & Fujita, H. (2016, October). Proactive healthcare and an early warning mechanism for coronary artery disease patients using internet-of-thing devices. In *2016 IEEE international conference on systems, man, and cybernetics (SMC)* (pp. 001400–001405). IEEE.
79. Chatterjee, P., Armentano, R. L., & Cymberknop, L. J. (2017, December). Internet of things and decision support system for eHealth-applied to cardiometabolic diseases. In *2017 IEEE international conference on machine learning and data science (MLDS)* (pp. 75–79). IEEE.
80. Yang, L., Ge, Y., Li, W., Rao, W., & Shen, W. (2014, May). A home mobile healthcare system for wheelchair users. In *Proceedings of the 2014 IEEE 18th international conference on computer supported cooperative work in design (CSCWD)* (pp. 609–614). IEEE.
81. *IoT for health management.* Retrieved from: https://healthcare.cioreview.com/cxoinsight/three-ways-iot-is-advancing-population-health-management-nid-25683-cid-31.html. Last Accessed on 30 May 2021.
82. Zanjal, S. V., & Talmale, G. R. (2016). Medicine reminder and monitoring system for secure health using IOT. *Procedia Computer Science, 78*, 471–476.
83. Dohr, A., Modre-Opsrian, R., Drobics, M., Hayn, D., & Schreier, G. (2010, April). The internet of things for ambient assisted living. In *2010 IEEE seventh international conference on information technology: New generations* (pp. 804–809). IEEE.
84. Raad, M. W., Sheltami, T., & Shakshuki, E. (2015). Ubiquitous tele-health system for elderly patients with Alzheimer's. *Procedia Computer Science, 52*, 685–689.
85. *Graying with grace.* Retrieved from: https://www.grayingwithgrace.com/best-wireless-bed-alarm-elderly/. Last Accessed on 30 May 2021.
86. *Cordless motion sensor alarm with nurse call button & AC adapter*. Retrieved from: https://www.nursinghomeaids.com/cordless-motion-sensor-alarm-with-nurse-call-button-ac-adapter/. Last Accessed on 30 May 2021.

87. *Wireless pager & bed exit alarm*. Retrieved from: https://val-u-care.com/wireless-bed-alarm-bed-sensor-pad-cd-pager.html. Last Accessed on 30 May 2021.
88. *IoT for disabled.* Retrieved from: https://blog.gemalto.com/iot/2017/07/27/iot-helping-people-living-disability/. Last Accessed on 30 May 2021.
89. Lenz, L., Pomp, A., Meisen, T., & Jeschke, S. (2016, March). How will the internet of things and Big Data analytics impact the education of learning-disabled students? A concept paper. In *2016 IEEE 3rd MEC international conference on Big Data and smart city (ICBDSC)* (pp. 1–7). IEEE.
90. Domingo, M. C. (2012). An overview of the internet of things for people with disabilities. *Journal of Network and Computer Applications, 35*(2), 584–596.
91. Castillejo, P., Martinez, J. F., Rodriguez-Molina, J., & Cuerva, A. (2013). Integration of wearable devices in a wireless sensor network for an E-health application. *IEEE Wireless Communications, 20*(4), 38–49.
92. *IoT for divers.* Retrieved from: https://www.rtinsights.com/sharknet-and-aeris-jump-in-on-iot-tracking-for-divers/. Last Accessed on 30 May 2021.
93. Ray, P. P. (2015, December). Towards an internet of things based architectural framework for defence. In *2015 IEEE international conference on control, instrumentation, communication and computational technologies (ICCICCT)* (pp. 411–416). IEEE.

Chapter 5
Industrial IoT and Its Applications

Jyotsana Grover

Abstract This chapter focuses on IIoT (Industrial Internet of Things) and its different applications such as the manufacturing industry, healthcare, and food industry. In all of these different application domains, the core technology and the core technological ideas remain the same; the only thing that changes is the type of sensors that would be used. Traditional manufacturing posed different challenges like unavailability of real-time data, unbalanced workload, and longer changeover time. Smart factory tries to overcome these challenges by integrating IoT and operational technology (OT). The chapter also presents how IIoT can help in transforming present-day healthcare and making healthcare much more affordable, much more efficient, and much more autonomous. IIoT solutions can be used to alleviate some of the problems that are encountered by people with respect to health. There are different sensors like ECG sensor, blood pressure sensor, glucose-monitoring sensor, and temperature sensor that can be procured by patients themselves for monitoring their health conditions at their homes. Further, these systems can be internetworked so that if any patient has a critical condition, different levels of alerts would be sent to the hospital to which this patient is registered. The chapter also discusses IIoT implementation in the food industry. The process involves sowing seeds, growing crops, applying fertilizers, applying pesticides, maturity of crops, harvesting crops, food grain processing, packaging of food grains, and transporting to a wholesale market and finally to the retail market. This is called the supply chain from field to plate. IIoT devices can be used in the agricultural field for monitoring the sowing of seeds, for growth of crops, for applying fertilizers, and for irrigation. These devices can also be used at each step. We describe each of these applications in detail in this chapter.

Keywords IoT · IIoT · Cloud computing · Big data analytics · Smart factory

J. Grover (✉)
Department of Computer Science and Information System, BITS Pilani, Pilani, India
e-mail: jyotsana.grover@pilani.bits-pilani.ac.in

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_5

1 Introduction

Different IoT solutions [1] are being implemented in the industry for solving different industrial problems to make industrial processes and manufacturing processes much more efficient than the way it is at present. The main aim of IoT is to interconnect different smart objects so that they are uniquely identified and able to interoperate between themselves.

In these smart objects, intelligence is embedded. IIoT is an application of IoT in the industries to modify various existing industrial systems. IIoT links the automation system with enterprise, planning, and product life cycle.

IIoT borrows some of the features of IoT and some from the vision of Industry 4.0 and tries to have a separate vision and technology for itself. So Industry 4.0 gives a framework for automation and data exchange in manufacturing technologies. So, Industry 4.0 basically tries to improve the automation and data exchange in manufacturing and technologies; it tries to incorporate concepts from cyber physical systems, IoT cloud computing, and so on. IIoT and IoT are not the same [2] as IoT focuses on consumer-level services and consumer-level products whereas IIoT has the focus on the enterprise. IoT traditionally focuses on the convenience of the individuals, and IIoT focuses on the efficiency, safety, and security of operation. In terms of machine-to-machine (M-2-M) communication, its use is limited in IoT, whereas M-2-M communication is extensively used in IIoT; the whole industrial operation in a plant is automated. So, one machine talking to another machine is quite extensive in the industrial sector IIoT.

Some of the integral components for building IIoT are machine learning, big data technology, M-2-M communication, and automation. So machine learning involves learning from the existing data and then trying to make predictions so as to make things better in the future. The machine-to-machine communication is about two machines communicating with each other for getting a particular work or a task accomplished without any human intervention. So, a robotic arm opening the door of a refrigerator and then performing certain other tasks in the refrigerator are an example of machine to machine. So, maybe the robotic arm goes and opens the door of the refrigerator and checks whether there is sufficient milk in the milk pot of the refrigerator or not; if there is no sufficient milk, then the system as a whole will send an SMS to the milk person. So, we have machine-to-machine communication and automation without human intervention. So, these are the different aspects of IIoT. So, IIoT is supported by a huge amount of data collected from sensors, and it is based on wrap-and-reuse approach rather than rip-and-replace approach. We are not building a new system from scratch; we use the existing industrial systems; wrap them with sensors, actuators, and so on; and make things efficient. In IIoT, we try to reengineer the existing systems and the processes and we do not build anything brand new from scratch.

IIoT combines the second generation of Internet (where things are connected), fourth generation of industrial automation, and cloud computing. So, cloud has become a very popular technology since about more than half a decade and it is being used in the industrial sector as well. So, cloud offers computational environments, computational infrastructure, computational platforms, and computational software in addition to regular storage. Using cloud one can get access to everything in an industry without basically having to purchase these of their own.

In the IIoT network, we have physical objects, different systems and subsystems, platforms, and applications that are interconnected. IIoT networks can communicate with one another, external environment, and different people. The acquisition of IIoT has led to the availability and affordability of sensors, processors, and other technologies, which facilitate the capture of and access to real-time information.

IIoT has a plethora of applications like transportation [3–4], power plant security and safety [5], inventory management and quality control [6], facility management [7], milk processing and packaging industries, healthcare, manufacturing industries, and food industry. But in this chapter, we only focus on the three applications: healthcare, manufacturing industries, and food industry.

2 Requirements of IIoT

There are four broad requirements of IIoT. We need the hardware and software connectivity and a cloud platform with respect to processing, infrastructure, and data storage. Application development and big data analytics are very important. All these different sensors and actuators are fitted to these different machines and manufacturing equipment. They throw in a huge volume of data in real time that can reveal a lot of information. Data also have other characteristics, such as they come in huge velocities, and there are different types of such data, like speech, images, and video, all coming at the same time need to be handled, and so on. By mining such data using big data analytics, one can predict different things to make these industrial processes much more efficient.

Another IIoT requirement is with respect to access, i.e., one can access anything, at anytime, from anywhere in IIoT. So here end-to-end security is important.

User experience has to be taken into account as one of the fundamental requirements for building IIoT, i.e., what users exactly want, how their problems can be addressed and solved, how through the use of IIoT system users can do things better and how their experience as a whole can be improved.

Transition to smart machines: By adding sensors and actuators we are making the machine smart. Asset management is very important; through asset management, different sensors and actuators (which are the industrial assets) can be managed in a much more efficient way.

We now take a deep dive into the issues and challenges [8] in the adoption of IIoT:

- There are data integration challenges, so here we refer to a large number of machines. These different machines have their own heterogeneity in place.

Integration of these heterogeneous data coming in high volumes and velocities is a herculean task.

- Cyber security is very important because we are referring to a connected world, where there is connectivity between (i) machines themselves, (ii) machine and human beings, and (iii) machines and human beings. Due to this connectivity, it is quite likely that some vulnerabilities are introduced in the overall network, making it possible for different types of attacks and newer types of attacks to be launched.
- There is lack of standardization; large automation supplier firms do not encourage open standardization as it will reduce customers' reliance on them. So, small automation supplier firms basically lack the capability to incentivize this huge step. Lack of standardization leads to different issues related to device interoperability and semantic interoperability. So, different machines supporting different IoT devices need to talk to each other and they have been made by different vendors.
- Legacy installations are a very crucial aspect to be considered. Technology is evolving fast. Industries have a huge legacy machine base and these machines have been operating successfully since decades. With the incorporation of IIoT, we need to have support for these newer technologies and newer machines being procured. So fast-evolving technology should coexist with the legacy equipment without leaving any kind of vulnerability in the whole integrated system.
- The next issue with IIoT adoption is lack of skilled workers. The workers do not have IIoT-related skills, like data integration. The technologies associated with IIoT are new in nature, so workers should have fast and diverse knowledge about these technologies.

Now we discuss the different design considerations for building IIoT.

3 Design Considerations for IIoT

To use an IoT device for industrial applications, the following design objectives [9] have to be considered:

- Energy: Time for which the IoT device can operate with the limited power supply
- Latency: Time required to transmit the data
- Throughput: Maximum data transmitted across the network
- Scalability: Number of devices supported
- Topology: Communication among the devices, i.e., interoperability
- Safety and security: Degree of safety and security of the application

So, now we discuss some of the application areas of IIoT like healthcare industry, manufacturing industry, and food industry.

4 IIoT Applications: Healthcare

4.1 IIoT in Healthcare

Earlier, people used to die due to lack of healthcare. People used to forget about their health due to busy life, and this has also happened in the recent past. Additionally, the number of diseases has also increased in recent times. So, we have our IIoT solutions that could be used to alleviate some of the problems that are encountered by people with respect to health. So, IIoT solutions can help in making healthcare easier and affordable. There are different sensors [10] such as the ECG sensor, blood pressure sensor, glucose-monitoring sensor, and temperature sensor that are currently available in the market; these are affordable and can be procured by patients themselves for monitoring their health conditions at their homes or these could also be purchased by different healthcare facilities, hospitals, and so on. If any patient has a critical condition, different levels of alerts would be sent to the healthcare facilities or hospitals to which these patients are registered.

4.1.1 Concerns in Healthcare Regarding IIoT

There are different healthcare concerns with respect to their implementation of IIoT. Populations are aging all over the world, different diseases are increasing, and medical expenses are also increasing. These are some of the generic healthcare challenges, but from an IIoT perspective as well. Catering to these requirements is also a challenge; scalability of IIoT solutions will have to be taken into consideration not only in terms of numbers, but also in terms of diversity. So, all generic healthcare challenges also have implications on IIoT implementation in the healthcare industry.

4.1.2 IIoT Implementation for Old Population

Looking at the age groups, populations are growing older; between 2017 and 2050, the persons aged over 60 years are expected to increase more than double. By 2050, the number of elderly persons is expected to grow to 2.1 billion worldwide which is a huge number. We have a growing population and it will invite taking care of their health. Monitoring of elderly people’s health condition has to be done efficiently. So, telecare applications, smartphone, or telemedicine basically can help elderly people to live safely. So, we can have telemedicine solutions [11] being deployed in the homes of these elderly persons so that the doctors can remotely monitor the condition of these elderly people from their homes. This applies to not only elderly people but also other population.

4.1.3 IIoT Implementation for Increase in the Diseases

Not only the number of diseases but also the types of diseases are increasing day by day. So, we need to have a suitable and efficient large-scale monitoring system that will cater to the particular problem and address that problem. So, nowadays, we talk about having telecare applications, smartphone applications, and telemedicine applications for elderly people. We are also talking about these kinds of solutions for catering to the other segments of the population. So, continuous monitoring of patients' health can be done, and this can also help in reducing the number of cases of hospitalization. Sensors can collect blood pressure, respiration, pulse rate, heart rate data, and weight data continuously and as and when required; if any alarm has to be triggered, this can be done in a much more efficient manner, and this can be done if any abnormal solution is detected or any abnormality is going to arise in the future.

4.1.4 IIoT Reduces the Expenditure

IIoT-based solutions for healthcare can help in reducing the expenditure; different wearable healthcare devices can help in reducing the cost of health checkup; remote continuous monitoring of patients using smart sensors would be made possible. In hospitals and other healthcare units, smart beds can be deployed which can send notification to the doctor about the patient's activity.

4.1.5 Cloud-Enabled IIoT Healthcare Solution

There are different layers in the IIoT healthcare architecture. It starts from acquiring the data from the sensors; at the very bottom; sending the sensed data; processing the data; storing the data and getting different information knowledge etc. about what is going on, underneath from the data and trying to make more sense out of the data, through information processing, knowledge processing and so on.

Now we discuss the cloud-enabled IIoT healthcare architecture. On one end we are going to have all devices which are typically sensor enabled, i.e., the smart sensors and then some gateway. So, from these sensors, the data are going to be sent to the IoT-enabled cloud platform where different analytics will be performed. Not just analytics, but health data verification can be performed and the data can be stored and also processed in a computer or a computational resource in the cloud and so on. And there are lot of other different things that could also be done at the cloud and finally we are going to have this two-way communication and we are going to have healthcare applications which are going to be the beneficiaries from the cloud where analytics is performed. At the application end, different patient data about their health condition such as ECG, EMG, then may be pulse oximeter data, and many other different types of healthcare data could be made available after suitable processing and analytics. So, this is going to be the data flow architecture at very high level for healthcare IoT.

4.1.6 Benefits of IIoT in the Healthcare

Now we discuss the benefits of IIoT in healthcare. With IIoT, one can monitor the patient's health condition remotely. So, remote healthcare is possible. Remote real-time continuous monitoring of patients' health condition 24×7 is possible. Hospital staff can predict the arrival of a patient in their emergency units; it is also possible to have a hygiene monitoring system which can detect the cleanliness of the hospital and the healthcare facility. Medical staff can provide quality medical services with a small budget using IIoT. So, these are some of the benefits that IIoT implementation in healthcare can provide.

4.2 IIoT-Based Healthcare Devices

4.2.1 Wireless ECG Monitors

Wireless ECG monitors are there which can collect biosignals from ECG devices; the collected data could be sent to the cloud; and medical staff can analyze the health-related data in real time. In fact, we can have some programs which can autonomously analyze the data that are coming in and can send alerts. So, one example of a wireless IoT-enabled ECG sensor [12] is QardioCore. QardioCore is a wireless device for ECG monitoring.

4.2.2 Glucose Level Monitoring Device

IIoT is very useful in glucose level monitoring [13], particularly for diabetes patients as they need to check the glucose level quite often. Particularly, the ones who have higher degrees of diabetes are required to check the blood sugar quite often. So, if we have an automated IoT-enabled system to which the patients can be fitted, then automatically the data from these different patients can be made available to doctors who are treating the diabetes patients and so on; an example of continuous glucose monitoring device is the Dexcom. Dexcom devices can help in continuous glucose monitoring.

4.2.3 IIoT-Based Blood Pressure Monitor

Similarly, we have IIoT-based blood pressure monitors. Using IIoT devices the patient's blood pressure is measured in real time. Doctors can monitor the patient's blood pressure in real time; can get alerts if the blood pressure is beyond a particular threshold; and, depending on the blood pressure data, can prescribe medicines to the patients. One such example of blood pressure monitoring system is iHealth BP5.

4.2.4 IIoT-Based Body Temperature Sensor

In body temperature monitors, wearable sensors are there to continuously monitor the human body temperature. This sometimes is very much required for some patients who are suffering from diseases which make the patients vulnerable to a sudden increase or sudden decrease in the body temperature. So, there are different body temperature sensors in the market. One such body temperature sensor is by Kinsa. So, they have their smart thermometer which is an IoT-based body temperature monitoring device.

4.2.5 IIoT-Based Asthma Treatment

For oxygen saturation monitoring, particularly for asthma patients, this is very important; oxygen saturation can be monitored with the help of IoT devices such as pulse oximeter. So, pulse oximeter can help in measuring the oxygen saturation, so this pulse oximeter could be integrated with connectivity solutions such as Bluetooth, which can send continuously the data of the oxygen saturation level of the patient who is being monitored.

4.2.6 IoT-Based Contact Lenses

IoT-based contact lenses are also there in the market. There are different IoT-based contact lenses which also offer Wi-Fi connectivity with smartphones so that the condition of the patient, their eye condition, their sugar level, etc. could also be monitored.

4.2.7 Smart Inhalers

IoT-based asthma treatment solutions are already in the market. Smart inhalers are a very essential requirement of asthma patients, and hence they are being manufactured. ADAMM is an intelligent asthma monitoring device that has been developed. This particular device can keep track of the body temperature, coughing rate, heart rate, etc. which are preliminary symptoms of an asthma attack.

4.2.8 Smart Phone-Based Healthcare Solution

Different smartphone-based healthcare solutions are already available. Smartphone devices connected to electronic devices such as sensors can help in collecting the data of patients. Smartphone is used to monitor the health of users and detect diseases. Smartphone healthcare apps provide low-cost healthcare devices which are sort of diagnostic apps that help in detecting the health condition of patients; it can also help in medical communication between the patients and the hospitals and can also offer medical education in the form of tutorials to the patients.

4.2.9 Smart Phone App: Health Assistant

Health assistant is one such app which keeps track of the health condition of the patients. Google Fit is another solution which keeps track of different physical activities of the patient. ECG self-monitoring is another solution which serves as ECG device, based on the "ECG Self Check" software.

4.2.10 IIoT Healthcare Technology

Different solutions can be explored that are there in terms of healthcare and IoT implementations in healthcare. Cloud is enabled together with big data because most of this data generated from these healthcare sensors have the nature of big data. So, cloud enablement, big data analytics, etc. are very important in IoT implementation in healthcare.

4.3 IIoT Healthcare Requirement and Challenges

Security is of paramount importance in the healthcare sector. Privacy of individuals is very important because the data that is being carried forward from one device to another through a particular communication channel should not be hacked and unauthorized users should not be able to get access to the data. So, ensuring the confidentiality of the data, integrity of the data, authentication mechanisms and their implementation, and availability of the data are very important in terms of security requirements and their implementations in IIoT healthcare.

IIoT healthcare has different challenges. There are challenges with respect to limited computational capability, not being able to perform expensive operations. There are hassles with respect to having very less device memory, energy limitation, and also taking care of the mobility of these different devices because the patients themselves are mobile. So, consequently, these devices themselves are wearable devices, and the sensors themselves are also mobile. Taking care of the mobility of these different devices from a technical point of view is a challenge; both from a communication and algorithmic point of view there are different challenges. So, taking care of all of them is important for consideration.

5 IIoT in Manufacturing Industries

5.1 Smart Factory

Motivation for smart factory: Now we discuss the drawbacks of traditional manufacturing industries. The machines work in isolation; that is, these machines are not connected. And because of this reason, there is unbalanced workload in these different machines. There are other drawbacks like unavailability of real-time data, longer changeover time (converting a line or machine from running one product to another), and extending production time (lack of proper information and data of the production line). These drawbacks are removed in the smart factory with the implementation of IIoT.

Smart factory involves machinery and equipment which improve processes through self-optimization and automation. Benefits of smart factory are as follows: supply of real-time data, data analysis and quality control, reduced changeover time, reduced production time, and flexibility and easy management.

The manufacturing devices, equipment, workforce, supply chain, and work platform are internetworked and linked to accomplish smart production. The integration has to be done in order to reduce operational costs, improve the productivity of the worker, and reduce injuries at the workplace. The reason for using IIoT is to improve safety in the manufacturing plant. With the help of IIoT, resource optimization and waste reduction can also be achieved, which are also again very important aspects in industrial engineering.

Let us now look at how the smart factory is going to work. So, at the very bottom we have the sensor-enabled tools and workers. They throw in a lot of data which are going to pass through the gateway to the cloud where further data analysis, data visualization, etc. are going to be done and the results are going to be made available to the respective stakeholders based on their corresponding policies.

5.2 Features of Smart Factory

In a smart factory [14] we talk about connected devices which are going to send a lot of data in real time continuously. Optimized components and optimized data without any human intervention or with minimal human intervention are characteristics of a smart factory.

Smart factory is transparent in the sense that we get a lot of live data on the metrics that are implemented and those data can be used suitably at different levels of management for quicker decision, and hence, transparency is also promoted in a smart factory.

Proactive feature means that we know proactively we can predict the future outcomes and take preventive actions depending on the situation and what is going to happen in the future. So, based on the prediction of future outcomes one can take preventive actions proactively. So, all of these things are possible in a smart factory. Now we discuss smart factory applications.

5.3 *Smart Factory Applications*

5.3.1 Airbus: Factory of Future

Airbus is a German company which is a state-of-the-art implementation of IIoT. So, Airbus is a major pair player in the aviation sector. It is a European aircraft manufacturer and it applies a lot of IoT technologies in its production process. So, essentially what happens is that during manufacturing productions on the floor of the plants and also after the products are deployed in a real aircraft, a lot of data can be collected. Further lot of data can be collected from the flight recorders while the flights are in operation. So, collecting data on flights will help to improve the in-flight experience and the workers on the factory floor can use these IoT devices to know about how much the manufacturing has processed, what are the different gaps, etc. in the process. So, Airbus is the digital manufacturing initiative which is also known as the factory of the future.

So, in a factory of the future, we talk about different components, such as IoT sensors for supply-chain management. Now we discuss modular equipment; use of different robots, robotic arms, etc.; use of concepts of industrial augmented reality; use of computer vision; image processing and video processing in real time; and so on.

Now we delve into the implementation of factory of the future; Airbus now has mechanisms for digital tracking and monitoring. There are tools and machines, and sensors are integrated into them. Further wearable sensors like smart glasses can be used with maybe augmented reality support. So, Airbus is using all of these different things for its implementation.

3D real-time visualization of the production process is possible and all of these things are deployed on A330 and A350 models and their assembly lines which are there in the Toulouse manufacturing plant; they have also deployed this factory of the future for the A400M model and their assembly operations in the UK.

5.3.2 Amazon: Robotic Shelves

Amazon has robotic shelves, and as this name suggests, Amazon uses different types of robots that will carry these shelves and rearrange them. Amazon basically is an e-commerce company and these shelves and their rearrangement robotically are very important and that makes the processes much more autonomous and efficient. So, the good part of this thing is that because it is an autonomous robotic system, using this system, the robots can efficiently locate and search different items from different shelves. Thus, in 2014, the operating cost was cut down by 20% using these robotic shelves by Amazon.

5.3.3 Caterpillar: Augmented Reality App

Caterpillar has the augmented reality (AR) app which is integrated with IoT; Caterpillar is a heavy equipment maker, and it has come up with the augmented reality app that generates end-to-end view of the factory floor. So, the machine operators can detect the need for tool replacement whenever it is required after viewing the end-to-end view through that particular AR app. This app sends instructions for doing things like tool replacement, air filter change, and fuel monitoring.

Caterpillar has the IoT-driven ship maintenance that is done by their marine division. They use the shipboard sensors to perform predictive maintenance analytics. The sensors that are deployed can monitor generators, engines, GPS, air-conditioning systems, and fuel meters. The analysis of the sensed data provides useful insights with respect to the power usage of refrigerated containers, cost of hull cleaning, and optimized cleaning schedule and their data; these are all provided through the analysis of the data that are obtained through these different sensors that are deployed in the onboard devices of the ships.

So, preventive maintenance analytics talks about the use of all these machine learning techniques. Tools and techniques like Python and Weka could be used to come up with these different predictive analytics and so on. It is used to have easier fault correction, reduced downtime, and increased profitability, using predictive maintenance analytics, and this is what Caterpillar is doing.

6 IoT and the Food Industry

Before discussing implementation and application of IIoT in the food industry, we take a deep dive into the chain of activities to be followed in the food industry.

6.1 Field to Plate

We first discuss what happens in the food industry. It starts with the agricultural field where the crops are grown. Farmers sow seeds, grow crops, apply fertilizers, apply pesticides, etc. and then after the crops get matured, these get harvested.

Following harvesting, these food grains need to be processed and packaged. These packages are transported typically to a wholesale market. Then it goes to the retailer, and finally, the consumers buy, cook, and consume the agricultural produce. This is typically the chain from the agricultural field to the plate. So, this is typically the chain of activities that are followed; this is the supply chain because ultimately for each of the activities, the supply will have to be ensured through this entire cycle.

So, in this kind of scenario, for each activity, sensors can be used; IIoT devices can be used in the agricultural field for monitoring the growth of crops and the sowing of seeds and for applying fertilizers precisely adequately. So, sensors, actuators, and different agricultural robots can be used over here. So, not only over here, even in harvesting, food grain processing, packaging, transportation, and logistics these could be used. The sensors and actuators help the systems or the machines that are helping in the activities to be much more efficient and autonomous. Now we discuss how IIoT is implemented in the food industry [12].

6.2 Implementation of IIoT in the Food Industry

IIoT is implemented in the food industry using three layers: sensing layer, communication layer, and application layer. These layers need to be internetworked. In the sensing layer, we have network sensors for food quality monitoring along the supply chain. The food grains go through the warehouses and the network sensors also monitor the temperature in those warehouses and also other environmental conditions.

The communication layer deals with stakeholder access supply-chain data, communication between different stakeholders, and communication between different components of the supply chain.

The application layer has applications for farmers, retailers, government, analysts, consumers, and insurance companies.

We need sensors for doing many tasks like sensors for monitoring humidity, temperature, and composition of food products. The sensors throw a lot of data in real time which have to be analyzed in in order to make the most out of those data that have been retrieved. So, we need easier process control, increased food safety, etc. and it is also very important to have adequate end-to-end traceability. So, if we have the field to plate and the corresponding supply chain adequately implemented using suitable IoT solutions, then it would be possible for example to trace a rice packet back to the paddy field.

6.3 Impact of IoT on the Food Industry

We now discuss the impact of IIoT in the food industry. We have an efficient production line as IoT monitors the equipment performance and detects the anomaly in the production line. This efficient production line enables the generation of real-time solutions by predictive maintenance.

Using the temperature tracking sensors we can implement adequate, suitable, and efficient food safety measures. Also, we have Automatic Hazard Analysis and Critical Control Points (AHACCP) checklist, which ensures food safety.

IIoT ensures transparency of the supply chain as the real-time data about the products are available, and it is easier to find the inefficiencies and meet the food safety regulations. We have minimized wastage of food resources, and we can analyze in real time, for example, the information of food products and reduce food wastage. So, all of these things are possible if we have IIoT implementation in the food industry.

6.3.1 Utilizing the IoT in the Farms

On the farm we can have sensors to monitor weather, crop maturity, presence of insects, and conditions of the field with respect to the soil conditions, for example, how much soil moisture is there in the field, how much is the water level, how much is the fertilizer content of the field, and soil nutrient condition of the field. So, all of these things are possible with the help of IoT implementation in the food industry.

6.3.2 Utilizing the IoT in the Livestock Barns

In the livestock barns, sensors can help in monitoring the health parameters of different animals such as cows, buffaloes, and different other livestock, including sheep and goats. So, all of the life monitoring in real time is done using IoT-enabled devices. Automated feeding cycles can be set up with the help of IoT implementation, and diet control of these different livestock is possible with the help of IoT implementation. Automated temperature control in the brooding barns and hatchery is also possible with the help of suitable IoT implementation.

6.3.3 Utilizing the IoT for Equipment in the Food Industry

On the equipment level, IoT enables GPS tracking whenever these animals are moving around their exact location, and their position could be tracked. This is just an example; similarly, GPS could be used for tracking the mobility of different components in the food industry. Drone-assisted field monitoring applications in agriculture are quite common and are being implemented using IoT.

6.3.4 IoT for Maintenance in the Food Industry

In the food industry, IoT implementation can be done for maintenance. Sensors are embedded in different machines, such as farm machinery and tractors, to monitor their condition and performance and to detect whether any machine is going to go down in the future. The early detection of warning signs, smart maintenance of these machines, and extending the lifetime of these equipment are possible with respect to maintenance in the food industry through IoT implementation.

6.3.5 IoT to Improve Margins in the Food Industry

IoT implementation in the food industry can improve the margins through predictive analytics, spotting early warning signs, making well-informed decisions, and maximizing profits.

6.3.6 IoT for the Consumer

For the consumer, there are different initiatives; smart label is an initiative by the Grocery Manufacturers Association (GMA), which uses a QR code to provide product-related information to consumers. These consumers consequently can get information about the ingredient details of a particular food item, allergen exposure of that particular food item, nutrition, value, and other information.

6.3.7 IoT for the Product in the Food Industry

Consumers can scan the QR code to get details about the product; the product information includes nutrition, ingredients, allergens, third-party certification, social compliance programs, usage instructions, advisories, and also safe handling instructions.

6.3.8 IoT for the Food Processing Factory

In the factory, IoT implementation can help different machineries in the food processing industry and also aid different workers who are working in the food processing industry to remain connected autonomously. This connectivity can help in gaining insights to improve the quality of the food product, the quality of the food processes, and so on, and consequently, they can also help in the reduction of the time to market (TTM).

6.3.9 IoT for Empowering the Workers in the Food Industry

IoT implementation in the food industry can also help in empowering the workers through augmented reality, safety glasses, and other wearable sensors, thereby increasing the overall productivity and efficiency of their processes, workers, and the machinery they are using.

6.4 *IoT Solutions for the Food Industry*

We now discuss different IoT solutions [15] for the food industry.

6.4.1 City Crop

City crop is an intelligent indoor garden that provides intelligent indoor gardens to grow fruits, herbs, vegetables, greens, and edible flowers; they have implementation of automated climate control, automated livestock monitoring, and automated smart notifications that can be sent to the concerned stakeholders and also to the plant doctors.

6.4.2 Diagenetix

Diagenetix has the product bioranger which can help in detecting the presence of microbial diseases in the food. Bioranger is a small handheld device that connects with Android and instantly detects pathogens in the food.

6.4.3 Eskesso

Eskesso is a company that has the cooking sorcery which is basically for smart cooking. So, they have Wi-Fi-connected smart cooking device that can help in easy monitoring of the cooking status via the smartphone app. Smart cooking helps by placing the food packet and Eskesso device in a pot of water; selecting the recipe and starting via smartphone app one can get the food cooked in a smarter way through minimal involvement.

6.4.4 Culinary Science Industries: Flavor Matrix

Culinary science industry *Flavor Matrix* infuses foods and beverages with unique flavors; they collect data on the food ingredients, collect user data, and use different implementations of machine learning and data analysis to enhance the flavor of dishes and provide user-specific food and beverage pairing.

6.4.5 IntelliCup

IntelliCup is the smart cup solution which is a smart beverage vending machine which reduces the waiting time and increases the profit at beverage shops. These are like IoT-enabled cups which have NFC-integrated chips at the base of the cup, and they connect the cups to the mobile banking platform and IntelliHead which is a modular dispensing unit. This NFC chip helps in connecting each user to a cup. So, the cups are usable and made with biodegradable material. There are separate apps for the merchants and the customers, the customers create IntelliCup accounts using the app, they transfer the funds to the e-wallet, and the cups are linked to the e-wallet by scanning a QR code via the app and docking the cup on the dispensing unit using the IntelliHead. So, customers enjoy the beverage that is finally produced through this smart cup.

7 Conclusion

In this chapter, we focused on IIoT and its applications in the food industry, manufacturing industries, and healthcare. Most of the industries are transforming globally. They have been mandated to transform to be Industry 4.0 compliant. And they are transforming towards the adoption of IIoT technologies. So, we discussed the different aspects of both Industry 4.0 and IIoT. We also focused on the requirements of IIoT and design consideration for IIoT. It makes the industrial processes much more efficient and autonomous. In all of the different applications of IIoT, the core technological ideas remain the same and cannot be changed. So, the only thing that changes is the type of sensors that would be used. The specific requirements that a particular industry has are mostly dealt with in the application level, and at the device level more or less the concepts remain similar.

References

1. Miorandi, D., Sicari, S., De Pellegrini, F., & Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges. *Ad Hoc Networks, 10*(7), 1497–1516.
2. https://internetofthingagenda.techtarget.com/
3. Agarwal, P., & Alam, M. A. (2018). Use of ICT for sustainable transportation. *8th International conference on future environment and energy (ICFEE 2018), 150*(1), 1–7.
4. Agarwal, P., Chopra, K., Kashif, M., & Kumari, V. (2018). Implementing ALPR for the detection of traffic violations: A step towards sustainability. *Procedia: Computer Science, 132*(2018), 738–743.
5. Ramamurthy, A., & Jain, P. (2017). *The internet of things in the power sector: Opportunities in Asia and the Pacific*. ADB sustainable development working paper series, no. 48.
6. Vrat, P. (2014). *Material management*. Springer.
7. *IoT and analytics: Changing the reality of facility management.* https://www.softwebsolutions.com/resources/IoT-intelligent-building-for-facility-management.html
8. Boyes, H., Hallaq, B., Cunningham, J., & Watson, T. (2018). The industrial internet of things (IIoT): An analysis framework. *Computers in Industry, 101*, 1–12.
9. Xiaoxiao, X., Han, M., Nagarajan, S. M., & Anandhan, P. (2020). Industrial internet of things for smart manufacturing applications using hierarchical trustful resource assignment. *Computer Communications, 160*, 423–430.

10. Al-Turjman, F., & Alturjman, S. (2018). Context-sensitive access in industrial internet of things (IIoT) healthcare applications. *IEEE Transactions on Industrial Informatics, 14*(6), 2736–2744.
11. Kim, S., & Kim, S. (2018). User preference for an IoT healthcare application for lifestyle disease management. *Telecommunications Policy, 42*(4), 304–314.
12. Thahir, S. (2016). *6 applications of IoT in the healthcare industry*. CABOT.
13. Chen, B., Wan, J., Shu, L., Li, P., Mukherjee, M., & Yin, B. (2018). Smart factory of industry 4.0: Key technologies, application case, and challenges. *IEEE Access, 6*, 6505–6519.
14. Pal, A., & Kant, K. (2018). IoT-based sensing and communications Infrastructure for the fresh food supply chain. *Computer, 51*(2), 76–80.
15. Khan, W. Z., Rehman, M. H., Zangoti, H. M., Afzal, M. K., Armi, N., & Salah, K. (2020). Industrial internet of things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering, 81*, 106522.

Chapter 6
An Interactive Analysis Platform for Bus Movement: A Case Study of One of the World's Largest Annual Gathering

Emad Felemban and Faizan Ur Rehman

Abstract Analysis of traffic conduct, mainly in densely populated urban areas, provides an excellent opportunity to study traffic patterns and extract useful information to help in planning and development. During activities that draw in a massive number of people, such as religious pilgrimages or sporting events, collisions of automotive traffic flows can result in interruptions and unsafe situations for the subjects, often creating chaos and congestion. The scenario becomes more ambitious in Hajj when millions of pilgrims move in a restricted area during a fixed period of time. Hajj is a 5-day Islamic pilgrimage whereby millions of pilgrims from across the globe assemble in Makkah to perform a number of spatiotemporal rituals every year on fixed dates. This chapter presents an interactive platform that utilizes large-scale GPS traces to detect the motion of buses during Hajj. For a period of 2 months, GPS traces are gathered for over 17,000 buses used to carry pilgrims performing Hajj activities. An interactive big data platform was developed to analyze and visualize the massive amount of spatial data. The analysis was done for various stakeholders, including the bus companies. Using our map-based visualization, they were able to visualize the movement of buses; identify drivers' behavior, speed violations, and location of the violations; and determine the quality of data provided by various AVL providers. The information extracted can be used to generate an intelligent transportation system featuring schedule, evacuation, sustainability, resource optimization, and environmental and economic efficiencies to benefit stakeholders and improve the mobility of pilgrims throughout Hajj.

E. Felemban
Computer Engineering Department, College of Computing and Information Systems, Umm Al-Qura University, Makkah, Saudi Arabia
e-mail: eafelemban@uqu.edu.sa

F. Ur Rehman (✉)
Institute of Consultation Research and Studies, Umm Al-Qura University, Makkah, Saudi Arabia

LIG Lab, University of Grenoble Alpes, Saint-Martin-d'Hères, France
e-mail: fsrehman@uqu.edu.sa

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_6

Keywords GPS data · Trajectory · Hajj · Categorization · Big data · Clustering

1 Introduction

Hajj is an annual pilgrimage that millions of Muslims partake in every year. Mandatory for all healthy individuals of the Muslim faith, Hajj witnesses an attendance of millions of Muslims from all over the world. In Fig. 6.1, you can see the map of the Mashaer area. It depicts Mina, Muzdalifah, and Arafat, where pilgrims perform various spatiotemporal rituals. During the pilgrimage, Muslims travel via multiple routes to perform a number of rites. For instance, on day 9 of Dhul Hijjah, all pilgrims travel to Arafat. There, they spend the entire day and perform several rituals till sunset. This continues in Muzdalifah—where pilgrims head next and proceed with the rituals till the next day. They move from here for more rituals to other parts of the area. This specific simultaneous mass transportation results in traffic jams and movement blockage.

Seamless traffic flow has a significant role in major activities being hosted in a region. These could be sports events, religious gatherings, or any other festival/event. When traffic flow is not maintained properly, it could lead to catastrophic disruption and/or prolonged delays in all such events hosting a large crowd. There

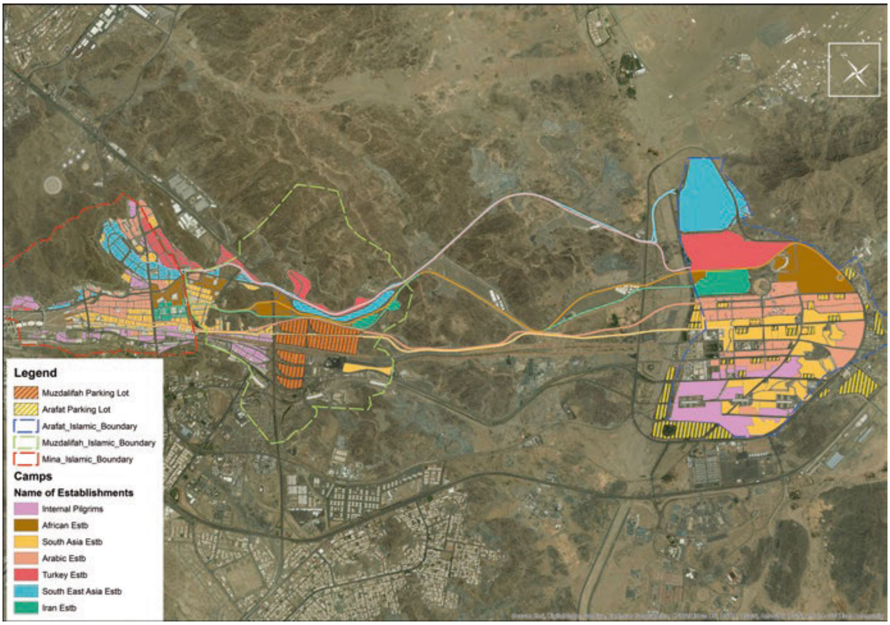


Fig. 6.1 Map-based distribution of establishments in Mashaer area (Mina, Muzdalifah, and Arafat), their assigned routes, and parking lots

is an immense need for supervision and proper scheduling of traffic flow in such scenarios. However, officials cannot make appropriate conclusions while managing a huge volume of crowd movement. They need to manage the distribution and regulation of pedestrians as well as traffic facilities while also handling and constantly monitoring other components like management measures, knowledge, and instruction to drivers. We have presented a case experiment [1] highlighting the management and regulation of the pilgrim traffic during Hajj. The need to amplify pilgrim traffic movement through all scheduled places for rituals during Hajj is evident from this experiment. It is absolutely essential to not only inspect and investigate mobility behavior, especially during the Hajj pilgrimage, to effectively control the traffic movement during this time.

Pilgrims from different parts of the world are assigned different establishments. Mashaer is split into seven establishments with their specific camps, parking spaces, and dedicated routes to handle the movement of traffic, as depicted in Fig. 6.1. Additional measures like blocking connection to internal roads are also taken to help prevent a stampede due to high crowd movement in the area. With dedicated spatiotemporal rituals—including specifically marked areas and itineraries for movement—decision-makers will find it effective to make quick and convenient choices during an emergency.

This chapter presents an interactive platform for all stakeholders, which will make it easier for them to make decisions regarding improvements in the pilgrim movement while they are staying in Makkah. GPS-traced data from 17,000 buses allocated for pilgrim movement was fed to a big data framework to analyze and provide a variety of results effectively. We analyzed the data based on multiple considerations, including the movement of buses, driver's behavior, speed limit violation, and start and end points. This system uses the data to present an interactive visualization for end users to observe traffic activity at any and all hours of the day during the entire Hajj season.

This chapter summarizes our previous research [2, 3] and provides an interactive platform that helps visualize the bus traffic movement during Hajj post-analysis. Section 2 provides a detailed explanation of GPS data, process of analyzing, and big data. In Sects. 3 and 4, the focus is on data management and platform overview, respectively. While Sect. 5 presents the implementation and result, Sect. 6 is used to showcase the data analysis. Finally, Sect. 7 highlights the conclusion and lists the future challenges.

2 Literature Review

Managing a huge crowd along with transportation, especially in a controlled area, can be a challenging task. Moreover, during religious events like Hajj, interference with daily traffic is expected. In such a situation, recurring traffic jams are a sign of increased obstruction of traffic. This makes controlling crowd movement an even more arduous task for the officials who focus on ensuring that pilgrims enjoy their travel experience. Now, scientists are looking towards GPS and big data analytics to envision and verify brand new solutions to help optimize traffic movement.

2.1 GPS Data and Analysis

On their route to Arafat and Muzdalifah and back to Mina by bus, it is common for pilgrims to get stuck in long queues of traffic. Over the last 10 years, there has been a massive increase in the number of pilgrims and, subsequently, buses. Now the roads and routes are incapable of handling this extra inflow, leading to severe traffic issues. With the help of the global positioning system, or GPS as it is commonly known, the first and most vital step is the identification of travel modes. Data quality test also evaluates data consistency. Measures like consistency satisfaction, reliability, and status of completion are duly distinguished in general data output. Data loss, reduction, and dispute are also addressed and kept to a minimum [4]. With the progress of intelligent transportation systems (ITS), the importance of traffic data mining is increasing as well [5, 6]. Recently, GPS technology has witnessed an extensive utilization because of which GPS-related vehicle data source is a treasured data source for ITS. However, it presents a challenge for data mining due to its volatile estate. Authors in [7] have utilized appropriate data density and data ideality to effectively provide an explanation of the quality of GPS traffic data, leading to the proposal of a multidimensional cube approach to data quality.

All over the world, GPS systems are used to a great extent and can easily provide long-term consistency along with apt semipermanent precision for location data. At the same time, it could be vulnerable to external disturbances like electronic signal jamming or something as simple as the loss of signal in a tunnel. The solution for this issue is using an integrated navigation system (INS) [8]. The authors in [9] have proposed a unique propagation neural network as a solution for various problems, including topology, atmospheric, and natural formations. All of this can be remotely monitored, and data acquisition is maintained with the help of GPS technology. The proposal for GPS data acquisition and analysis software over the IP platform is presented in [10].

During tawaf—the seven-time circumambulation of the Kaaba—GPS can quantify both location and time with precision [11]. In [12], we have presented a spatio-temporal model of the mobility of a pilgrim between Mina and Jamarat. This, then, can be used to construe the restrictions encountered by pilgrims during movement from one ritual place to another. Smartphones with sensors can be used to collect data which can be implemented for road quality assessment. In [13], the authors have done precisely this. They have evaluated various on-road phenomenon and driving incidents by collecting data with the help of smartphone sensors and analyzing it on different parameters to recognize patterns. The authors in [14] have proposed a unique data study of road conditions based solely on mobile GPS sensors and accelerometers. It is recommended to find the approximate travel speed on the GPS data retrieved from a road network by using machine learning techniques [15].

The researchers in [16] showcase a complex study of the complete freight route, including the environmental impact of highways, as well as driver's behavior, but using the proposed application to be used to collect GPS data. In [17], authors have proposed predictions of blockage in traffic based solely on trajectory data from GPS. In order to recognize various activities and trips, Fang et al. provides a dedicated research scheme that detects information on the purpose and mode of transport [18]. While in [19] authors have provided a GPS data-based solution for predicting the flow of traffic along with an application that simulates traffic, in [20], a proficient system has been developed that detects traffic congestions and accidents using real-time GPS data procured from trackers or smartphones of drivers.

2.2 Big Data

Big Data is, quite aptly, a big thing in research and technology these days. It is a collection of data from multiple sources, which not only is huge in volume but also witnesses exponential growth over time. Some of the most commonplace data processing techniques are part of big data. It includes everything from artificial intelligence to data mining, from machine learning to data fusion, and even social networking. Today, intelligent transportation system (ITS) analysis projects around the world are employing big data research. In [21], authors have analyzed the history as well as relevant features of both big data and ITS while also examining their relevance in ITS analysis.

Big data tech can solve a number of problems and help cut costs if it is incorporated into urban transport systems, especially smart systems. It can help automate operations of the network while also improving the efficacy of problem-solving. It is a central repository of massive data that is free to be analyzed for anything and everything using various data mining techniques. In [22], the value of big data tech, especially in intelligent public transport systems, is discussed in detail.

Users can now be provided with greater opportunities to deliver smart services now than ever with the help of ITS, especially with great advancement in big data and the Internet of Things (IoT) [23, 24]. In [25], the case study proposed a smart intelligent transportation system by applying big data technology capable of handling real-time issues such as location tracking and parking. In [26], the authors proposed a novel technology that amalgamates IoTs and intelligent transportation systems in order to improve the current transportation systems. The methodologies [27, 28] make use of the sensors that closely observe the environment to notify the drivers about the positioning of the device and its particulars that helps to display the ongoing bus path to the passengers. Therefore, the proposed technology regulates the number of tickets acquired as it determines its effectiveness.

Over the last few years, ITS has managed to generate a massive amount of big data only on traffic. This poses a unique problem of scalability whenever road traffic data is being analyzed and investigated. In [29–31], authors have used MapReduce to overcome a similar problem and help generate a precise visualization so that traffic information is available effectively. This allows users to discover required information quickly, like weather trends and other related things. The authors have suggested a timeline analysis for traffic big data in [32], and a dynamic predictive algorithm to analyze big data in [33]. Together, these will help enhance traffic forecasts in real time.

While big data helps in resolving real-world problems like transportation, a substantial amount of information is required for it. Visual analysis of transit systems, crowdsourcing, and other data-based services are needed to provide viable solutions. In [34], authors have summarized the various sources, approaches, and application systems required.

The automotive sector views each automobile on the road as a unique data source, as does transport engineering. This leads to massive data imports. When it comes to various data obstacles like range, value, velocity, and quantity, big data provides an umbrella concept to deal with it effectively. In [34], authors have also suggested a distributed data model that works around these problems by creating a different design option for each data processing step.

The IoT in ITS plays a crucial role using advanced technologies for off-line analysis and real time, such as live tracking, seat availability, smart parking, and many more. In Hajj, millions of individuals—pilgrims—traverse from one place to another at the same time to perform certain specific rituals. This makes for a unique event. The state-of-the-art systems have been designed to mine and analyze the city's GPS data from buses and taxis. However, during Hajj, buses on both sides of the road move in the same direction, which creates a peculiar problem and requires a different approach to tackle the issue by considering the large gatherings, spatio-temporal rituals, road capacity, and routes.

3 Big Data Components

In order to develop any system, data is the key and the primary source. Usually obtained from multiple channels like cameras, sensors, and IoT devices, stored on a variety of local servers or on clouds and in different formats, this data needs to be handled with expertise. This is where big data comes in.

The steps of big data are shown in Fig. 6.2. The first step in any data management process is preprocessing. This involves the removal of irrelevant data and completing any missing data. Data mining requires the use of data transformation to ensure processes like attribute selection, discretization, normalization, and hierarchy generation. Data reduction can be a long-drawn process since it analyzes massive amounts of data; however, it is absolutely essential since it helps increase storage efficiency and reduces the cost of analysis. It employs processes like data cube dimensionality reduction, aggregation, numerosity reduction, and attribute subset selection. Thereafter, depending on the type of data, it can be presented as tables, graphs, or charts.

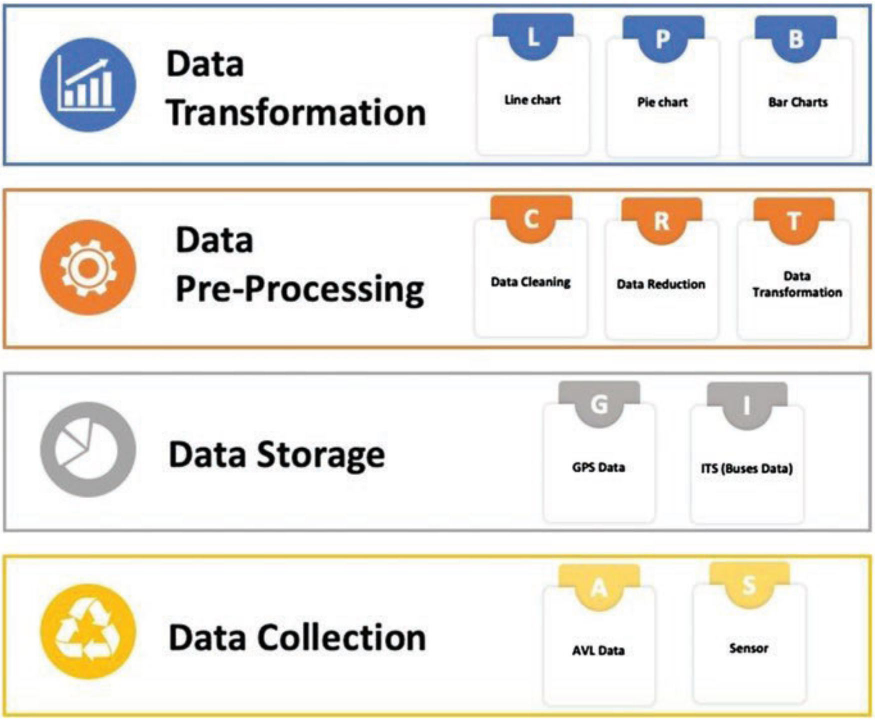


Fig. 6.2 Big data framework

3.1 Data Collection

Automatic vehicle location (AVL) is used to collect data during the Hajj season. Data from 20,380 buses were collected from 10 AVL service providers by the Ministry of Hajj and Umrah. The official transport authority of the Ministry, Naqaba, was our main source of the data. We collected approximately 737,145,410 location history records, including data from 20,000 buses, operators of these buses, their offices, companies, garage, establishments, various zones, and movement stages. This data was then arranged in 98 relational tables. Location history, with record_id, bus_id, company_id, angle, latitude and longitude, as well as GPS signal, speed, and finally record_time, is depicted in Table 6.1.

Figure 6.3 provides more detail on the dataset that has been used in this case study. The MYSQL data has not been cleaned properly, and as such, quite a bit of it is rendered redundant. It requires better cleaning and preprocessing. In Fig. 6.4, the relationships between buses, offices, establishments, and zones are depicted. It should be noted that this data is most important and absolutely essential for the calculation of bus count as per establishment and routes.

Table 6.1 Violation details of bus no. 1

Start timestamp	End timestamp	No of consecutive observations	Violation duration in seconds	In minutes
2018-08-23 04:47:44	2018-08-23 04:55:44	2	480.0	8
2018-08-23 06:05:46	2018-08-23 06:13:45	2	479.0	7.98
2018-08-23 06:31:55	2018-08-23 06:31:55	1	0.0	0
2018-08-24 05:51:27	2018-08-24 05:51:27	1	0.0	0
2018-08-25 06:29:29	2016-09-25 06:37:29	2	480.0	8
2018-08-25 11:43:35	2016-09-25 12:19:36	17	2161.0	36.01
2018-08-25 12:29:43	2016-09-25 12:35:37	3	354.0	5.90

3.2 Data Storage

Collected from multiple AVL sources, the data is stored in a distinct MySQL database. From there, it is transferred and converted to MS SQL server dump format. From here, a single query may take as long as 15 minutes, or in certain cases even longer, to retrieve or access.

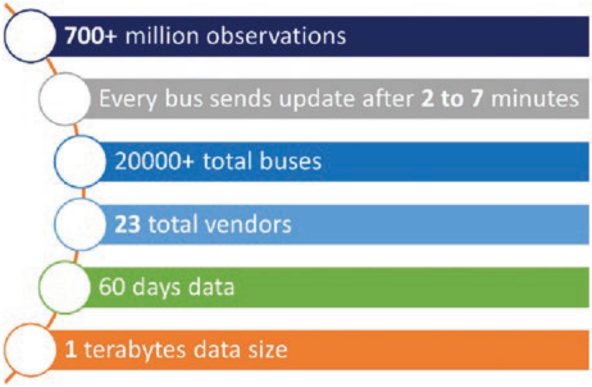
3.3 Data Preprocessing

Preprocessing is one of the most significant and time-taking steps of data analytics. The gathered data may not always be in the correct format, or it may contain redundancies and missing information. This is where data preprocessing comes in. When applied to existing data, it creates ranges out of various variables. Furthermore, data cleaning is applied to fix some missing and/or incorrect information.

3.4 Data Transformation

Before we can begin the analysis, data transformation is an essential step as it helps transform existing data into a new set of attributes. This helps in making an analysis of this data. From the records of 20,000 buses, we were able to acquire some important data points that were missing. With the help of data cleaning and data extraction techniques, we were able to successfully extrapolate these data points. We were even able to obtain some new features, for instance, the total distance that the buses

Fig. 6.3 Summary of dataset



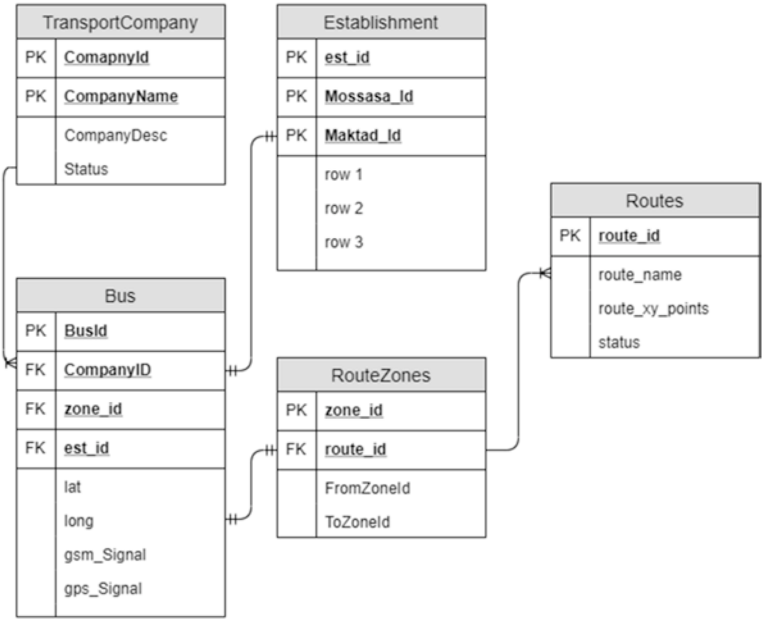


Fig. 6.4 ER diagram of the partial schema

traveled or the distance between the buses. We could even calculate the speed of any bus at any given point of time. With the help of these extra data points, we were able to compute even more data points, like the total distance that each bus under a certain establishment traveled in a certain area and even the time a bus spent in any parking. With the help of data transformation, all this and more was possible. We now had all bus routes, parking spots, bus stations, and complete routes that each establishment was running as well as the location of their offices.

4 Platform Overview

All the data received from Naqaba—which is the official transport authority of the Ministry of Hajj and Umrah—is saved in the MS SQL format for data analysis. However, in this format, even the most basic query can take 13–15 min easily, which is very inefficient for our study. We took this data and put it in the big data platform using a Cassandra cluster to speed up the process. In Fig. 6.5, you can see that we developed a high-level view to analyze this data in a big data platform.

MS SQL service and master data service are both covered in the data lake layer. While MS SQL contains the data received from Naqaba, for visualization purposes, we use the master data. This is how various data points like offices, establishments, or even bus numbers are visualized.

The big data layer is composed of two elements—one is a Cassandra cluster and the other is what is known as the big data aggregation service. Cassandra cluster is used to clean the location history data since it uses a NoSQL database management system and a distributed wide column store and improves the competency and scalability of the given data. This also helps remove any noise as it uses an ETL engine. As for the big data aggregation system—it is nothing but a combination of Presto and Hadoop. Presto can easily analyze queries based on data in the HDFS, or Hadoop Distributed File System, format. Social media giants like Facebook employ the use of Presto in their operations. The major difference between Hadoop and Cassandra lies in how they perform analyses. While Cassandra handles queries based on time, Hadoop is for queries based on batch.

REDIS is an in-memory arrangement store, used as a distributed, in-memory key-value database, cache, and message broker, with optional durability. It supports different sorts of abstract data structures, like strings, lists, maps, sets, sorted sets, HyperLogLogs, bitmaps, streams, and spatial indices. Using REDIS cache helps improve the system's overall performance.

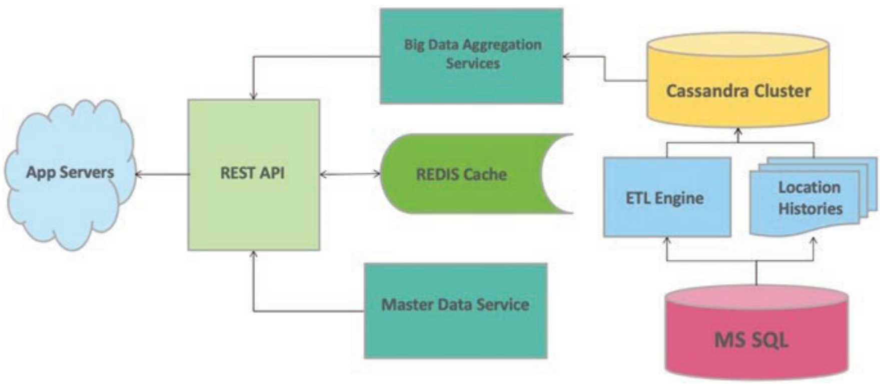


Fig. 6.5 Overview of the platform

For any front-end requests, Naqaba REST API has enlisted various APIs. This works by front-end calling the API upon receiving a request. From there, data is fetched from various services, including REDIS cache big data aggregation as well as master data services. Thereafter, the front end visualizes this data and provides results. By using this interface, an end user can select various variables like the office (Maktab), establishments (Mosasa), their number, bus number, the company, and even the route and access their results in a visualized manner.

5 Implementation and Results

In order to be able to visualize the movement of 20,000 buses, it was imperative to develop the big data-based interactive platform. It is capable of performing numerous analyses like identification of driver's behavior, violations of speed, and their area with the help of simple map-based visualizations. For this platform, all front-end implementation is handled by a leaflet JavaScript superimposed on Google Map's layer. Python coding is used for ETLs and back end, and for REST APIs, application server's queries are handled in PHP. The Cassandra cluster has been given two dedicated nodes with 100GB exclusive storage and a fast 4GB RAM. The workstation powering this system is equipped with an Intel Xeon CPU E5–1620 v2 @ 3.70 GHz processor, 64 GB RAM, and a 2 TB SSD storage for effective computing. By simply increasing the dedicated nodes for the Cassandra cluster, the entire platform can be scaled up to enable the storage of data from yet another Hajj season for the purpose of comparison or machine-language algorithmic prediction.

Once all the data of 20,000 buses were transferred from MS SQL to the big data platform, for the queries that used to take a long time with MS SQL, the speed increased exponentially. For instance, queries that earlier took a few minutes were now resolved within 300 milliseconds with Cassandra's two nodes. Figure 6.6 is a depiction of the query response time of five buses once it is migrated to our big data-based platform.

In Fig. 6.7, you can see the front end of the platform developed for end users to visualize bus movement with the help of GPS-traced data. It contains the data of all the buses that were used during the 2018 Hajj pilgrimage (1439H). Similarly, in Fig. 6.8, you can see the panel that end users can use to easily filter the data based on numerous parameters like bus number, establishment, and offices. They can also track the bus location (within the boundary or not) and movement (moving or parked).

Let us look at some more visualizations now. Figure 6.9 showcases various GPS data points for all the buses allocated to the South-East Asia Establishment during the 2018 Hajj. Figure 6.10 is a representation of other notable features. These include (a) the number of buses that were allocated to each office, (b) all the data for the specific company-owned buses, and (c) data extraction as well as visualization within a user-specified boundary as well as its last known location, whether it was in the Mashaer region or outside. The red points indicate parked vehicles, whereas

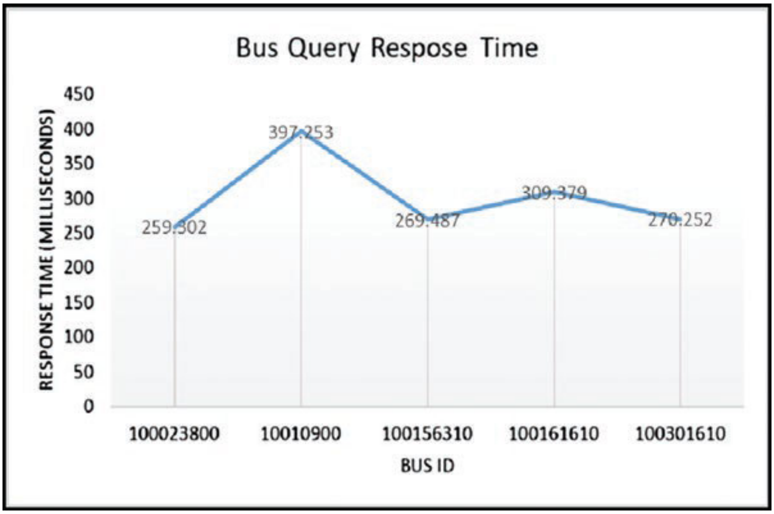


Fig. 6.6 Buses query time on our platform

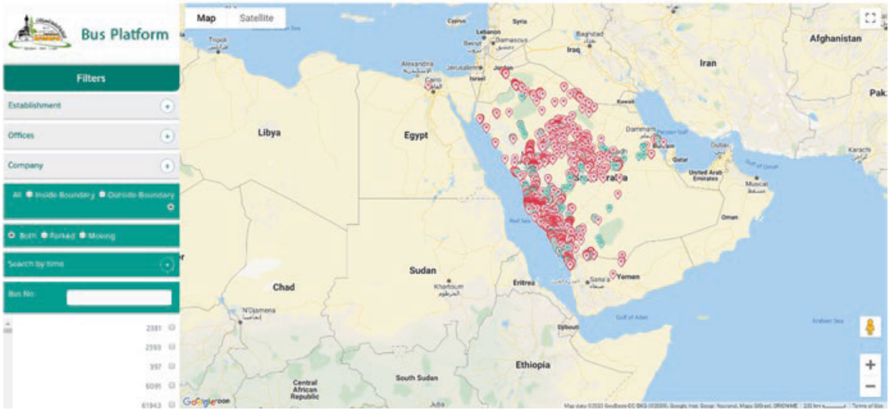


Fig. 6.7 Interactive platform to visualize the buses data

green points are indicative of vehicles on the move. Moreover, Fig. 6.11 provides an interface to visualize the movement of an individual bus by clicking on the bus number as shown in Fig. 6.11.

6 Data Analysis

This section is dedicated to the analysis of the data based on the speed of the bus in order to understand various other parameters like driver's behavior, violations, and the area of these violations with the help of data visualization on the map. In

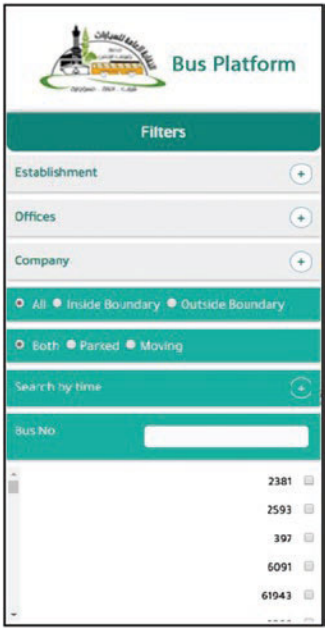


Fig. 6.8 Filter panel of the interactive platform

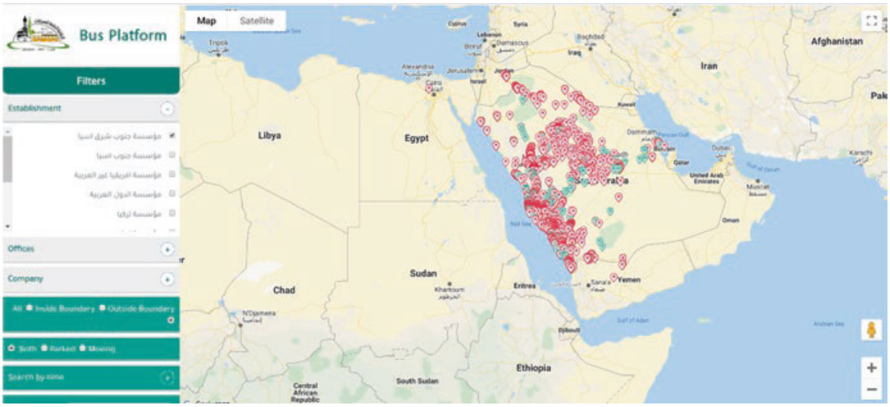


Fig. 6.9 Zones and parking lots in Mashaer area based on the establishment

Fig. 6.12, you can see the speed profile of any driver who was employed during the 2019 Hajj pilgrimage in the form of a timeline. In this particular figure, it is clear that bus no. 1 maintained varying speeds throughout the Hajj pilgrimage of 2018 and reached a maximum of 102 km/h, while the minimum speed at any point beyond the specified limit for this bus was 81 km/h.

The likelihood of speed violations is quite high. In Table 6.1, we have presented the duration of any violation in minutes and seconds with the help of beginning and ending timestamps. This was possible only by identifying various bus speeds and

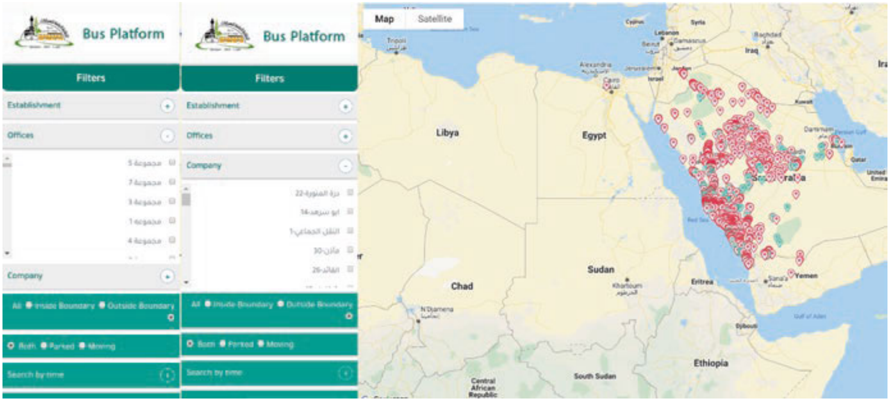


Fig. 6.10 Features of the platform

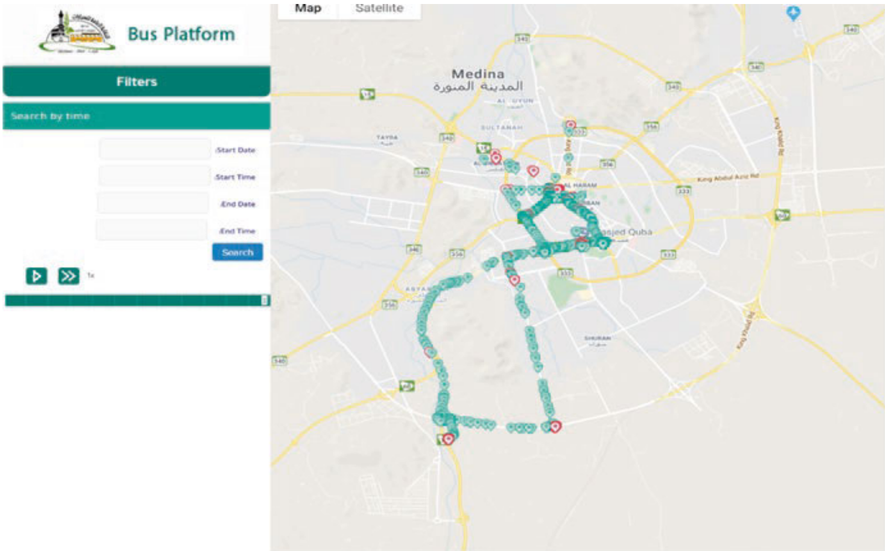


Fig. 6.11 Movement of the individual bus

continuous observations of instances where the speed exceeded the limit of 80 km/h. Now, in order to provide an in-depth analysis of this data, we will classify each violation into respective categories based on their duration in Table 6.2.

In order to identify the intensity of any violation of the common locations, we have used heat and geo maps, as depicted in Figs. 6.13 and 6.14. It is clear that the bus has 145 violations while exiting Jeddah on the Makkah-Madinah highway.

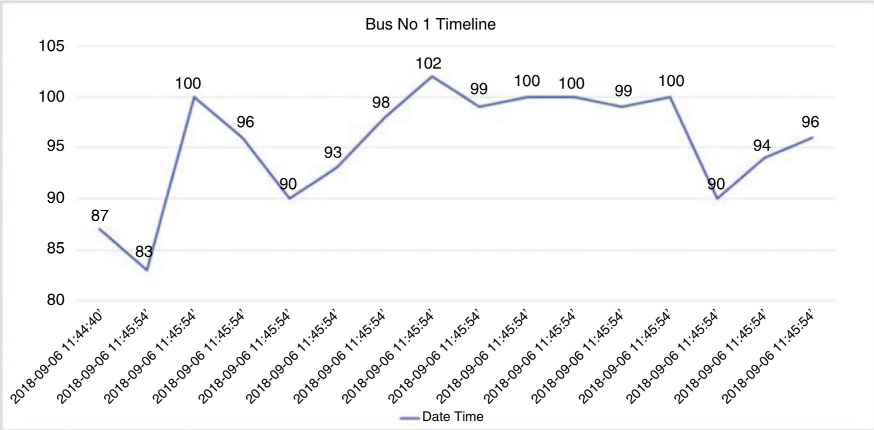


Fig. 6.12 Driver's behavior (speed timeline)

Table 6.2 Violations classification

Severity	Explanation
Once	Some violations do not have continuous number of violating observations. These violations occur only in single observation, so we cannot get a starting timestamp and ending timestamp, so we call it **once**
Normal	If the violation's duration is less than 10 minutes, then we call it **Normal** violation
High	If the violation's duration is between 10 and 20 minutes, then we say it is **high**
Severe	If the violation's duration is more than 20 minutes, then we say it is **severe**

7 Conclusion

The Hajj pilgrimage is an incredibly challenging event for the organizers and all those involved. It is not easy to control the movement of millions of pilgrims as they travel from one place of ritual to another. Held in Makkah, Saudi Arabia, this pilgrimage is the most important pilgrimage a Muslim person makes in their entire life, and for many, it is only a once-in-a-lifetime opportunity, making it even more challenging for organizers to ensure that they have a safe, peaceful, and enjoyable time during their stay. The possibility of a stampede when such a huge crowd is moving from one place to another is extremely high. In this study, we have presented a unique interactive platform that can track the movement of all the buses during the Hajj and provide numerous solutions in the process. In our prototype, we studied the movement, routes, parking spaces, establishments, offices, travel times, speeds, and violations in the Mashaer area at different and multiple times during the day for the entire period of the Hajj pilgrimage. We successfully identified the best and worst drivers, the areas with the most violations, and the companies with the best and worst records. We have successfully demonstrated that our platform is fit

Fig. 6.13 Bus violations heatmap

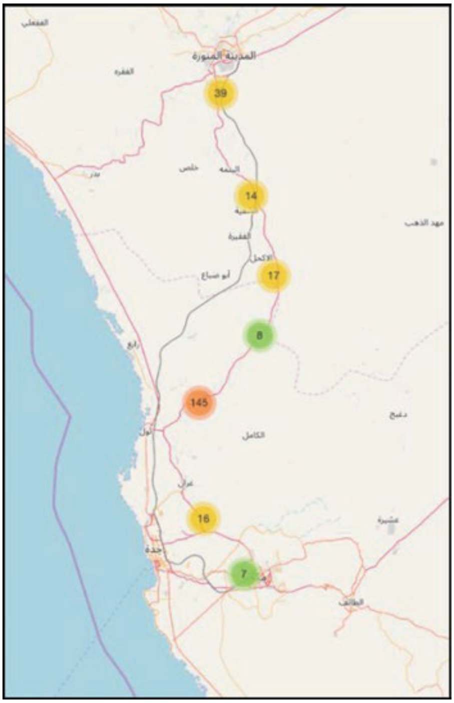


Fig. 6.14 Bus violations frequency

for storing, analyzing, and visualizing big spatial data. This can be amazing for all stakeholders, organizers, and decision-makers working in the area of intelligent transport systems all over the world. In the future, we hope to evolve this into a capable predictive system with the help of machine learning, so that analytics like pickup and drop-off time and travel time for each shuttle can be calculated in real time.

References

1. Tayan, O., & Al-binali, A. (2013). Evaluation of a proposed intelligent transportation framework using computer network concepts: A Case Study for Hajj-Pilgrim Traffic Monitoring and Control. *International Journal of Computer Science and Applications, 10*(2), 325–331.
2. Felemban, E., Rehman, F. U., Biabani, A. A., Naseer, A., & Alabdulwahab, U. (2019, December). Towards building an interactive platform for analyzing movement of buses in Hajj. In *Proceedings – 2019 IEEE international conference on Big Data, Big Data 2019* (pp. 3775–3778). IEEE. https://doi.org/10.1109/BigData47090.2019.9005521
3. Felemban, E., Rehman, F. U., Wadood, H., & Naseer, A. (2019, December). Towards building evacuation planning platform using multimodal transportation for a large crowd. In *Proceedings – 2019 IEEE international conference on Big Data, Big Data 2019* (pp. 4063–4066). https://doi.org/10.1109/BigData47090.2019.9006226
4. Aebi, D., & Perrochon, L. (1993, October). Towards improving data quality. In CiSMOD (pp. 273-281).
5. Turner, S. M., & Albert, L. P. (2000). ITS data quality control and the calculation of mobility performance measures (No. REPT-1752-2). Texas Transportation Institute.
6. Tang, L. A., Yu, X., Kim, S., Han, J., Peng, W. C., Sun, Y., Leung, A. & La Porta, T. (2012). Multidimensional sensor data analysis in cyber-physical system: An atypical cube approach. *International Journal of Distributed Sensor Networks, 8*(4), 724–846.
7. Zhang, C., Zheng, Z., Zhang, F., & Ren, J. (2011). Multidimensional traffic GPS data quality analysis using data cube model. In *Proceedings 2011 International conference on transportation, mechanical, and electrical engineering, TMEE 2011* (pp. 307–310). IEEE. https://doi.org/10.1109/TMEE.2011.6199204
8. Budiyono, A. (2012). Principles of GNSS, inertial, and multi-sensor integrated navigation systems. *Industrial Robot: An International Journal, 39*(3). https://doi.org/10.1108/ir.2012.04939caa.011
9. Wang, G., Xu, X., Yao, Y., & Tong, J. (2019). A novel BPNN-based method to overcome the GPS outages for INS/GPS system. *IEEE Access, 7*, 82134–82143. https://doi.org/10.1109/ACCESS.2019.2922212
10. Chamorro, H. R., Cruz, A. M., Niño, F., Ferro, R., Gomez, E., & Cantor, E. L. (2011). GPS data analysis and acquisition software over IP platform. In *2011 IEEE 12th annual wireless and microwave technology conference, WAMICON 2011*. https://doi.org/10.1109/WAMICON.2011.5872906
11. Koshak, N., & Fouda, A. (2005). Analyzing pedestrian movement in Tawaf using GPS and GIS. *Proceedings of Remote Sensing Arabia*, 7–11.
12. Koshak, N. (2005). A GIS-based spatial-temporal visualisation of pedestrian groups movement to and from Jamart area. In *Proceedings of computers in urban planning and urban management (CUPUM '05) conference June 29–July 1, 2005, London, UK*, pp. 1–12, [Online]. Available: http://www.cadgis.com/n/publications/jamarat_movement_simulation.pdf.

13. Kalra, N., Chugh, G., & Bansal, D. (2014). Analysing driving and road events via smartphone. *International Journal of Computers and Applications, 98*(12), 5–9. https://doi.org/10.5120/17233-7561
14. Amirgaliyev, B. Y., Kuatov, K. K., & Baibatyr, Z. Y. (2017, July). Road condition analysis using 3-axis accelerometer and GPS sensors. In *Application of information and communication technologies, AICT 2016 – Conference proceedings*. https://doi.org/10.1109/ICAICT.2016.7991662
15. Kyaw, T., Oo, N. N., & Zaw, W. (2019, January). Estimating travel speed of Yangon road network using GPS data and machine learning techniques. In *ECTI-CON 2018 – 15th international conference on electrical engineering/electronics, computer, telecommunications and information technology* (pp. 102–105). https://doi.org/10.1109/ECTICon.2018.8619908
16. Pluvinet, P., Gonzalez-Feliu, J., & Ambrosini, C. (2012, January). GPS data analysis for understanding urban goods movement. *Procedia Social and Behavioral Sciences, 39*, 450–462. https://doi.org/10.1016/j.sbspro.2012.03.121
17. Sun, S., Chen, J., & Sun, J. (2019, May). Traffic congestion prediction based on GPS trajectory data. *International Journal of Distributed Sensor Networks, 15*(5), 155014771984744. https://doi.org/10.1177/1550147719847440
18. Fang, Z., Lv, J. Y., Tang, J. J., Xiao, W., & Fei, G. (2018, October). Identifying activities and trips with GPS data. *IET Intelligent Transport Systems, 12*(8), 884–890. https://doi.org/10.1049/iet-its.2017.0405
19. Necula, E. (2014, December). Dynamic traffic flow prediction based on GPS data. In *Proceedings - International conference on tools with artificial intelligence, ICTAI*, December 2014, pp. 922–929. https://doi.org/10.1109/ICTAI.2014.140.
20. D'Andrea, E., & Marcelloni, F. (2017, May). Detection of traffic congestion and incidents from GPS trace analysis. *Expert Systems with Applications, 73*, 43–56. https://doi.org/10.1016/j.eswa.2016.12.018
21. Pevec, D., Vdovic, H., Gace, I., Sabolic, M., Babic, J., & Podobnik, V. (2019, July). Distributed data platform for automotive industry: A robust solution for tackling big challenges of big data in transportation science. In *ConTEL 2019 – 15th international conference on telecommunications, proceedings*. https://doi.org/10.1109/ConTEL.2019.8848542
22. Liu, Y. (2018, January). Big Data technology and its analysis of application in urban intelligent transportation system. In *Proceedings – 3rd international conference on intelligent transportation, Big Data and Smart City, ICITBS 2018, April 2018* (pp. 17–19). https://doi.org/10.1109/ICITBS.2018.00012
23. Kusmawan, P. Y., Hong, B., Jeon, S., Lee, J., & Kwon, J. (2014). Computing traffic congestion degree using SNS-based graph structure. *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, 2014*, 397–404. https://doi.org/10.1109/AICCSA.2014.7073226
24. Damaiyanti, T. I., Imawan, A., & Kwon, J. (2014). Extracting trends of traffic congestion using a NoSQL database. *2014 IEEE fourth international conference on big data and cloud computing*, (pp. 209–213), https://doi.org/10.1109/BDCloud.2014.97
25. Muthuramalingam, S., Bharathi, A., Rakesh Kumar, S., Gayathri, N., Sathiyaraj, R., & Balamurugan, B (2019). IoT based intelligent transportation system (IoT-ITS) for global perspective: A case study. In *Internet of Things and Big Data analytics for smart generation* (Intelligent systems reference library) (Vol. 154, pp. 279–300). Springer.
26. De La Garza, J. M., Taylor, C. J. E., & Sinha, S. K. (2014). An Integrated and a smart algorithm for vehicle positioning in intelligent transportation systems, PhD diss., Virginia Tech, pp. 2–115.
27. Bojan, T. M., Kumar, U. R., & Bojan, V. M. (2014). An internet of things based intelligent transportation system. In *2014 IEEE international conference on vehicular electronics and safety* (pp. 174–179). https://doi.org/10.1109/ICVES.2014.7063743
28. Ashokkumar, K., Baron Sam, R., & Arshadprabhu, B. (2015). Cloud based intelligent transport system. *Procedia Computer Science, 50*, 58–63.

29. Chen, X. Y., Pao, H. K., & Lee, Y. J. (2014). Efficient traffic speed forecasting based on massive heterogeneous historical data. In *Proceedings – 2014 IEEE international conference on Big Data, IEEE Big Data 2014* (pp. 10–17). https://doi.org/10.1109/BigData.2014.7004425
30. Zheng, X., & Wang, S. (2014, July). Study on the method of road transport management information data mining based on pruning Eclat algorithm and MapReduce. *Procedia Social and Behavioral Sciences, 138*, 757–766. https://doi.org/10.1016/j.sbspro.2014.07.254
31. Yu, J., Jiang, F., & Zhu, T. (2013). RTIC-C: A big data system for massive traffic information mining. In *Proceedings – 2013 international conference on cloud computing and Big Data, CLOUDCOM-ASIA 2013* (pp. 395–402). https://doi.org/10.1109/CLOUDCOM-ASIA.2013.91
32. Imawan, A., & Kwon, J. (2015, December). A timeline visualisation system for road traffic big data. In *Proceedings – 2015 IEEE international conference on Big Data, IEEE Big Data 2015* (pp. 2928–2929). https://doi.org/10.1109/BigData.2015.7364125
33. Yang, Z. X., & Zhu, M. H. (2019, March). A dynamic prediction model of real-time link travel time based on traffic Big Data. In *Proceedings – 2019 international conference on intelligent transportation, Big Data and Smart City, ICITBS 2019* (pp. 330–333). https://doi.org/10.1109/ICITBS.2019.00087
34. Zheng, X., et al. (2016, March). Big Data for social transportation. *IEEE Transactions on Intelligent Transportation Systems, 17*(3), 620–630. https://doi.org/10.1109/TITS.2015.2480157

Chapter 7
Vehicle Payload Monitoring System

Nishant Yadav, Nishita Yadav, and Anjali Garg

Abstract In the present day, the dashboard of most vehicles consists of a speedometer, a temperature indicator, a pressure indicator, and fuel gauges. No provision has been provided for indicating load on the vehicle. The load of the vehicle is measured in the specific weight range. This chapter presents the hardware implementation of sensing of the load on the vehicle and displaying on the LCD wherein the deflection of leaf spring is used to measure the load. The load will get displayed on the display module which will be integrated in the driver cabinet. Automatically calculating or estimating the total payload delivered to the vehicle by the excavator's work tool is one way to measure the total weight of the material loaded into a truck. This approach of measuring the load on the vehicle has proved to be effective in solving many issues related to drivers, vehicle owners, and government along with providing various safety measures. The hardware developed in this chapter will help in keeping track of the total weight of each payload.

1 Introduction

Owing to the expansion of the vehicle transportation system and concerns that we lack accurate vehicle payload measurements, a system that can measure the vehicle's onboard weight at any time has become important in recent years. It is critical to build and incorporate modern systems for monitoring and controlling these devices to make fast decisions and integrate planning. Furthermore, transportation management strengthened because of increased rivalry between modes of transportation and businesses, resulting in a rise in the number of fully loaded trucks and their full weight. There have been a lot of illegally loaded vehicles recently, and road damage is directly proportional to the axle weight to the fourth power.

N. Yadav · N. Yadav · A. Garg (✉)
Department of Electrical, Electronics and Communication Engineering, The NorthCap University, Gurugram, Haryana, India
e-mail: Nishant17ecu029@ncuindia.edu; nishita17ecu014@ncuindia.edu

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_7

Traditionally, vehicle weights were weighed and obtained by putting the vehicle on a scale while it was stationary.

Therefore, making a system that can measure the weight of the vehicle with the extra load seems to be necessary. The load on the vehicle is sensed and shown on the LCD in this project, and the load is measured using the deflection of the leaf spring. At four different locations on the chassis, four ultrasonic sensors are bonded. By measuring the height between the chassis and axle, ultrasonic sensors are used to measure the deflection caused by the load. With the assistance of the control unit, the algorithm can transform the height difference measured by the sensor into the load applied to the chassis. The load (result) will be shown on a display module that will be built into the driver's cabinet. The weighting of vehicles and their geographical position will be very important to use in many departments and organizations, particularly municipalities that use vehicles.

2 History

The current rate of growth in the logistics sector, as well as the increasing global population, necessitates the use of many more vehicles. As a result, vehicle overloading is becoming more severe. As a result, vehicle overloading is to blame for India's rapid degradation of highways. The rise in traffic accidents caused by vehicle overloading has become one of India's most pressing traffic management concerns. When measuring a car, a conventional static weighing device is used. Established weighing technologies, in which measuring devices must be mounted in the concrete of roads to calculate vehicle weight, have a number of unanswered issues. They are inconvenient to maintain as a result of this. In India, heavy-duty vehicles do not have easy access to weighing stations.

2.1 Existing Solution

1. A simple system that uses strain gauges as a sensing element to display vehicle load on an LCD: But this system is not very effective as load cells are fragile and costly.
2. A device that utilizes the innovation to provide a more precise method for monitoring the vehicle's weight and/or load-based tire pressure: Different tire pressures in a truck, on the other hand, are rare. Furthermore, when the car is accelerated, motion-induced heating causes these tires to have different tire pressures. As a result, correct weight estimation is not possible.

2.2 Proposed Solution

This chapter describes the implementation of a device wherein the load on the vehicle is sensed and displayed on the LCD where the deflection of leaf spring is used to measure load. Four ultrasonic sensors are bonded onto the chassis at four different places. Ultrasonic sensors are used to measure the deflection caused by the load by measuring the height between the chassis and axle. After the sensor measures the height difference, the algorithm will convert this height difference into the load applied on the chassis with the help of the control unit. The load (result) will get displayed on the display module, which will be integrated in the driver cabinet.

2.3 Software Algorithm: Arduino IDE

All Arduino boards can be interfaced using an integrated programming environment (IDE) that is open-source called Arduino IDE, which is supported by Windows, Linux, and Mac OS X. This IDE can be used to build and upload programs to the Arduino Board.

2.4 Advantages

1. Low-cost solution, hence affordable.
2. Portable.
3. Continuous net payload measurement to notify any pilferage, especially when vehicle is stopped.
4. Progressive net payload measurement to reduce the time taken for loading/unloading and improve productivity especially in B2B transportation.
5. Ensure correct loading specially to meet regulations.
6. Improve productivity and save penalty especially in market load operations.
7. Correct loading.
8. Prevent accidents and improve the safety of the driver.
9. Improve vehicle life and reduce downtime.

2.5 Limitations

1. Arduino boards require constant power to run.
2. The system needs to be swamp-proof, so that it becomes more durable.

3 Hardware

This section presents the whole framework, from the Arduino Uno board to the appliance. The data to the Arduino is provided by the integrated system using Arduino Uno board and ultrasonic sensors which provides data to Arduino and output from it is displayed on the LCD.

3.1 Arduino Uno

The Arduino Uno microcontroller board is built using the ATmega328. 6Mhz ceramic resonance, 6 analogue channels, USB connection, ICSP header, restart button, 14 optical I/O pins, and an ICSP header are among the features [1]. It is quite easy to use this board in different applications since everything is included on the board that supports the microcontroller and can be operated simply using a USB connection to connect to a computer device to configure and operate the board. The Arduino software may be used to program the UNO (IDE). To import fresh programming without the need for an external hardware programmer, select “Arduino UNO” from the toolbox > Board’s menu (depending on the microcontroller unit (MCU) on your board). It comes pre-programmed with a boot loader. It connects through the STK500 protocol (reference, C header files). Many machines provide an additional layer of protection in the form of a fuse that provides internal guarding. Now, if more than 500mA is added into a USB connection, the fuse would immediately blow. Until the short or overload is eliminated, the link is left in place.

3.2 JSN SR04T Sensor

With a range of 20 cm–600 cm and a precision of 2 mm, the ultrasonic measurement module JSN-SR0T4-2.0 allows for quasi-distance measurement; the module contains a transceiver and control circuit for an integrated ultrasonic sensor. The JSN-SR04T-2.0 module from the division is used in mode one. Any MCU on the market employs an industrial grade optimized ultrasonic probe that is waterproof and produces a consistent yield. The module’s accuracy is consistent, and the distance calculation is precise. SRF05, SRF02, and other ultrasonic rangefinder modules from other countries are equivalent. Great level of accuracy module, blind (20 cm), and the device’s recommended limit are solid foundations for success on the market [3]. Following are the benefits of the sensor. Table 7.1 shows the datasheet of the ultrasonic sensor.

1. Compact size
2. Ease of use
3. Low-voltage power

Table 7.1 JSN-SR04t Sensor datasheet [2]

	Pulse width output/serial output
Output voltage	DC 3.0-5.5 V
Working current	Less than 8 mA
Probe frequency	40 KHz
The longest range	600 cm
The most recent range	20 cm
Accuracy in distance	(+-)1 cm
Resolution	1 mm
Angle measuring	75°
Enter the signal for the trigger	1, the serial port to deliver instructions 0x55 10[s above the TTL pulse 2
Echo signal at the output	Output pulse width level signal, or TTL
Wiring	3–5.5 V (power positive) Trig (Rx)Rx Echo (output)Tx GND (power supply negative)
Product dimensions	L42*W29*H12 mm
Temperature of activity	−20° to 70°
Color of the product	PCB board is used

4. Low power usage
5. Estimation with extreme accuracy
6. Effective anti-interference
7. Inbuilt closed waterproof cable probe
8. Perfect for muddy, difficult-to-measure conditions

Function Description:
MODE 1: Below is a description of the design [2].

The fundamental functioning principle:

i. TRIG activation range TO 10us high letter utilizing IO port.
ii. A signal to come back, the ECHO output out from IO port, a high stage, and a high time is the ultrasonic cycle from release to return time. Analyzing distance= (high time * sound speed (340M/s)) / 2.
iii. If you are unable to produce the echo (for example, whether the measurement range has been crossed or whether the probe is not on the object being tested).
iv. After 60 seconds, whether the measurement is right or otherwise, the ECHO port will fall, signaling the conclusion of the measurement.
v. As the module is illuminated, the LED indicator, also known as a non-power indicator, receives the light, indicating that the module is operational.

An individual transmits and reflects a brief ultrasonic pulse at time 0. This signal is received by the sensor, which transforms it into an electric signal, as shown in Fig. 7.1. As the echo fades, the next pulse will be broadcast. The cycle timeframe is the name given to this period. A minimum cycle time of 50 ms is recommended. The

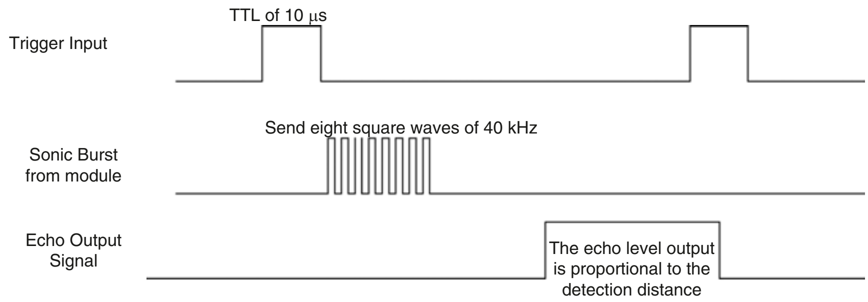


Fig. 7.1 Ultrasonic timing diagram

diameter of the echo pulse, which can be calculated using the method above, is equal to the predicted distance. If no impediments are discovered, the output pin will produce a high-level signal for 38 ms.

MODE 2: The pattern is described below:

The TTL level's serial output format for automated output distance loop the 100 MS module, the unit is in mm. Serial baud rate: 9600, n, 8,1. The range is performed every 100 ms inside the module when the module is turned on, and one frame is generated from the pin TX, comprising four 8-bit data. The format of the frame is shown by Eq. 7.1 [2]:

$$0XFF + H\,DATA + L\,DATA = SUM \tag{7.1}$$

1. 0XFF: used to determine which frame to use to begin the results.
2. H DATA: the upper 8 bits of the distance data
3. L DATA: the lower 8 bits of the distance data
4. SUM: data and for the effect of its 0XFF + H_DATA + L_DATA = SUM (only low 8)

MODE 3: When using the serial port:

When you toggle on the module, it goes into standby mode. Serial port baud rate: 9600, n, 8,1 TTL level serial output format. The module starts ranging and outputs one frame with four 8-bit data from pin TX when the RX port receives the 0X55 instruction. The frame format is shown in Eq. 7.2:

$$0XFF + H\,DATA + LDATA = SUM \tag{7.2}$$

1. 0XFF: used to determine which frame to use to begin the results
2. H DATA: the upper 8 bits of the distance data
3. L DATA: the lower 8 bits of the distance data
4. SUM: data and for the effect of its 0XFF + H_DATA + L_DATA = SUM (only low 8)

Description: If the data is not measured or the module is out of range, in the inactive region, the module returns the nearest distance value. If the data isn't measured or the module is out of limit, it returns 0.

The LED indicator, LED non-power indicator, will receive the 0X55 trigger signal once the module is lighted, showing that the module is functioning [2].

3.3 LCD Module

As a result of their minimal expense, wide accessibility, and programmer friendliness, LCD modules are regularly utilized in embedded projects. We have always seen these screens, whether on PCOs or calculators, in our everyday lives. Let us get a bit more technical now about the appearance and pinouts that we have already seen.

It is known as a 16 x 2 LCD because it has 16 columns and 2 rows. There are a variety of LCD sizes available, including 8 x 1, 8 x 2, 10 x 2, 16 x 1, and so on, but the 16 x 2 LCD is the most common. As a result, there will be a total of (16 x 2=32) 32 characters, each of which is made up of 5 x 8 pixel dots. The details and pin configuration are shown in Table 7.2.

3.4 I2C Model

I2C incorporates best SPI and UART characteristics. If we use I2C, a single master can link multiple slaves (like SPI), and also several masters may manage a slave or more. This is hugely advantageous when many microcontrollers are required to single-card log data or text projection solely on one LCD. Similar to UART communication, two wires for data transmission are included in I2C among devices:

The master and the slave use the line SDA for transmitting and receiving data, as shown in Fig. 7.2.

Serial Clock Thread (SCL): This is the line where the clock signal is stored.

3.5 How I2C Works

For data transmission, I2C employs messages. Data frames are used for dividing texts. Each message consists of a binary address framework and one or more frames of data containing the material needs to be sent. The message also contains start and stop terms, reading/writing bits, and ACK/NACK bits between each frame of the data. The SDA line must switch from high to low voltage before the SCL line can switch from high to low voltage. After the SCL line flips, the SDA line switches from low to high voltage which is the stop condition. When the master wishes to communicate with a slave, the address frame is a 7- or 10-bit sequence that uniquely

Table 7.2 LCD pin diagram

Item No.	Pin name	Description
1.	V_{SS} (ground)	System ground is attached to the ground pin.
2.	V_{dd} (+5 Volt)	+5V is used to power the LCD (4.7V-5.3V).
3.	VE (contrast V)	Determines the display's contrast ratio. To get the most comparison, it was grounded.
4.	Register select	Shifts between command and data registers when connected to a microcontroller.
5.	Read/write	Data can be written or read with this device. To write data to the LCD, it is normally grounded.
6.	Enable	For data acknowledgment, it was connected to a microcontroller pin and it toggled between 1 and 0.
7.	Data Pin 0 Data Pin 1 Data Pin 2 Data Pin 3 Data Pin 4 Data Pin 5 Data Pin 6 Data Pin 7	The 8-bit data line is formed by data pins 0–7. They will send 8-bit data while attached to a microcontroller. In this case, these LCDs can be used in 4-bit mode. Pins 4, 5, 6, and 7 will be left unconnected.
8.	LED positive	Backlight LED pin positive terminal.
9.	LED negative	Backlight LED pin negative terminal.

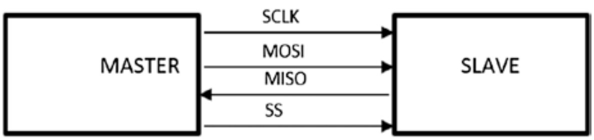


Fig. 7.2 I2C Master and Slave [4]

identifies the slave. Each frame in a post is followed by an acknowledge/no-acknowledge bit. When an address frame or data frame is successfully received, the receiving device sends an ACK bit to the sender.

Addressing: As I2C is slave-free, certain lines such as SPI necessitate a particular way of notifying the slave that sends data instead of another slave. While addressing, this process is done. The first frame is always the address frame after starting a new post. The address of the slave with whom it desires to speak is sent to any slave related to the master. Following that, each slave compares the master's address to its own. When the addresses correspond, the master will get a voltage drop in ACK bit. The slave does nothing until the addresses do not match, and the SDA line still seems to be active.

3.5.1 Steps of I2c Data Transmission

1. The master delivers a starting state to all linked slaves, flipping the high to reduced voltage SDA line prior to switching the high to low SCL line.
2. The master transmits a read/write bit to the slave you want to connect with a 7- or 10-bit address.
3. If the addresses are appropriate, the slave transmits a bit of ACK by holding down the SDA line for a bit. The slave sets the SDA line as huge when the address of the master doesn't match the slave's address.
4. The master provides the data framework or gets it.
5. The receiving device starts sending an ACK bit to the sender to recognize that the frame is received correctly.
6. Before switching SDA high to interrupt data transmission, the master conveys a slave's stop condition by switching SCL high.

4 Flowchart and Schematic Diagram

The working of the project is shown in Fig. 7.3. Due to the onboard load, there is a deflection in the leaf spring. We measure the distance from the chassis frame to the ground before and after the deflection is measured with the help of ultrasonic sensors on each corner. The data read by the ultrasonic sensors is then used to get the final load with the help of the principle of deflection of leaf spring which will be discussed later. The calculated load or the final load is displayed on the LCD.

4.1 Schematic Diagram

A schematic diagram of the proposed model is shown in Fig. 7.10

Connections of ultrasonic sensors with Arduino Uno are shown in Fig. 7.4.

Here:

1. The Rx pin of the ultrasonic sensor 1 is linked to pin 2 of the Arduino, while the Tx pin is linked to pin 3 of the Arduino.
2. The Rx pin of the ultrasonic sensor 2 is linked to pin 4 of the Arduino, while the Tx pin is linked to pin 5 of the Arduino.
3. The Rx pin of the ultrasonic sensor 3 is linked to pin 6 of the Arduino, while the Tx pin is linked to pin 7 of the Arduino.
4. The Rx pin of the ultrasonic sensor 4 is linked to pin 8 of the Arduino, while the Tx pin is linked to pin 9 of the Arduino.

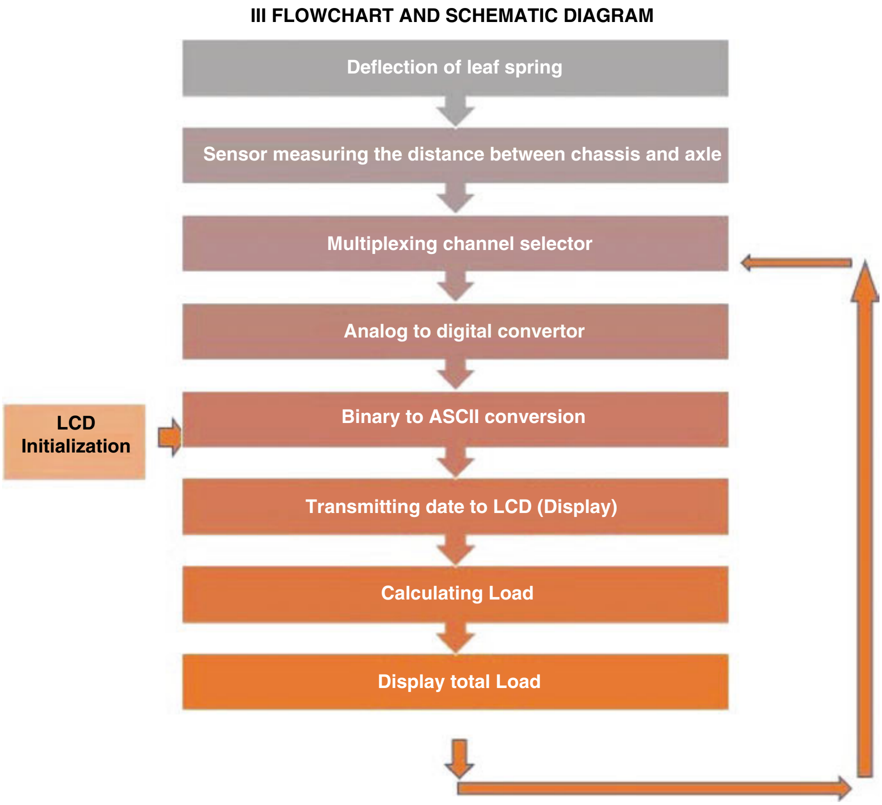


Fig. 7.3 Flowchart of the working of the proposed model

4.2 Block Diagram

Block diagram for the proposed model is shown in Fig. 7.5.

5 Serial Communication

There are many sorts of information that devices need to communicate among each other. These devices use special protocols designed for their communication. Different modules use these various protocols. One of such protocols called Universal Asynchronous Receiver and Transmitter is presented in this section.

Binary Number Systems Digital signals are used to transfer data across devices. Hence, the first thing that needs to be discussed is the form of the data that is being transmitted. A digital signal has rapid switching between high and low signals. The highs are represented by 1 and in unipolar signals low is represented by 0 and are

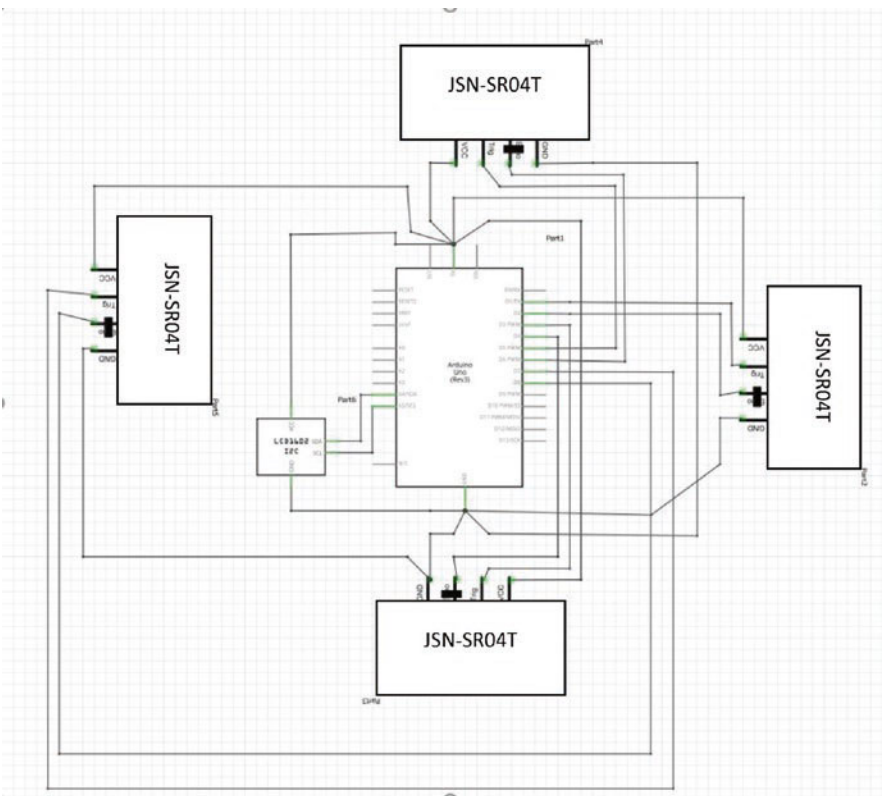


Fig. 7.4 Schematic diagram of the proposed model

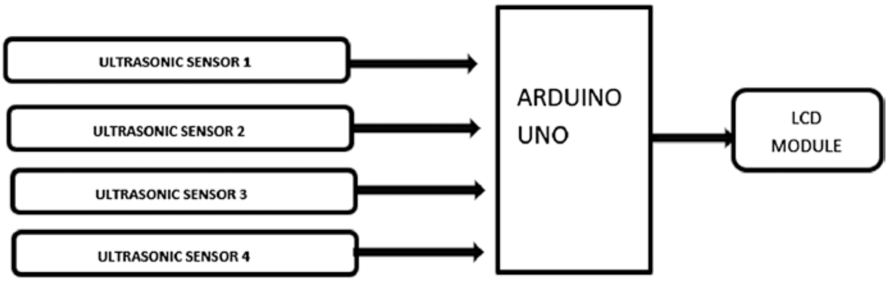


Fig. 7.5 Block diagram for the proposed model

called bits. A sequence of such 1s and 0s form data that a microcontroller can interpret; if a sequence consists of 8 bits, then it is called a byte. For example: 10111001 is a byte. Streams of bytes are transferred as data from one device to another.

Protocols of Communication For ensuring compatibility, there are only three main protocols of communication used widely across the electronics and electrical engineering community. This allows many devices to communicate freely with each other. Their protocols are namely UART, SPI, and I^2C. UART is the protocol used in the proposed model which is as follows:

5.1 UART

One of the protocols that transmit sequential bits and hence communicate serially is Universal Asynchronous Transmitter/Receiver. There is a requirement of one pin that transmits data and one pin that receives data, in the construction of a device that follows UART. This forms a serial port through which communication takes place. On the UNO board, there is a port dedicated to being used for communication through UART Protocol. This port consists of two pins GPIO 0 and GPIO 1 which can be connected to other devices for communicating. The software library for serial communication can be used while coding to access these ports. The asynchronous communication protocol does not transmit data depending on a clock. Rather it sends data and start and stop bits so that the other device can know when a stream of data begins and ends. Hence, the data is not dependent upon the clock settings of the sender and the receiver, so the scope of receiving and interpreting data as it was sent is higher.

6 Structured Approach

A structured approach for designing and implementing the proposed model is shown in Fig. 7.6.

7 Working Principle

The leaf spring is appended to the vehicle's hub. The entire vehicle load is on the leaf spring. With a basic pin association, the front finish of the spring is connected to the edge, while the back finish of the spring is connected with a shaft. The shackle is the flexible connection between the eye and the casing of the leaf spring [5, 6]. The components of the leaf spring are displayed in Fig. 7.7, where b is the width of the plate and L is the length of the plate or distance of the load W from the cantilever end.

Leaf springs are shaped like level plates (otherwise called level springs). The advantage of the leaf spring over the helical spring is that, notwithstanding the energy-retaining instrument, it is directed along a definite course. Notwithstanding

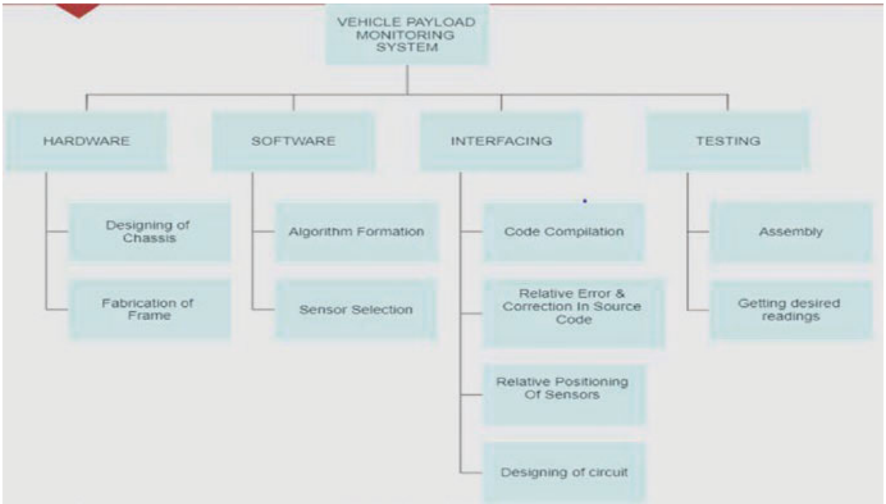


Fig. 7.6 Structured approach for designing and implementation

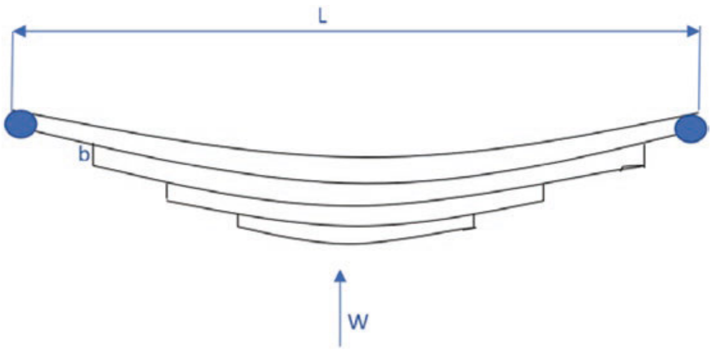


Fig. 7.7 Elements of leaf spring

shocks, the leaf springs could therefore deal with side burdens, slowing down force and driving force. In the proposed model, a solitary plate is joined to one end and stacked to the next. The plate might be utilized as a level wellspring [7]. The portrayal of the case outline before and after applying weight is displayed in Figs. 7.8 and 7.9, respectively. The avoidance brought about by the load is used to get wanted outcomes as given in Eq. (7.3).

$$\Delta(deflection) = \frac{3\left(WL^3\right)}{8\left(Enbt^3\right)} \tag{7.3}$$

where

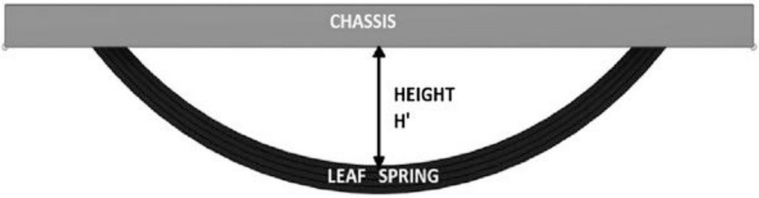


Fig. 7.8 Depiction of chassis frame before adding load

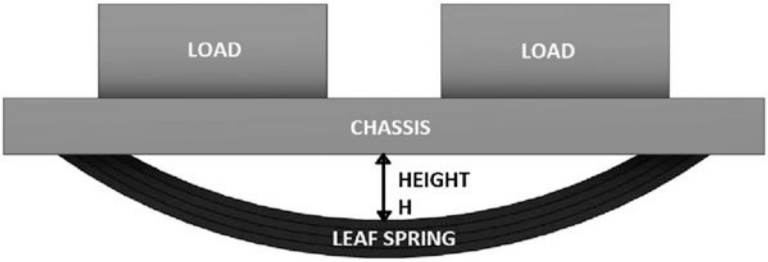


Fig. 7.9 Depiction of chassis frame after adding load

W=Maximum load on the spring
n=Number of leaves
L=Length of the spring
t=Thickness of the leaves
b=Width of the leaves
E = Modulus of elasticity

Equation (7.3) shows the maximum deflection by the leaf spring.

7.1 Conceptual Model

The design of the proposed model is shown in Fig. 7.10. The chassis design is built at the starting stage of the project to calibrate the desired results.

Figure 7.11 gives the overall placement of the ultrasonic sensors and the control system is shown.

8 Hardware Implementation of Proposed Model

This section presents the hardware implementation of the proposed model. The placement of the ultrasonic sensor under the frame is shown in Fig. 7.12. To design a chassis frame was part of the proposed model to test and calibrate the results. The chassis frame design is shown in Fig. 7.13.

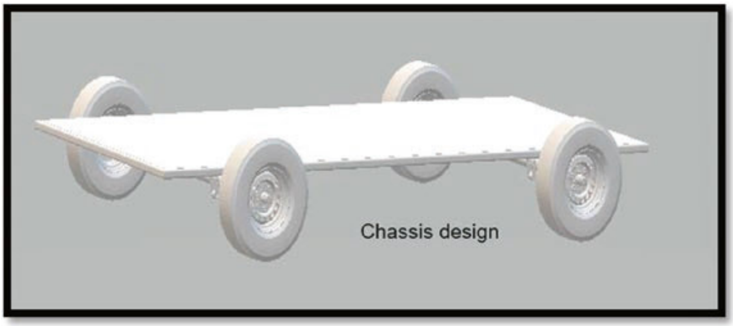


Fig. 7.10 Conceptual model of a chassis design

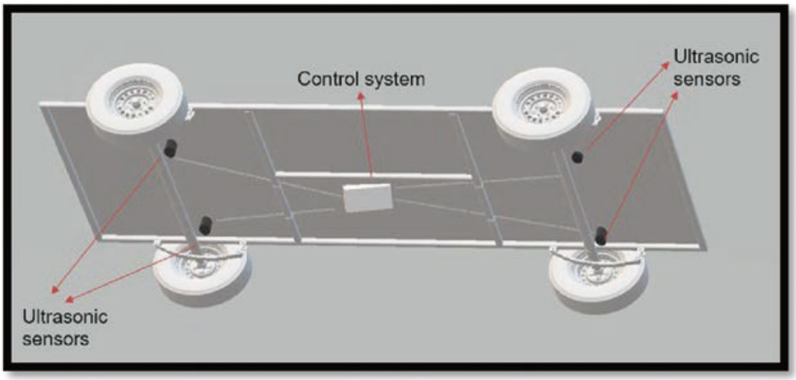


Fig. 7.11 Conceptual model of the sensor placement under the frame

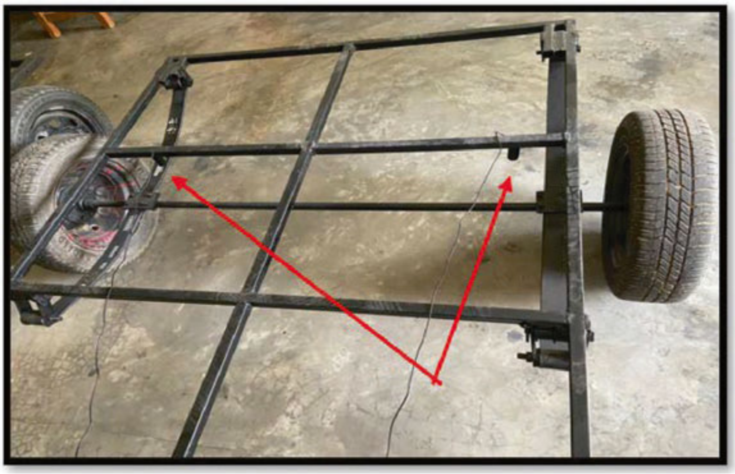


Fig. 7.12 Placement of the ultrasonic sensors

Fig. 7.13 Chassis frame design

Fig. 7.14 Final load displayed on the serial monitor

9 Results

The final load measured is displayed on the serial monitor of the Arduino IDE as shown in Fig. 7.14.

The final weight measured is also displayed on the LCD module as shown in Fig. 7.15.

10 Application

In this section, we discuss the wide range of applications of the vehicle payload monitoring system. The system can be used to detect unauthorized loading and pilferage, resulting in monthly savings of ₹8000–₹12000. It can also keep track of the amount of material handled and ensure that payload regulations are followed. The system can be used to shorten weighting times and avoid long lines. It also aids in monitoring unloading time at distribution points. But apart from this usage, there can be numerous applications of this system. This includes in the area of design, maintenance, and research on pavements; design, monitoring, and research for bridges; enforcement of size and weight; legislation and governance; planning and administration; monitoring of heavy vehicle live loads; dump truck management; garbage truck management; monitoring heavy vehicle load restrictions for highway, etc.

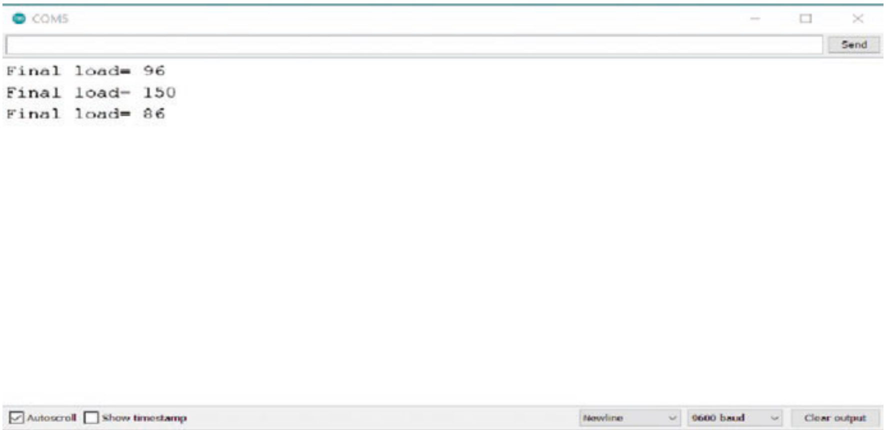


Fig. 7.15 Final weight displayed on LCD module

11 Future Work

Load balancing Helping the driver to place the load balanced in the truck, i.e., whether the load is equally balanced in the truck on both sides or needs to be shifted towards the driver side or the rear end of the trolley.

GSM enabled The system would provide the owner with the information about the weight of the truck through the number plate of the truck on an application continuously when the shipping is being done.

Optional security Providing the company with an optional security measure wherein if the load is more than the reference load the tire of the truck will stop moving and will rotate only when the load is less than the reference weight or is within the safety parameter [8].

12 Conclusion

This chapter proposed the hardware development of the payload monitoring system which will be helpful in resolving the problem of a vehicle's inability to reach weighing stations to weigh the cargo of its cars owing to long lineups. A prototype hardware model is developed after successfully testing the simulated model. The prototype worked exactly as a real-time system during demonstration. The proposed model measures the weight of any vehicle based on the deflection of leaf spring wherein the sensors measure the distance between chassis and the axle and displays on the LCD screen of the driver driving the vehicle after due process of conversions. The prototype as presented is not existing in the current heavy vehicles in India and has innovation of its own. It has numerous applications both for the driver or the owner of the vehicle and the government in implementing various regulations. The proposed system saves time and money to weight the load of the vehicle which is required during transportation and load balancing thereby saving many accidents. The application of the proposed system can be extended further while designing bridges, flyovers, etc. wherein the measurement of weight is an important component to maintain the life of the system in which it is implemented.

Appendix

Arduino UNO board datasheet [9]:

- ATmega328 microcontroller
- The operating voltage is 5 volts
- Input voltage: 7–12V (recommended)
- 6–20V input voltage (limits)
- 14 Digital I/O pins (of which 6 provide PWM output)
- Pins for Analog Input 6
- 16 MHz clock speed
- V_{in}: The Arduino board's input voltage when utilizing an external power source. If this pin is provided through a power socket, it can deliver voltage or access voltage
- 5V: The board regulator provides a regulated 5V pin. The DC Power Jack, USB, or Vin pin can deliver the power supply to the board (7–12V). The 5V or 3.3V voltage pins provide the controller to bypass which might destroy your board.
- 3.3V: The on-board control device produces a supply of 3.3 volts. The maximum drawing current is 50 billion
- GND: Ground pins
- Input and output
    - 0 (Rx) and 1 (Tx) in serial TxTTL receives serial data (Rx) and transmits them (Tx) by this device. These pins are connected to the equivalent pins of the ATmega328 USB to TTL chip. External interruptions are two and three. These pins may be set to interrupt with a low value, a rising or decreasing lip, or a change in value. An 8-bit PWM output is detected via analog write ().
    - 10 (SS), 11 (MOSI), 12 (MISO), 13 (SPI) (SCK): To connect to these pins, the SPI library is utilized. 13 of LEDs in an integrated LED are connected by a cable to the digital pin 13. The LED is activated when the pin is upward, and the pin is downward.

JSN SR04T Sensor

1. Product features:

The JSN-SR04T ultrasonic distance measuring module offers noncontact distance monitoring with a precision of up to 3mm from 25 to 450 cm. The kit includes an ultrasonic transmitter, receiver, and control unit. It functions similarly to JAN's HC-SR04 module.

2. Basic working principle:

- The IO port TRIG, which emits a high-level signal for at least 10 seconds, triggers the ranging.
- The module generates eight square waves at a frequency of 40 kHz and detects whether a signal is received or not.
- When a pulse is returned, the IO port ECHO outputs a high-level signal.

3. Pin assignmen:

Using VCC for a 5V supply. GND is the ground cable. TRIG is the trigger activation signal feedback. The echo signal's output is known as ECHO.

4. Electrical parameters:

- Working voltage: DC 5 Volt [10]
- Working current: 40 mA
- Acoustic emission frequency: 40 KHz
- Longest distance: 4.5 m
- Shortest distance: 25 cm
- Measuring angle: 30 degrees
- Input trigger signal: 10⌈S TTL pulse
- Output echo signal: output TTL signal, proportional to range
- Size: 41*29 mm
- Probe lead length: 2.5 m [1]

5. Ultrasonic timing diagram

The above sequence diagram (Fig. 7.A-1) demonstrates that all you need is a pulse trigger signal that lasts longer than 10 seconds, and the module can send out eight 40 kHz cycles and sense the echo. When an echo signal is observed, an echo output signal is received. The pulse diameter of the echo signal is the same as the measured wavelength. The time differential between the transmitted and received signals can be used to calculate the distance.

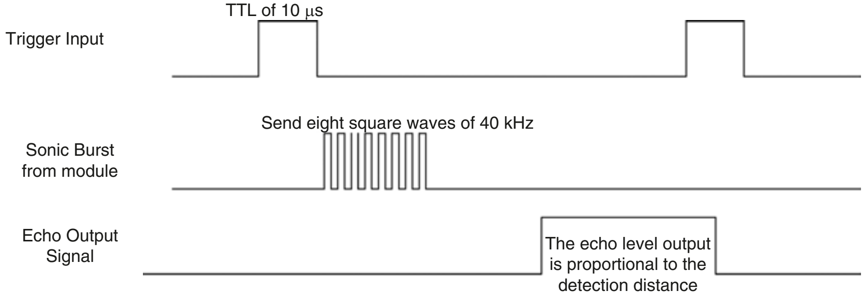


Fig. 7.A-1 Ultrasonic timing diagram

References

1. Andang, A., Hiron, N., Chobir, A., & Busaeri, N. (2019). Investigation of ultrasonic sensor type JSN-SRT04 performance as flood elevation detection. In *IOP Conf. Series: Materials Science and Engineering 550 012018*. IOP Publishing. https://doi.org/10.1088/1757-899X/550/1/012018
2. JahanKit Electronic, “JSN-SR04T-2.0. Datasheet” [Online]. 2019 [cit. 2020-07-02]. Retrieved from: https://www.makerguides.com/wp-content/uploads/2019/02/JSN-SR04T-Datasheet.pdf
3. Ramesh, P., Sudheera, S., & Reddy, D. V. (2021). Distance measurement using ultrasonic sensor and Arduino. *Journal of Advanced Research in Technology and Management Sciences (JARTMS), 3*(2) ISSN: 2582-3078, March–April 2021.
4. Ghanekar, A., Kishor, B., & Bandewar, S. (May 2016). Design and implementation of SPI bus protocol. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 5*(5).
5. Gowd, G. H., & Goud, E. V. (August 2012). Static analysis of leaf spring. *International Journal of Engineering Science and Technology (IJEST)*.
6. Teli, M. D., Chavan, U. S., & Phakatkar, H. G. (July 2019). Design, analysis and experimental testing of composite leaf spring for application in electric vehicle. *International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8*(9).
7. Clive, M. A. (June 2018). Design and analysis of composite leaf spring. *International Research Journal of Engineering and Technology (IRJET), 5*(6).
8. Odonkor, E. N., & Ofosu, W. K. (Jan 2020). Design and construction of vehicle loading monitoring system using load sensor and GSM. *International Journal of Applied Science and Technology, 10*(1).
9. Louis, L. (July 2018). Working principle of Arduino and using it as a tool for study and research. *International Journal of Control, Automation, Communication and Systems (IJCACS)*. https://doi.org/10.5121/ijcacs.2016.1203
10. Hassan, M. A. H. (2016). *Design and real time control of a versatile scansorial robot*. PhD thesis, University of Sheffield.

Chapter 8
Implementation and Comparison of MQTT, WebSocket, and HTTP Protocols for Smart Room IoT Application in Node-RED

Simran Kaur and Vandana Khanna

Abstract Internet of Things portrays a general notion for capability of devices to sense and collect data and share it via the Internet, where it can be refined and applied in numerous applications. This chapter gives an overview of IoT architecture and different components of IoT, along with some applications. IoT-based smart room application has been implemented and presented in this work in detail. The hardware that has been used for smart room application is a NodeMCU controller board and some sensors such as proximity sensor, gas sensor, flame sensor, and temperature and humidity sensor. Smart room application has been implemented with the help of IoT protocols like MQTT, HTTP, and WebSocket. All the hardware components and protocols used in the implementation of the application have been discussed in the chapter. Also, the theoretical comparison of three protocols—MQTT, WebSocket, and HTT—has been drawn. For implementation of design using IoT, Node-RED is used. Node-RED is an open-source software used to wire together all the hardware devices and APIs. The response time delays of all these protocols for the smart room test bed have been found and compared. Also, the present work has been compared with some of the related existing work in literature.

Keywords IoT protocols · Smart room · Node-RED · Sensors

S. Kaur · V. Khanna (✉)
Department of EECE, The NorthCap University, Gurugram, India
e-mail: simran19ecp002@ncuindia.edu; vandanakhanna@ncuindia.edu

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_8

1 Introduction

IoT stands for Internet of things. IoT is a giant network of devices that share data among themselves and process them in a manner that can be put to a practical use in a number of applications [1]. In today's world, where overpopulation is the biggest problem that mankind faces, IoT has capabilities to face the challenges of urbanization and make lives more secure and comfortable. In fact, many researchers concluded that IoT-driven cities provide solutions to problems from traffic management to waste management and from energy-efficient infrastructure to healthcare.

Before IoT, machines required to be operated manually, but with IoT all the devices can connect and share experiences among themselves, hence scaling down the need for manual labor in a machine cycle. All the devices can be pre-instructed. These devices can even adapt to the needs and requirements of the user and modify their operation accordingly. The basic examples where IoT influences our life on a daily basis are air conditioners that we can operate and control with our smartphones, smart cars providing the shortest route, smart watches that continuously track health and daily exercises etc. IoT is a system of connected devices that collect data about how they are used in the surroundings in which they are being operated. It is all achieved by using sensors. Nowadays, sensors are installed in every physical device. Sensors continuously emit and process the data about the working state of devices. IoT provides all the devices a common platform to share their data with each other. Data is transmitted from various sensors and is sent to the IoT platform securely. IoT platform combines the collected data from all the sensors or sources and further analysis is done and hence valuable conclusions are derived. These conclusions are shared and reviewed as per the requirements of the applications for a better user experience [2].

To take an example, in a refrigerator manufacturing company, both the manufacturing machine and the belt have various sensors connected to it. The sensors regularly send data regarding the machines' health and specific information about production to the manufacturer to identify any issues beforehand. A barcode is installed on each product before moving on from the manufacturing belt to the next step. This barcode contains information about manufacturing details, product code, special instructions, etc. This data is used to find where the product was dispersed and track retailer's inventory. Through the barcode attached to each product retailers can also know from which manufacturer the product is coming, check their inventory special instructions, and many more. The compressor of the refrigerator has another sensor attached that emits data regarding its health and temperature. This data can be analyzed allowing the customer services to contact the owner for repairs and timely services, if required. Similarly, there are numerous applications from smart air conditioners to smart homes, smart cars, and smart cities where IoT is reformulating the way we interact with the surroundings around us and technology. Future of it is limitless. Business insider intelligence estimated that 24 billion IoT devices will be present by the beginning of 2021 following with a lot of job opportunities in the upcoming future [3].

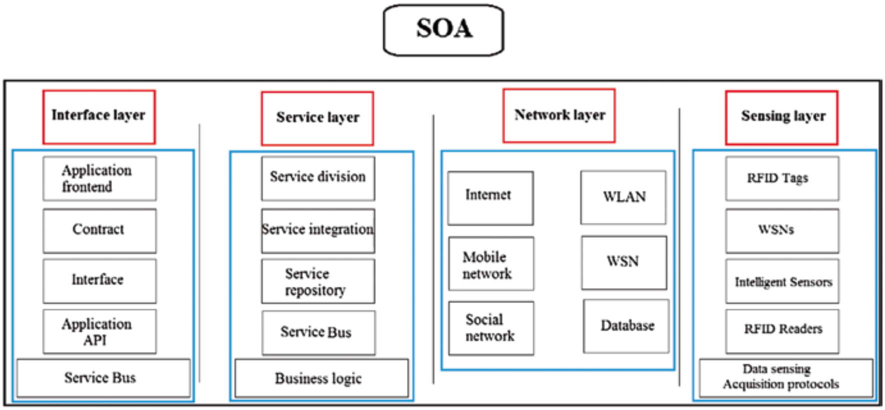


Fig. 8.1 IoT architecture

2 Architecture of IoT

Figure 8.1 shows the service-oriented architecture of Internet of Things. Service-oriented architecture or SOA means software components reusable through service interfaces. The interfaces can be used in common communication protocols in a way that they can be integrated with the new applications without performing deep integration every time [3].

There are four different layers in IoT architecture:

1. Sensing layer: Here data sensed from smart sensors, RFID tags, and RFID readers is collected and sent to the higher hierarchical layer, i.e., network layer.
2. Network layer: This layer serves and deals with all the network-related issues and tasks.
3. Service layer: This layer has service repositories, integrations, service bus, and service divisions.
4. Interface layer: Interface layer is the last layer in this architecture, which is in charge of all the applications' APIs, applications' frontend, interfaces, etc.

3 Components of IoT

There are some key components of IoT, which can also be called as enabling technologies for IoT, as without these components/technologies, IoT would not have been possible. A technology that helps in collecting/sensing data, analyzing or processing them, and taking the required action falls under the category of enabling technologies. Some common such technologies used in the Internet of things are sensors, actuators, embedded systems (computing boards), cloud storage, protocols, etc. Some of these technologies or key components of IoT are discussed below:

(i) **Device**: The things in "Internet of Things" are objects that are being monitored and have some sensors embedded to it for continuous data collection. A sensor is a device that detects any changes in the present conditions and forwards this information in a useful manner. For example, heat is converted into electrical signals in a temperature sensor. Some common sensors are pressure sensor, ultrasonic distance sensor, accelerometer, camera sensor, PIR motion sensor, temperature sensor, etc. [3].

(ii) **Embedded boards**: This is a very important component that converts IoT designs into a working system. The whole architecture of IoT revolves around these boards. These boards are equipped with practical computer components like microprocessor, ROM for OS storage, input and output ports, and memory, which can provide computing solutions with a low-profile architecture and significantly less power consumption. Due to their comparatively smaller size, they can be used in mobile and ubiquitous applications. Some common boards that are used in IoT are Raspberry Pi, Arduino Mega, Arduino Nano, Arduino Uno, Arduino Lilypad, Arduino Due, NodeMCU, etc. All these boards are easily available, online and off-line, at very affordable prices [4].

(iii) **Local network**: After reading the changes in ambient conditions through sensors, embedded boards forward this data in a certain manner through a gateway. Local network includes this gateway, which transcribes it for the internet protocol. The choice of a local network can significantly affect its cost and hardware requirements. As IoT applications are endless, there is not one right answer that will be applicable for all. The right choice can improve storage capacity, enhance resilience, and reduce overall cost of the application significantly [3].

Types of local networks are:

- Personal area network (PAN)
- Storage area network (SAN)
- Local area network (LAN)
- Virtual private network (VPN)
- Wide area network (WAN)
- Campus area network (CAN)

(iv) **Cloud**: IoT cloud is a refined network of servers developed to perform data processing for multiple devices and deliver accurate data analysis. To manage the data derived from devices and applications, IoT cloud is used. Cloud has tools to collect, process, store, and manage massive amounts of real-time data. The main advantage is that this data can be accessed remotely whenever required and take actions remotely. Cloud services are classified into three categories:

(a) Infrastructure-as-a-Service (IaaS): This is a form of cloud computing that provides virtualized computing resources over the Internet. The user manages the machine, selects the OS and application, and pays as per use.

Some common examples are Cisco Metapod, Microsoft Azure, and Google Compute Engine (GCE).

(b) Platform-as-a-Service (PaaS): This cloud computing model delivers software and hardware tools needed to use over the Internet. A PaaS provider hosts all these according to its own infrastructure; users have to build and maintain the application as per their requirements. Some common examples are Windows Azure, Heroku, Force.com, and OpenShift.

(c) Software-as-a-Service (SaaS): This cloud computing model provides a complete software application to the user. Another name for this model is application as service. Users can pay to this service monthly, yearly as a subscription fee to the SaaS provider. Some examples are Amazon web services, Adafruit, AzureIoT, Cisco WebEx, and Google Workspace [5].

The most common example of a cloud that is used to visualize the data is Thingsboard. Thingsboard is an open-source platform that allows users to control and monitor IoT devices remotely. Thingsboard is designed to be durable, fault-tolerant, scalable, and customizable with easy-to-add widgets and rule engine nodes. Thingsboard provides pay-as-you-go subscription plans [6]. Thingsboard offers MQTT, CoAP, and HTTP protocols to connect to your IoT device. All the devices can be connected via these protocols using Thingsboard gateway. With this cloud service, you can:

– Control all the IoT devices remotely and with maximum security
– Collect and easily visualize the data from IoT devices
– Accomplish complex processing with triggers and incoming alerts
– Move and push device data to other systems for further processing
– Build workflows based on a device life cycle
– Design dashboards that show real-time data being collected by your device and provide insights to your customers

(v) **User interface**: It is an interface that allows the user to interact and check on the system whenever required for example, if the user wants to check the temperature or any other parameter via a web browser or a smartphone app [1]. The user may also be able to perform certain actions that affect the system. Also, depending on the applications, it may take automatic actions based on some predefined set of rules. Another option can be to control any action through the app instead of controlling the device manually. For instance, rather than changing the temperature of the air conditioner manually, it can be changed through the mobile or web app remotely, to maintain certain temperature conditions throughout the day.

(vi) **IoT protocols**: According to connectivity, IoT protocols are characterized into 2 categories: network protocols such as HTTP, LoRaWan, Bluetooth, and Zigbee and data protocols such as MQTT, CoAP, M2M, AMQP, and XMPP. However, according to their functionality, these protocols are categorized into three:

(a) Based on connectivity: examples include 6LowPAN and RPL

(b) Based on communication: examples are Wi-Fi and Bluetooth
(c) Data protocols such as MQTT, CoAP, AMQP, and WebSocket

4 IoT Applications

IoT has an endless number of applications from smart wearable to smart cars, traffic management to smart grid, and smart cities. Some of the applications are discussed below [2]:

Everyday Life This is the first application to deploy IoT services. There are numerous applications where IoT can serve us on a daily basis. Examples of how IoT can be useful in everyday life can be such as the temperature of the room can be kept comfortable even if the outside temperature is varying. Air conditioners can be connected to a cloud service with a dashboard that has all the required relevant information like outside temperature and a preferred temperature for the person. The application can turn the air conditioner ON/OFF as per the observed temperature conditions inside the room. This is a very common example. IoT sensors are embedded in every device nowadays, from fit bits to smartphones, from home appliances to vehicles and to smart cities.

Healthcare Some of the major challenges faced today in general practice of healthcare are lack of real-time data tracking, lack of smart devices, and lack of standard analytics. To help with all these issues, IoT opens ways to connect with the ocean of valuable data through analysis and real time testing. One very common example is wearable technology in the form of fit bits that keeps track of a patient's heart rate, body temperature, etc. If the device will read any peak abnormalities in a patient's health, it can contact the listed emergency contact and the doctors and send them the patient's live location as well. Hence, quick action can be taken before it's too late. This application of IoT will help in decreasing mortality rate in case of heart attack patients where immediate medical care is required. It will also help the patients who are unable to speak for themselves, like kids, the elderly, and the disabled. Along with these advantages, it will also help medical professionals to keep track of the previous medical records of the patient. IoT not only improves the quality of healthcare but also empowers healthcare professionals. The main advantage of remote healthcare is that we can connect with healthcare professionals and doctors immediately. It removes queues, cuts waiting times with doctors, and reduces cost of healthcare overall.

Smart City Smart cities are a very dynamic subject in the field of IoT applications as problems are different in each city of the world. For instance, Delhi faces pollution problems each year whereas Mumbai has flood problems each year. The problems like waste control, housing, pollution, and traffic control impact different cities with different intensities. Let's consider traffic cameras as an IoT solution to traffic control. A traffic camera is installed at every corner of the city which is connected with a common gateway for the whole city. Through the gateway, all this data is stored in the cloud. This helps monitor traffic jams at peak hours, can determine accident prone areas, road blockages etc. Hence, it can be used for city-wide monitoring. So, all this data can be analyzed and processed, and hence, traffic can be easily rerouted in case of an accident or road repair works and real-time information can be sent to drivers via radio channels or GPS to their smartphones.

Agriculture Agriculture is the most neglected area in IoT applications, despite the importance it holds. There are some major issues like inaccuracies, irregularities due to physical monitoring, labor costs, and other human errors. To solve these issues, IoT proposes different solutions like smart irrigation system, smart greenhouse, and precision farming. If we take the example of smart irrigation, different sensors can be implanted in the soil which gives real-time information about the temperature, humidity, moisture content, and weather conditions via a network gateway, which can be accessed on any device. A relay can be connected to the water motors at the other end, so when the sensed values drop from the predefined values, the water motor will accordingly turn on/off automatically. Hence, this method can save tons of water. It is precise and efficient and cuts the labor cost and power consumption significantly.

Industrial Automation IoT applications in industrial automation improve the line of command immensely and optimize time significantly. Also, it cuts training staff costs. With IoT, remote operations can be performed with ease. Hence, it is easy to monitor the supply chain while keeping an eye on the inventory.

Disaster Management IoT cannot stop disasters from occurring, but it can help build resilience and help in preparedness. According to researchers, 95% of the deaths around the globe are in developing countries due to drastically increasing population, poor infrastructure, and severe weather conditions. IoT can help rescue these vulnerable countries exposed to disasters in different ways through prediction, preparedness, response, and recovery. The example of managing forest fire disaster can be that sensors can be installed at places inside the forest. These sensors can send data feeds about the temperature, humidity, heat index, and carbon emission to a control room nearby via a secured communication gateway. As this data is all in real time, if any emergency occurs, fire marshals and local police can be alarmed immediately.

5 IoT-Based Smart Room Application

The test bed created here for the comparison of the IoT protocols includes some sensors like DHT11 temperature and humidity sensor, PIR motion sensor, flame sensor, and gas detector sensor interfaced with a single NodeMCU Board to realize a smart room application via three different protocols, i.e., HTTP, MQTT, and WebSocket. It is necessary to identify which protocol is best suited and best choice for a certain application. Picking the best option for a certain application can lead to minimizing the risk and security threats. Also, it is cost effective, saves power consumption, and increases the efficiency of the application.

The three protocols implemented in this work have been discussed in detail in Sect. 5.1 and their theoretical comparison has been drawn. Later, after implementation of smart room application using all three protocols, the response time of these protocols has been compared. To implement these protocols, Node-RED software is used. Node-RED operates on module-based architecture where pre-constructed modules are connected to make a flow and perform operations. So, three different flows were created in the Node-RED editor and deployed. Further, the data collected from these sensors is being displayed on the Node-RED dashboard. The Node-RED software has been discussed in Sect. 5.2.

5.1 MQTT, HTTP, and WebSocket Protocols

5.1.1 MQTT Protocol

MQTT stands for Message Queue Telemetry Transport. It is an ISO standard protocol which was introduced by IBM in the late 1990s, and later standardized by OASIS in 2013. MQTT is a lightweight messaging protocol based on Publish/Subscribe system architecture. It is basically designed to provide connectivity between middleware and application on one side and communication and network on the other side. Main applications of MQTT are remote areas with limited bandwidth and small-code footprint. Figure 8.2 shows the various components of MQTT protocol.

Basic components of MQTT are:

(i) *Publisher*: Publisher is generally a lightweight sensor with the main topic that subscriber is usually interested in.
(ii) *Subscriber*: Subscribers are the clients, i.e., applications that are interested in the sensor data being published.
(iii) *Broker*: Broker in MQTT is a central communication point, which is in-charge of handling all the messages between senders (publishers) and receivers (subscribers).

Another key component in this protocol is the topic. Topic is the subject to which the client is subscribed, is updated like messages, and is often distributed by the message broker. It is a simple string separated by "/" that can have more than 1 hierarchy level [3]. For example, the topic for sending the room temperature is house/living-room/temperature. Similarly, the topic for subscribing to the

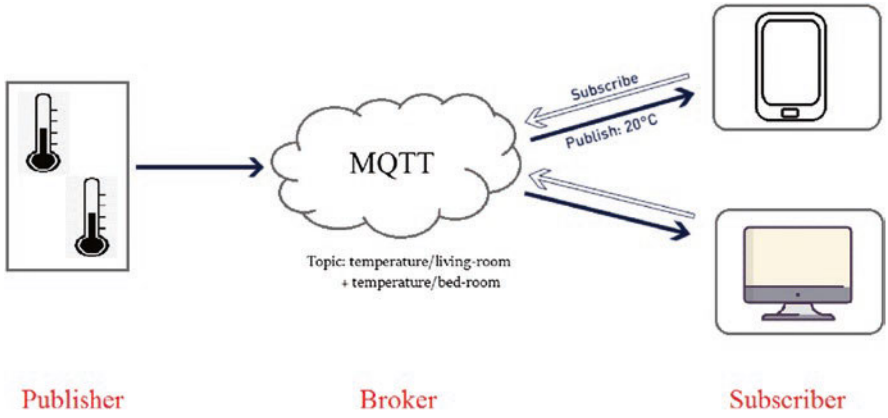


Fig. 8.2 Components of MQTT protocol

temperature of the kitchen can be house/kitchen/temperature. The client can subscribe to each topic individually or use a wildcard also. The topic "house/+/temperature" can be subscribed to, and it will have all of the previously subscribed topics, with a random value in place of the living room or the kitchen. The "+" sign allows arbitrary values for one hierarchy, hence called a single-level wild card. If more than one level is needed to subscribe to, "#" is used. For example, house/# will have all the topics beginning with house, like house/room/temperature, house/kitchen/light, and house/living-room/lights.

There are various MQTT methods which are listed as follows:

- Connect: waits to connect with the server
- Disconnect: waits for the TCP/IP session to end as soon as the client is done with the work
- Subscribe: sends a request to the server to let the client subscribe to topic(s)
- Unsubscribe: sends request to server to let client unsubscribe to topic(s)
- Publish: returns to application after passing request to MQTT client

Applications of MQTT [8]:

- Facebook Messenger uses MQTT protocol for online chat applications.
- The EVRYTHNG IoT platform uses MQTT as an M2M protocol for millions of connected products.
- Microsoft Azure IoT Hub uses MQTT as its main protocol for telemetry messages.
- Adafruit launched a free MQTT cloud service for IoT experimenters called Adafruit IO.
- Amazon Web Services uses Amazon IoT with MQTT.

5.1.2 HTTP [9]

HTTP stands for Hypertext Transfer Protocol. Since 1990, HTTP has been the foundation for communication for World Wide Web (WWW). It is a standardized way for computers to communicate among themselves. This stateless Internet protocol can be used for different purposes and extensions for its headers, request methods, and error codes. HTTP is a TCP/IP-based communication protocol, which is why this protocol is mostly used to deliver data on the Web.

Figure 8.3 shows the HTTP architecture; HTTP is a request/response model with client-server architecture where the user/client requests for information to a server via a web browser and then the server replies back with the requested data.

Web Client: Through a TCP/IP connection (HTTP) client sends a request in the form of a URL to a certain server, which contains a MIME-like message that consists of client information and a request modifier.

Web Server: Server accepts the request sent by the client, processes, and launches back a reply with success or error codes as well as server information, metadata, and the requested information.

For an HTTP request, following steps are required:

1. A link to the HTTP server is opened.
2. A request for information is sent.
3. Processing happens.
4. Once processing completes, a response is launched from the server.
5. Finally, connection is closed after sending the response.

Important HTTP commands are as follows:

- GET: Client request to access a web page from server
- PUT: Client request to store a web page on server
- POST: Client request to update some content of a certain web page
- DELETE: Client request to delete a web page from the server

Following are the features of HTTP:

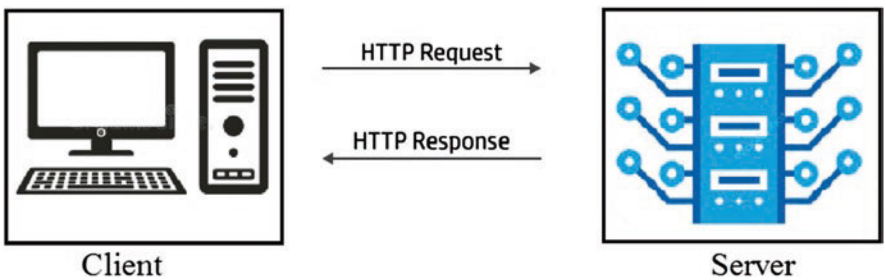


Fig. 8.3 HTTP architecture

- *Stateless*: When 2 consecutive requests are being executed on the same connection, as there is no connection among them, HTTP is referred to as stateless. HTTP cookies can be integrated into the workflow through the concept of header extensibility. Hence, sessions can be created on each HTTP request for sharing the same concept.
- *Simple*: HTTP is designed to be human-readable and plain. It does not enclose messages into frames.
- *Connectionless*: HTTP request is proposed by the client (browser) on the user's request for a message. The request is then processed by the server and a response is given to the client.
- *Extensible*: HTTP can be customized by adding a set of new rules agreed by client as well as a server.

5.1.3 WebSocket

Like HTTP, WebSocket is a client-server-based communication protocol, but the difference is that unlike HTTP, WebSocket is bidirectional and starts with wss:// or ws://. In WebSocket, the connection between client and the server is kept open until it is closed by either the client or the server; that is, unlike HTTP, WebSocket is a stateful protocol [3]. WebSocket provides a way for browser-based applications in which two-way communication is required with servers. This protocol consists of a handshake mechanism layered over TCP followed by basic message framing.

Figure 8.4 shows the handshake signals between a client and server during a WebSocket connection. During a handshaking mechanism, the client or web browser and a server start a communication between them, and it is kept alive until either of them decides to end it. When connection is established and kept alive, the data flow takes place until the handshaking is terminated. After handshaking, a new connection is opened and kept alive. This new connection is known as WebSocket. Once connection is confirmed and opened, data exchange will take place from both directions until it is closed by one of the parties. If either one of them closes the connection, then it will automatically close for the other one. Hence, status 101 stands for the switching protocol in WebSocket.

Applications of WebSocket protocol are mentioned as follows [8]:

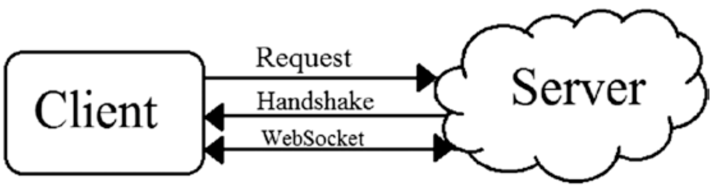


Fig. 8.4 WebSocket connection

- Real-time Web application: Almost all real-time Web applications use WebSocket to send information at client end, which is frequently sent by back-end server. Data flow is continuous into the same channel until it's kept open; that's why WebSocket is quick and improves application efficiency. For example: All the trading websites like Bitcoin use WebSocket to display the fluctuations in prices as data continuously needs to be updated by the back-end servers.
- Chat application: To create a connection to exchange, broadcast, and publish the message among clients, chat applications use WebSocket as it uses the same connection for one-to-one information transfer.
- Gaming applications: In games, data needs to be sent continuously without stimulating the UI screen over and over again. WebSocket is used as the UI gets automatically refreshed without any need for a new connection.

5.1.4 Theoretical Comparison Between IoT Protocols

Table 8.1 shows the theoretical comparison between three protocols discussed above, namely MQTT, HTTP, and WebSocket.

Following conclusions can be derived from the above theoretical comparison:

1. MQTT has minimum overhead during communication as compared to other protocols when multiple requests are sent.

Table 8.1 Comparison between three IoT protocols

S. No.	Parameters	MQTT	HTTP	WebSocket
1.	**Priority settings**	Priorities can be set	No features as such	No features as such
2.	**Overhead communications**	Minimum overhead during communication	Lots of overhead when handling multiple data requests	Lot of overhead when many IoT devices communicate
3.	**QoS options**	Yes	No	No
4.	**Transport**	TLS	TLS	SSL
5.	**Architecture**	MQTT uses publish/ subscribe	WebSocket has both publish/ subscribe as well as client/server	HTTP uses client/ server architecture
6.	**No. of devices**	Multiple devices can subscribe and publish messages	No option for data reception from the server; the client needs to request the data all the time	Designed for point-to-point communication
7.	**Applications**	Specially designed for reducing bandwidth and assurance of delivery	It works on the request/response model. It is more complex	Mainly developed for full-duplex/ bidirectional communication channel, example: Streaming

2. MQTT also has QoS options: 0 (at most once), 1 (at least once), 2 (exactly once) that defines the guarantee of delivery for a certain message.
3. WebSocket and HTTP are mainly designed for point-to-point communication while MQTT can be used to communicate one-to-many and send information to multiple clients.
4. While HTTP and WebSocket are developed full-duplex/bidirectional communication channels for live streaming, MQTT is mainly designed for assurance of delivery at a lower bandwidth.

For further comparison between other protocols like CoAP, XMPP, AMQP, and DDS, various research papers are available by different authors. For instance, Meera M S [10] compared and reviewed packet loss by creating a test bed for a marine IoT application scenario with data from various sensors using AMQP, CoAP, and MQTT protocols. Stefan Mijovic [11] studied CoAP, WebSocket, and MQTT by implementing them on low-cost hardware platforms.

5.2 Node-RED [12]

Node-RED is an open-source programming tool created by IBM technologies, used to wire together APIs, hardware devices, and online services. Primarily, it is a visual programming tool for IoT, which is also used to shape various flows of different services. Node-RED allows end users to work together with web services as well as hardware by removing low-level coding tasks. It can be easily done by visual drag-drop interface. Different components present in Node-RED create a flow and therefore, coding required is completed automatically. Node-RED is available in the Bluemix IoT starter application. It can also be deployed as Node.js separately. Node-RED operates on module-based architecture where pre-constructed modules are connected to make a flow and perform operations. Node-RED has a browser-based editor which makes it easier to wire together different nodes already downloaded in the palette and deploy them with a single click.

5.2.1 Node-RED Concepts

There are some common definitions that are used in Node-RED:

1. *Node*: A Node in Node-RED is the basic building block of the flow. They are triggered by waiting for some external stimuli like a timer, GPIO Hardware change, or an HTTP request or receiving some messages from previous nodes in a flow. They process and send the appropriate reply to continue the function of the flow. Some common nodes used are Inject, Debug, Switch, Function, and Change (shown in Fig. 8.5).

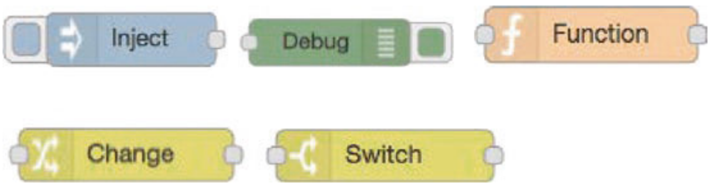


Fig. 8.5 Some common nodes in Node-RED

(a) Inject: Inject node is used to manually trigger the flow by clicking the button on the left. It is sometimes also configured to trigger at a required interval by itself.
(b) Debug: Debug node is used to print output messages in the debug window. The button on the right of the debug node is to disable the node in case the user does not want to print the messages.
(c) Function: Function node allows writing codes in JavaScript and configuring the flows.
(d) Change: Change nodes can be used to modify a message's properties without changing code in the function node.
(e) Switch: By defining a set of rules against each output, switch node is used to reroute the messages to different branches.

2. *Configuration Nodes*: A configuration node is a type of node that has reusable configuration and is shared by regular nodes in the flow. For example, MQTT In and Out can be used as MQTT broker configuration nodes to share connections with the broker.
3. *Workspace*: Workspace is the area where nodes are dragged from the palette of notes on the left and flows are completed. On top of the workspace, there are different tabs for different flows, if required. On the right, there's a debug window and all the extra information is also displayed there.
4. *Flow*: Flow is used to describe a set of connected nodes. A flow is described as a tab within the editor. A simple example of a basic flow to print "Hello World!" is shown in Fig. 8.6.
5. *Wire*: Wires are used to connect the flows and it represents how the message is passed between different nodes.
6. *Palette*: On the left side of the editor, palette is present where all the nodes are available. Extra needed nodes that are not present by default, like Dashboard and Mosca broker, can be downloaded using the palette manager.
7. *Import/Export*: Using import and export options, flows can be shared with others. Flows can be easily downloaded in the form of JSON files and can be shared via email, Github, etc.

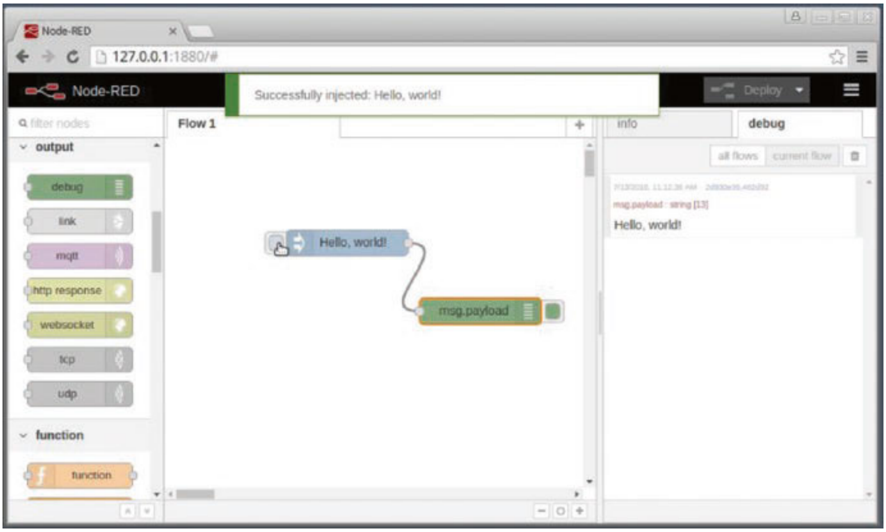


Fig. 8.6 "Hello World!" flow

8. *Subflow*: A subflow is a compilation of nodes that is represented as a single node in the editor. They reduce visual complexity of the flow and can be reused easily in multiple flows.
9. *Deploy*: Flows are deployed when they are complete. Any present errors are shown in the debug windows when flows are deployed.
10. *Dashboard*: Node-RED Dashboard provides a set of nodes to quickly create a live data dashboard. Users can create and install widget nodes like Graph, Gauge, and Switch. One example dashboard is shown in Fig. 8.7.

Node-RED can be used to build, test, and deploy IoT automation projects and develop web services using Node-RED built-in nodes. Some common projects that can be deployed using Node-RED are:

- Using twitter to control Raspberry Pi
- Many IoT automation projects like smart room and smart city implementation
- Developing web services using Node-RED built-in HTTP nodes
- Retrieving data from a certain web page
- Creating your own UI by customizing widgets on node red dashboard

5.3 NodeMCU Development Board

NodeMCU is an open-source software development board built around a system-on-chip (SoC) known as ESP8266. It contains a microcontroller, ESP8266 for WiFi connection, RAM, and many GPIO ports, which makes it an excellent choice for

Fig. 8.7 Node-RED dashboard [13]

many Internet of Things automation projects. NodeMCU is a modest open-source IoT platform used for prototyping board designs. Specifications of the NodeMCU board are as follows [14]:

- Operating voltage: 3.3 V
- Input voltage: 7-12 V
- Digital I/O pins: 16
- Analog input pins: 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Clock speed: 80 MHz
- SRAM: 64 KB
- Flash memory: 4 MB
- Can be used in any IoT project due to its small size
- USB-TTL based on CP2102 is included onboard, enabling plug n play
- PCB antenna

NodeMCU pinout diagram is shown in Fig. 8.8.

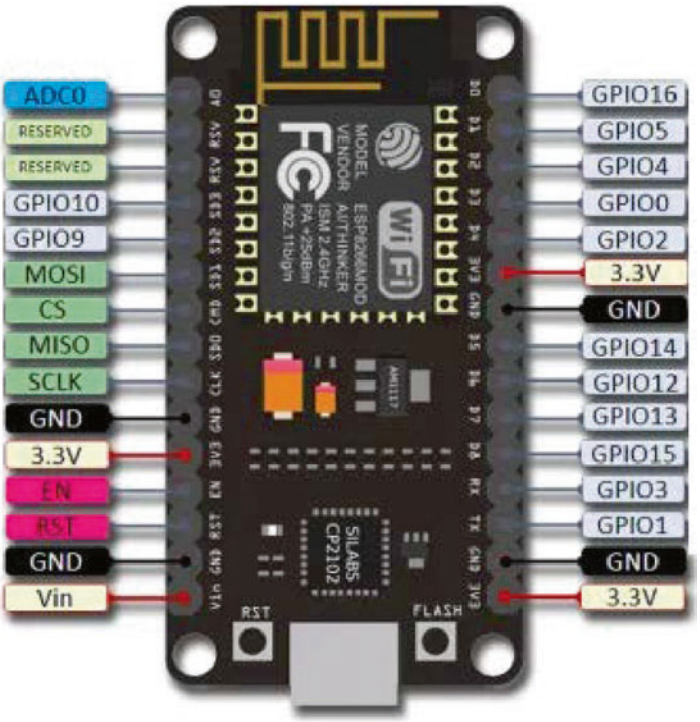


Fig. 8.8 Pinout diagram of NodeMCU [7]

5.4 Sensors

5.4.1 DHT11 Sensor [7]

The DHT11 sensor, also known as temperature and humidity sensor, has an 8-bit microcontroller that provides temperature and humidity in the room as serial data. This sensor is usually calibrated and easy to use with other microcontrollers as well. The range of humidity and temperature for DHT11 is 20% to 90% and 0 °C to 50 °C with a precision of ±1% and ± 1 °C, respectively. Table 8.2 gives the pin description and Fig. 8.9 shows the pin-out diagram of the DHT11 sensor.

Specifications of DHT11 sensor are as follows:

- Output: Serial data
- Operating current: 0.3 mA
- Operating voltage: 3.5 V to 5.5 V
- Temperature range: 0 °C to 50 °C
- Humidity range: 20% to 90%
- Accuracy: ±1 °C and ± 1%

Table 8.2 Pin description of DHT11 sensor

PIN	Name	Function
1	DOUT	Single-bus, serial data
2	GND	Ground
3	VCC	+5 V

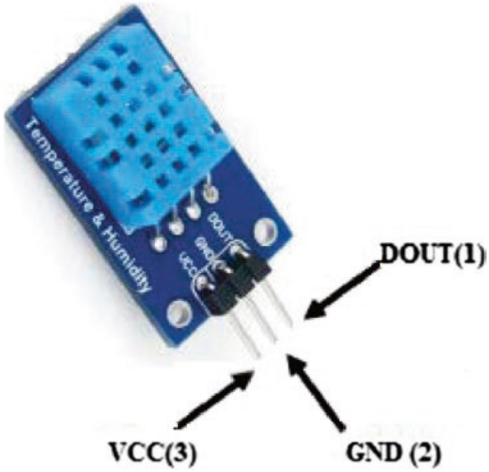


Fig. 8.9 Pin-out diagram of DHT11 module [15]

Some common applications for DHT11 are:

- Measure temperature and humidity where the sensor is kept
- Can be used at local weather stations
- Automatic climate control by interfacing this module in an automation project

5.4.2 PIR Sensor [7]

PIR stands for passive infrared sensor. All the objects emit infrared light which does not fall in the visible spectrum. PIR sensor is an electronic sensor that detects that light. PIR sensors are mostly used in motion detectors. Table 8.3 lists the pin description and Fig. 8.10 shows the pin-out diagram of the PIR sensor. Indoor and outdoor ranges of PIR sensor, in which it can detect the motion, are given as follows:

- Indoor passive infrared: Detection distances range from 25 cm to 20 m.
- Outdoor passive infrared: The detection distance ranges from 10 meters to 150 meters.

Table 8.3 Pin description of PIR sensor

PIN	Name	Function
1	VCC	+5 V
2	OUT	Output
3	GND	Ground

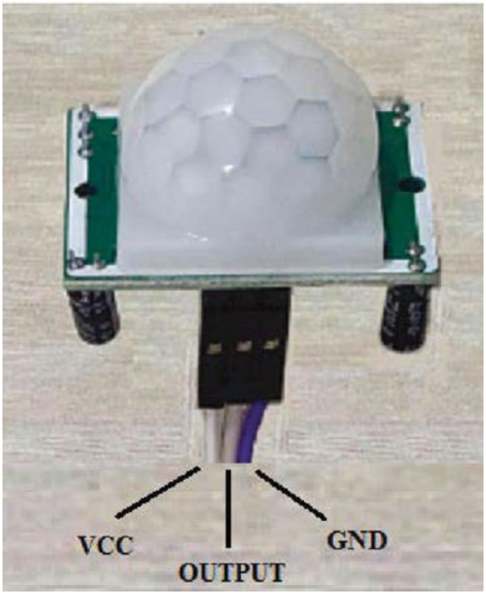


Fig. 8.10 PIR sensor

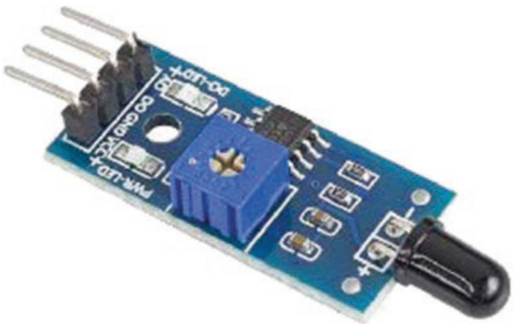


Fig. 8.11 Flame sensor module [15]

5.4.3 Flame Sensor [16]

A flame sensor is mainly designed for detecting as well as alerting the occurrence of a fire or a flame. The response of these sensors is more accurate as well as quicker than any smoke detectors because of its working mechanism. This sensor works through a coating of oil, water vapors, dust, etc., as it uses the infrared flame flash method. The flame sensor is shown in Fig. 8.11 and its pin description is given in Table 8.4.

Table 8.4 Pin description of flame sensor

PIN	Name	Function
1	VCC	3.3 V – 5.3 V
2	GND	Ground
3	AOUT	Analog output
4	DOUT	Digital output

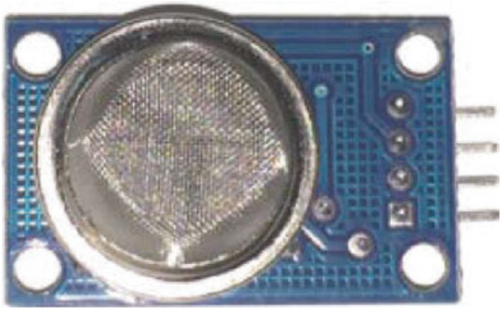


Fig. 8.12 MQ135 gas detector module [7]

Table 8.5 Pin description

PIN	Name	Function
1	VCC	+5 V
2	GND	Ground
3	AOUT	Digital output
4	DOUT	Analog output (0-5 V)

5.4.4 MQ135 Gas Detector Sensor Module [7]

The gas sensor module is a sensing material that is covered with a steel exoskeleton. This sensing element is connected to the connection leads. The current flows through these leads are also known as heating current. When any gas comes in contact with the sensing material, it gets ionized and is absorbed by the material. Hence, due to this ionization, the resistance through the sensing element varies, therefore altering the value of current going out of it. Figure 8.12 shows the gas detector sensor module and Table 8.5 gives its pin description.

5.5 Hardware Test Bed for the Smart Room Application

Figure 8.13 shows the hardware test bed for the smart room application. All the four sensors, as discussed in Sect. 5.4, namely PIR sensor, flame sensor, DHT11 temperature and humidity sensor, and gas sensor, are connected to a single NodeMCU

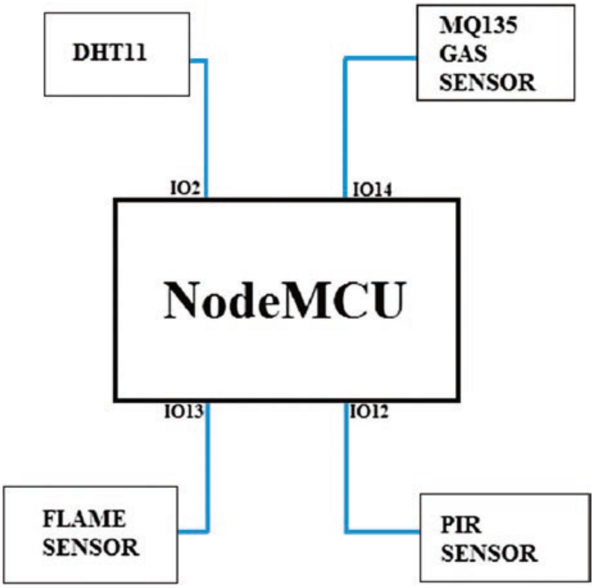


Fig. 8.13 Hardware test bed for the smart room application

board. IoT-based smart room will have the features of i) always keeping track of temperature and humidity conditions inside the room by DHT11 sensor, ii) detecting any motion inside the room by PIR sensor, and iii) detecting gas leakage or fire in the room with the help of gas sensor and flame sensor, respectively. Three equivalent flows are realized for three different protocols, MQTT, HTTP, and WebSocket, on Node-RED and the readings that are received by different sensors will be displayed on the Node-RED dashboard. The data received is heat index, temperature, and humidity from DHT11 sensor, and there are three indicator LEDs that will turn on if there will be any gas leak, flame detection, or person detection in the room, from gas sensor, flame sensor, and PIR sensor, respectively.

Circuit connections are as follows:

- DHT11 temperature and humidity sensor is connected at GPIO Pin 2.
- MQ135 gas detector module is connected at GPIO Pin 14.
- Flame sensor is connected at GPIO Pin 13.
- The PIR sensor is connected to GPIO Pin 12.
- Power supply and Gnd are given same with the help of breadboard (not shown in the above diagram).

After three protocols are implemented by deploying the flows made on Node-RED, response time is observed and compared.

5.6 Implementation

A smart room test bed is designed to evaluate every protocol and compare them on the parameters like delay and security. The hardware interfacing has been discussed in Sect. 5.5. Smart room implementation was done by interfacing DHT11 sensor, PIR sensor, MQ135 gas module, and flame sensor with NodeMCU as shown in Fig. 8.14. DHT11 is used to determine the humidity and temperature conditions and heat index of the room, and PIR sensor is used to alert the user if there is anyone present in the room. Other sensors used are flame sensor and MQ135 gas sensor module to alert if there is any gas leak or fire in the room. This test bed was used to compare three protocols, MQTT, HTTP, and WebSocket, using Node-RED flows.

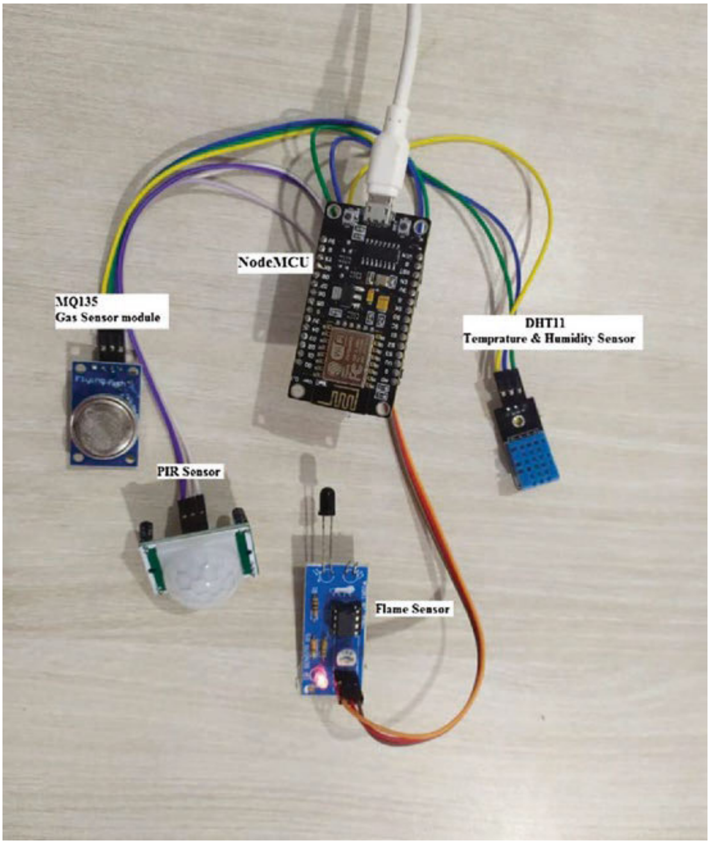


Fig. 8.14 Smart room implementation with DHT11, MQ135, PIR, and flame sensor

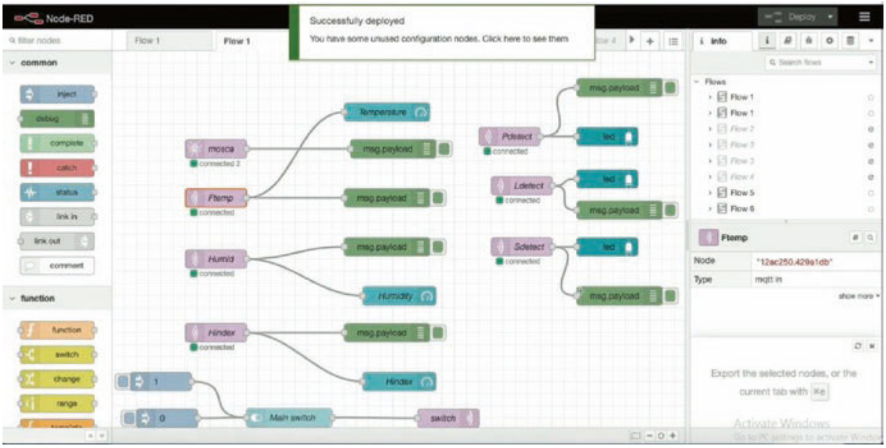


Fig. 8.15 MQTT flow of smart room implementation in Node-RED

5.6.1 MQTT Flow on Node-RED

To implement the MQTT protocol, a Mosca broker was used. Mosca broker is the most commonly used Node.js MQTT broker due to its simple and efficient performance. The MQTT flow diagram implemented in Node-RED is shown in Fig. 8.15.

To configure MQTT, a broker needs to be mentioned in every mqtt node. For configuration, name, IP address of local server, QoS, and topic for subscription are required. All the sensors are connected to NodeMCU and send data on the dashboard via broker. Hence, MQTT connection is established. When we click deploy, the flow is completed and all the readings for temperature, humidity, and heat index are displayed on the dashboard, as shown in Fig. 8.16. The temperature, humidity, and heat index recorded by the sensor used are 32.6 °C, 42%, and 33.58 °C, respectively. And the presence detector is red, which indicates someone's presence in the room. The grey color in gas detector and fire detector indicates the absence of fire and gas leakage.

5.6.2 HTTP Flow on Node-RED

In an HTTP connection, a request is sent from the client to a backend server and a response is sent immediately. After the required information is sent to the client, connection gets closed. Hence, HTTP is known as a unidirectional protocol. To establish HTTP protocol in Node-RED, HTTP request nodes are used and temperature, humidity, and heat index values are requested, and the same will be shown on the dashboard using dashboard (gauge) nodes. An alert will be sent to the UI as soon as the PIR, gas leak sensor, or flame sensor will be turned on due to detection of person, gas, or flame, respectively, in the room using the dashboard (led) nodes. Figure 8.17 shows the Node-RED flow for smart room implementation through HTTP. After the flow is deployed, the http status nodes read a value of 200, which

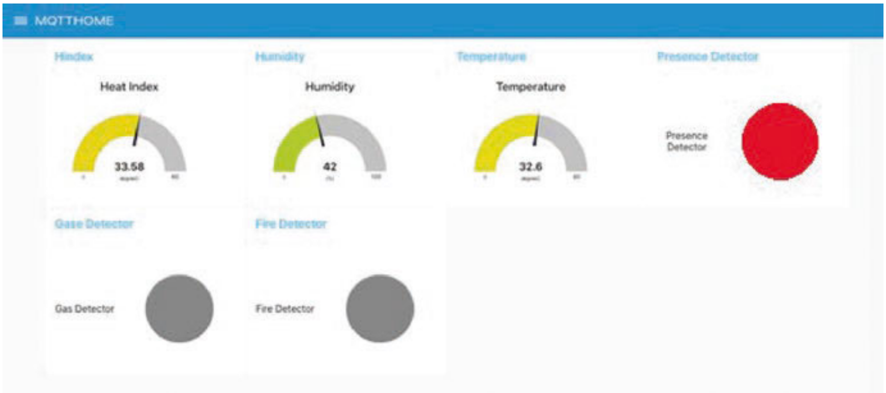


Fig. 8.16 Node-RED dashboard from MQTT protocol implementation

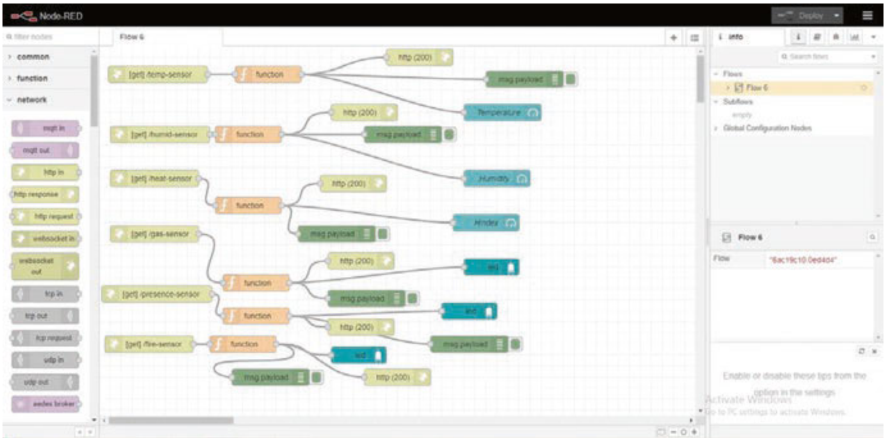


Fig. 8.17 HTTP flow of smart room implementation in Node-RED

means that the client has requested the information and the server has replied. Hence, HTTP connection is established.

Values can be seen on the dashboard as soon as the request is sent, as shown in Fig. 8.18. The temperature, humidity, and heat index recorded by the sensor used are 32.6 °C, 42%, 33.58 °C, respectively. The grey color in gas detector and fire detector indicates the absence of fire and gas leakage, and the presence detector is red, which indicates someone's presence in the room.

5.6.3 WebSocket Flow

WebSocket protocol is used when bidirectional communication is required. WebSocket connection often starts with ws:// & wss://. Once a connection is open and a handshake occurs, bidirectional exchange of messages starts, and this

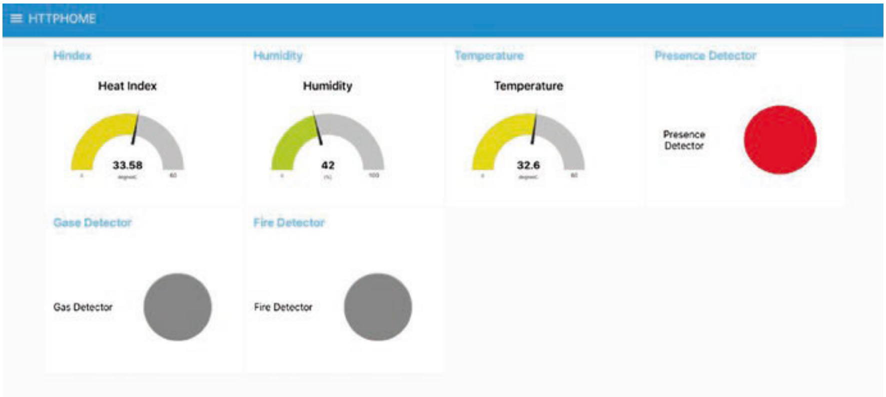


Fig. 8.18 Node-RED dashboard from HTTP implementation

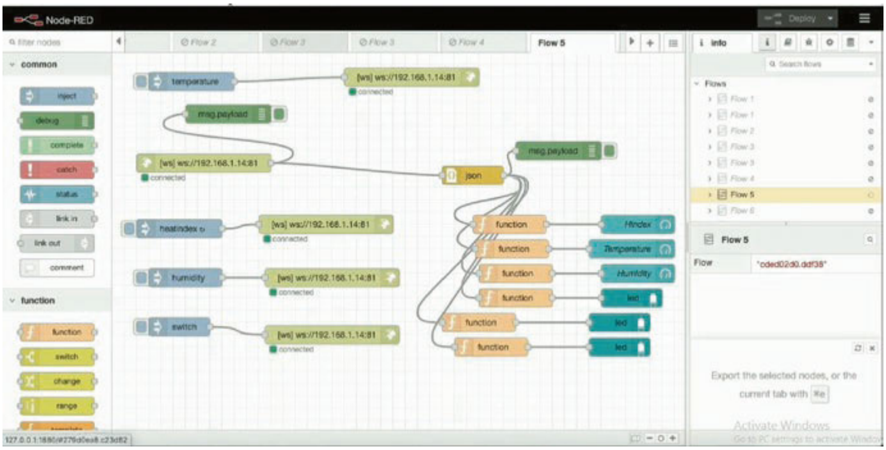


Fig. 8.19 WebSocket flow of smart room implementation in Node-RED

connection stays alive until the server or the client dies or either of them decides to kill the connection. To establish the realization of WebSocket protocol in Node-RED, WebSocket out nodes are used with the IP address of the local server. WebSocket flow implemented in Node-RED using ws nodes and dashboard nodes to display the data received from sensors is shown in Fig. 8.19.

After deploying this flow, the temperature, humidity, and heat index readings and fire, gas, and presence detection are updated as shown on the dashboard as shown in Fig. 8.20. The temperature, humidity, and heat index recorded by DHT11 temperature and humidity sensor are 32.6 °C, 42%, and 33.58 °C, respectively. The presence detector is red, which indicates someone's presence in the room, and the grey color in gas detector and fire detector indicates the absence of fire and gas leakage.

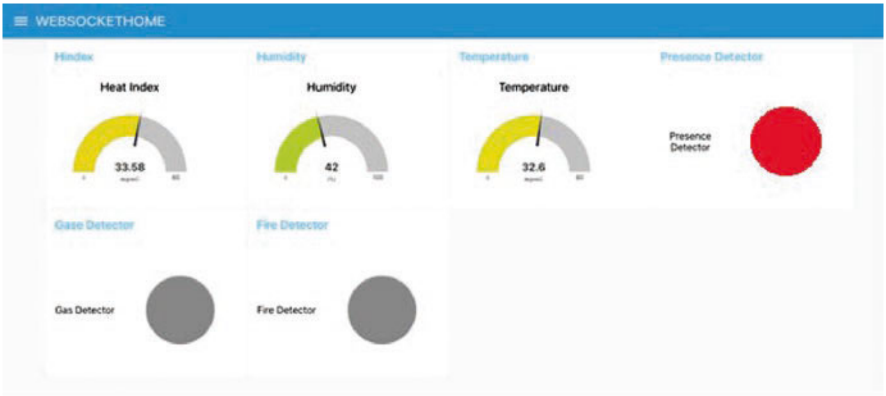


Fig. 8.20 Node-RED dashboard from WebSocket protocol implementation

5.7 Results and Discussion

After performing the above experiment on three different protocols, it was observed that all the readings of temperature, humidity, and heat index are same because same circuit (with same NodeMCU board and same sensors) was used for all three Node-RED flows and the data was collected on the same time and day and in the same room. The response time of all three protocol flows was recorded and is shown in Table 8.6.

Following conclusions can be drawn from the comparison table above:

(a) Out of all three protocols, MQTT turned out to be the fastest protocol. MQTT has some architectural advantages which makes it the fastest protocol. It uses the publish/subscribe method to communicate back and forth. Apart from having the fastest response time, with publish/subscribe architecture, MQTT is a better and viable option for a connection that requires QoS options and priority settings with a transport layer security.

(b) In comparison, WebSocket and HTTP use a client/server approach. Every time a request is sent from the client via HTTP, the server connects with the client, sends required information, and disconnects immediately. A new request is sent every time the client needs information. WebSocket has a handshake mechanism, in which the connection is kept alive until one of the parties cancels it. Hence, even though HTTP and WebSocket both use client/server architecture, WebSocket is a faster protocol than HTTP with a response time of 50 ms only. Therefore, WebSocket is a better choice where a faster bidirectional client/server mode of communication is required.

Table 8.6 Response time of three protocols in smart room implementation

S. no	Protocol	Response time
1.	MQTT	30–40 ms
2.	HTTP	1–2 sec
3.	WebSocket	50–60 ms

5.7.1 Comparison with Existing Work

Many authors have observed and compared the response time of many application-layer protocols. In [17] Năstase L concluded that MQTT, XMPP, and AMQP have a round-trip time of 0.448 ms, 0.373 ms, and 90.75 ms, respectively. In ref. [18], H Kasuma Aliwarga compared two protocols, MQTT and HTTP, in terms of data latency and observed that MQTT is faster than HTTP for a certain kind of IoT application. Also, the power consumption in MQTT protocol was lower than that of HTTP [19]. Thus, MQTT uses less power to maintain an open connection. Authors in ref. [19] concluded that the MQTT is the best communication technique to use in many IoT automation projects.

6 Summary and Conclusion

- IoT is a global network that shares data among themselves and processes them in a manner that can be put to a practical use in a number of applications. There are endless possibilities where IoT architecture can be implemented.
- Business insider intelligence estimated that the 24 billion IoT devices will be present by the beginning of 2021 following with a lot of job opportunities in the upcoming future.
- IoT communication protocols are modes of communication that ensure maximum security of data being exchanged between connected "things." These protocols are divided into two categories—network protocols and data protocols.
- Some common applications for IoT are healthcare, everyday life, agriculture, disaster management, industrial automation, smart cities, smart traffic and weather management, etc.
- Node-RED is an open-source programming tool created by IBM technologies, used to wire together APIs, hardware devices, and online services. It allows end users to work together with web services as well as hardware by removing low level coding tasks. It is available in the Bluemix IoT starter application.
- A smart room implementation was done in hardware using NodeMCU with DHT11 sensor for temperature, humidity, and heat index readings; PIR sensor for intruder alert; and flame sensor and gas detector module for flame and gas detection, respectively. Three equivalent flows were implemented in Node-RED for three protocols—MQTT, HTTP, and WebSocket. It was concluded that MQTT, along with the priority settings and minimum overhead communication,

has the fastest response time among the three protocols worked upon and is suitable for automation projects.
- From the comparison provided above, protocol choice would be easy for a certain application. Implementation was done on Node-RED software with a smart room circuit. More protocols can be implemented easily on Node-RED and their response time can be compared.

7 Future Scope

From the comparison of protocols presented in this work, choice of a protocol would be easy for a certain application. More protocols like CoAP and AMQP can be implemented easily and their response time and power consumption can be compared on the test bed presented in this work or any other IoT-based application using Node-RED.

References

1. Oracle. (1977). Register. https://www.oracle.com/in/index.html.
2. Edureka. (2011). Introduction to Internet of things: IoT tutorials with IoT applications. https://www.edureka.co/blog/iot-tutorial/.
3. Swayam Cental. (2017). Introduction to Internet of Things. https://onlinecourses.nptel.ac.in/noc21_cs63/preview
4. GeeksforGeeks. (2009). World of IoT Boards. https://www.geeksforgeeks.org/world-of-iot-boards/.
5. Vasudevan, S. K., Nagarajan, A. S., & Sundaram, R. M. D. (2019). *Internet of things* (1st ed., pp. 14–18). Wiley.
6. Thingboard. (2016). Register. https://thingsboard.io/
7. Electroniclinic. (2019). India. https://www.electroniclinic.com/nodemcu-esp8266-pinout-features-and-specifications/
8. Wikipedia. (1999). New York, United states. https://en.wikipedia.org/wiki/MQTT.
9. Tutorial s Point. (2006). HTTP Overview. https://www.tutorialspoint.com/http/http_overview.htm
10. Meera, M. S., & Rao, S. N. (2018). Comparative analysis of IoT protocols for a marine IoT system. Paper presented in Proceedings of the 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2049–2053, Bangalore, India, September 2018.
11. Mijovic, S., & Shehu, E., Buratti, C. (2016). Comparing application layer protocols for the Internet of Things via experimentation. Paper presented at Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), IEEE 2nd International Forum on IEEE, pp. 1–5, 2016.
12. Node-RED. (2013). New York, United States. https://nodered.org/
13. IoTSharing. (2018). http://www.iotsharing.com/2018/05/how-to-build-iot-dashboard-using-node-red.html
14. Mechatronics Blog. (2012). Colorado, United states. https://mechatronicsblog.com/esp8266-nodemcu-pinout-for-arduino-ide/.

15. Wikipedia. (1999). New york United states – https://www.waveshare.com/wiki/DHT11_Temperature-Humidity_Sensor; https://www.waveshare.com/wiki/Flame_Sensor
16. Elprocus. (2012). Flame Sensor & Its applications. https://www.elprocus.com/flame-sensor-working-and-its-applications/.
17. Năstase, L., Sandu, I.E., Popescu, N. (2017). An experimental evaluation of application layer protocols for the internet of things, pp. 403–412.
18. Kasuma Aliwarga, H., Herfian, A., & Fandi, B. (2020). *Performance comparison of fleet management system using IoT node device based on MQTT and HTTP protocol*. AIP Conference Proceedings.
19. Wukkadada, B., Wankhede, K., Nambiar, R., & Nair, A. (2018). "Comparison with HTTP and MQTT" in Internet of Things (IoT), in Proceedings of the International Conference on Inventive Research in Computing Applications, Coimbatore, p. 249–253.

Chapter 9
Comparative Study of Static and Hybrid Analysis Using Machine Learning and Artificial Intelligence in Smart Cities

Shagil Chaudhary, Ramesh Amgai, Shouvik Das Gupta, Nida Iftekhar, Sherin Zafar, and Anil Kumar Mahto

Abstract Smart cities, with fast increment in urban development, could be a concerning issue, indeed for created nations. It is developing as one of the complex systems around the world with the increment in request and supply based on assets and administrations. In this modern era, brilliant gadgets are very much required within the building of the foundation of a savvy city. The increment in the populace has expanded challenges in the organization and management of keen cities. These sorts of challenges can be restrained by the usage of specialized progressions by the inhabitants. However, the requirements of a keen city and the improvements around it ought to give benefits, not as it were to the living environment but to consider the human-centered administrations. Additionally, keeping up a more advantageous environment needs the improvement of intelligent information frameworks with enabled IoT innovations. In building a keen city, the method ought to be intuitively kind, so an IoT-based stage is necessary. Malware penetration is getting more regrettable day by day and is considered one of the greatest security dangers to the web. Malware is any pernicious program with the expectation to perform noxious exercises on a focused-on framework. In this term paper, we have examined the two wide methods that are utilized in arrange to viably perform Malware examination and location on venture frameworks to diminish the harm of malware assaults.

Keywords Static analysis · Dynamic analysis · Machine learning · Malware

---

S. Chaudhary · R. Amgai · S. D. Gupta · N. Iftekhar (✉) · S. Zafar · A. K. Mahto
Jamia Hamdard, New Delhi, India
e-mail: nida.iftekhar@jamiahamdard.ac.in

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_9

1 Introduction

AI and machine Learning can alter the way shrewd cities work in different areas. In any case, the executing and coordination of software and equipment stages, savvy system and calculation, hypothetical arranging, and scientific computing demonstrate of ICT foundation is very much basic for an AI-and ML-based savvy city. This extraordinary issue persuades and motivates scholastics and analysts to show their work enabling AI and machine learning-based measures within the improvement of a shrewd city. Modeling Malware is an executable or a parallel that malevolent in nature. Malware is utilized by assailants to perform a spread of noxious activities like spying on the target, RAT's, Keyloggers, information exfiltration, information encryption, and devastation. Malware alludes to any twofold or executable that's malevolent, in any case, Malware is sorted into advanced categories backed by its usefulness. Here are the shifted assortments of malware:

1. Trojans: This sort of malware masks itself as a genuine program for social building purposes. It can devastate and exfiltrate information and may indeed be utilized for spying [5].
2. RAT's: This kind of malware permits the aggressor to remotely get to and execute commands on the framework. Its usefulness will be amplified with modules like keyloggers
3. Ransomware: This sort of malware scrambles all records on the framework and holds the framework and its information for ransom.
4. Dropper: This sort of malware whose reason is to download/drop extra malware.

Malware examination is the method of analyzing a malware sample/binary and extricating the most extreme sum data as conceivable from it. The information extricated makes a difference to get the scope of the usefulness of malware, how the framework was tainted with the malware, and the way to guard against comparable assaults inside the future. To exfiltrate valuable markers like registry entries/keys and filenames for the aim of generating signature which is able to be wont to distinguish future discovery. Taking after are the shapes of Malware Analysis:

1. Static examination is the method of analyzing malware without executing or running it. The target is to extricate the most extreme sum metadata from the malware as conceivable.
2. Dynamic analysis is the method of executing malware and analyzing its usefulness and behavior. The target is to know precisely how and what the malware does amid the execution. This could be depleted by a debugger.
3. Code analysis is the method of analyzing/reverse designing gathering code. This may well be both statically and powerfully done (inactive and energetic code analysis).
4. Behavioral investigation is the method of analyzing and checking the malware after execution. It includes checking the forms, registry sections, and the workings of the malware.

2 Literature Review

Process Hacker could be a free and open-source process viewer. This multipurpose tool will assist you with debugging, malware detection, and system monitoring process [1]. Hackers could be a very valuable tool for advanced users [2]. It can help them to troubleshoot problems or learn more about specific processes that are running on a specific system [3]. It can help identify malicious processes and tell us more about what they are trying to do. within the default settings, it shows you the Processes tab with all the running processes in tree-view [4]. Wireshark is that the world's foremost and widely used network protocol analyzer [12]. It enables you to see what's happening on your network at a microscopic level and is that the factual (and often de jure) standard across many commercial and nonprofit enterprises, government agencies, and academic institutions [13]. Wireshark development thrives due to the volunteer contributions of networking experts around the globe and is that the continuation of a project started by Gerald Combs.

3 Static Analysis

Static Analysis is that the process of analyzing malware/binary without executing it. the target is to extract useful information from the malware, this can help us get a concept of the kind of malware and what the malware can do. This information is beneficial for future analysis because it will allow us to efficiently analyze the sample going forward. Following are the steps to approach a sample:

1. Identifying the file type: Target OS, architecture, and format (dll, exe)
2. Identifying the malware: Generating a hash of the malware. This may give the malware a novel identifier. Using the hash to work out if anyone else has analyzed the malware.
3. Strings: Strings give us an idea/glimpse of what malware can do.
4. Packing and obfuscation: Obfuscation and packing are techniques accustomed prevent detection. Unpacking or DE obfuscating can reveal additional information.
5. PE headers: The PE header reveals lots of data on the malware functionality.

Identifying the record sort i greatly imperative since it makes a difference in us distinguishing the target OS and thus the comparing design. An illustration of a Windows executable record is that the PE (Portable Executable) [6]. A PE would be inside the frame of; .exe, .dll, etc. To precisely distinguish a file sort, we need to inquire about the record signature. this may be to maintain a strategic distance from wrong positives caused by the business of twofold expansions. The record signature exists on the record header. The file signature for PE records is spoken to by hexadecimal values of 4D 5A or MZ inside the primary 2 bytes (0–1).

PE programs indeed have the take note “This program can’t be run in DOS mode” The PE header starts at hex 50 45. The devices to be utilized are HxD – Hex Editor, Exeinfo PE – Recovers the windows PE header data. It too recognizes in case the executable has been stuffed and recognizes the packer form and the way to unload it, Pestudio and CFF pioneer [7]. Test of the malware in an unloaded frame is exceptionally imperative for security reasons. The unloading is accomplished by means of Olly debugger.

4 Analysis Using Tools

4.1 Hdx Tool

Entities Hdx tool comes pre-packaged with the flare VM that may be accustomed get a summary of the hexadecimal code of the malware. All we’ve to try and do is solely drag the unpackaged file to the tool and it will immediately open up. At initial glance, we are ready to see the name “MZ” this means that the file could be a portable executable also the second indicator of the file being portable executable is that the first two bytes being “4D and 5A” this can be the signature of a file being portable executable (Fig. 9.1).

Another indicator for a conveyable executable file is that it shows the subsequent message “this program cannot be run in DOS mode,” within the decoded text as shown by the hdx tool we are able to see the start of the PE header indicated by “PE.” Through this fashion, we can do the file signature detection manually.

4.2 CFF Explorer Tool

This tool is additionally within the category of an automated tool for file type detection. As soon as we input the malware files, the tool immediately tells us that the file is portable executable and is of 32 bits, which is clearly written within the file-type section. Additionally, it also provides the data about the hash of the file both MD5 and SHA-1, also the tool provides the knowledge about when the malware was created, modified as we accessed [8]. All this information is extremely crucial for the analysis of malware (Fig. 9.2).

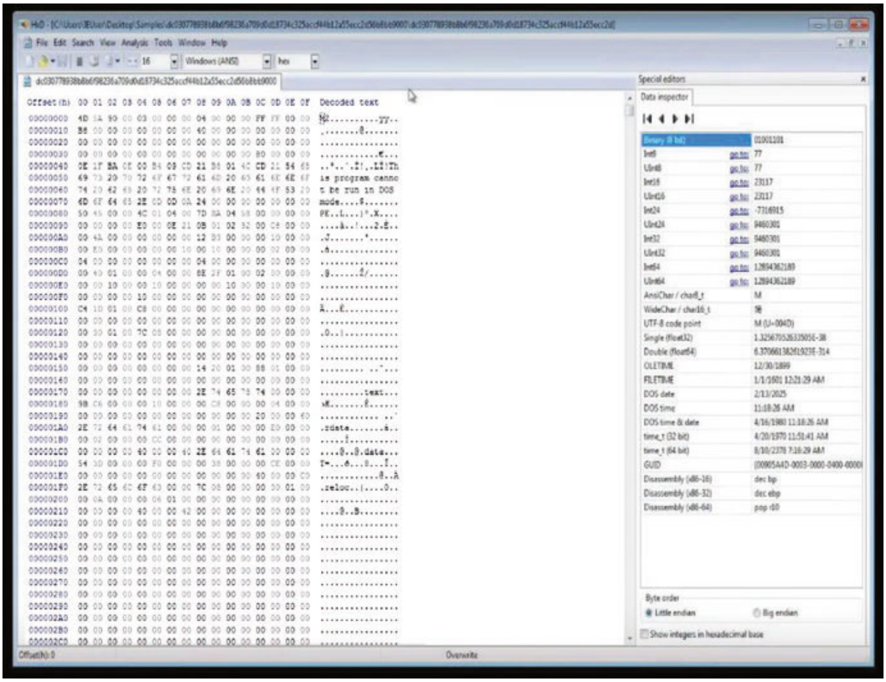


Fig. 9.1 Interface of Hdx Tool

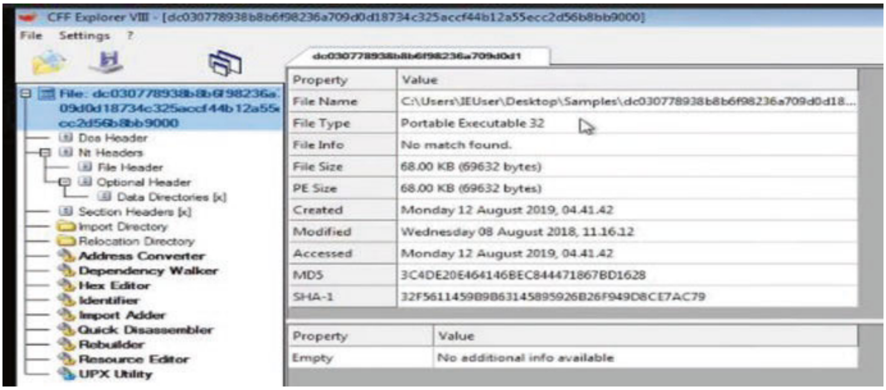


Fig. 9.2 Interface of CFF Explorer

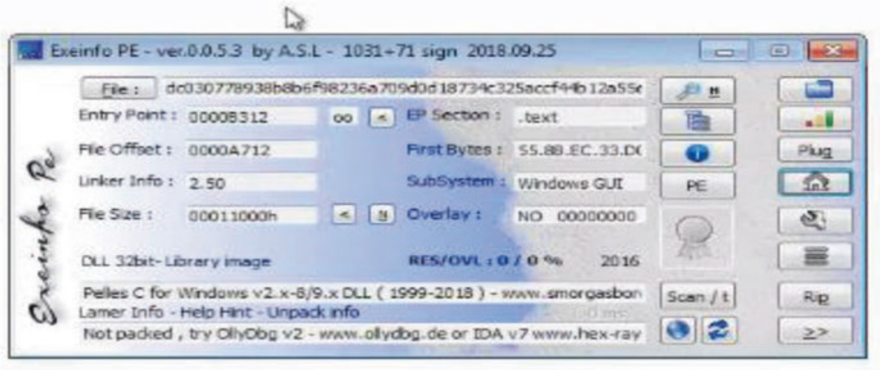


Fig. 9.3 Detailed view of Exeinfo PE

4.3 *Exeinfo PE Tool*

The EXEinfo PE tool is more concerned with the PE header a part of the malware. However, it does give the knowledge about the kind of file we are managing (Fig. 9.3).

It provides more information about the file like file offset, linker info, and file size. It also tells whether the file is unpacked or packed. If we click on the offset, we immediately visit the subsequent screen with the hex code that shows all the indications of a conveyable executable file as discussed within the summary of the previous tool [9]. Additionally, this tool also tells us whether the file is packed or not, which is vital, since if the file is packed, it will be unpacked with a tool like OllyDbg.

4.4 *Malware Hashing*

Malware hashing is the process of generating cryptographic hashes for the file content of the target malware. We are hashing the malware file. The hashing algorithms utilized in malware identification are MD5, SHA-1, and SHA-256. The hashing process gives us a novel digest referred to as a fingerprint. This suggests we will create unique fingerprints for malware samples. We need to hash for accurate identification of malware samples, instead of using file names for malware. Hashes are unique [10]. Hashes are wont to identify malware on malware analysis sites (Virus Total). Hashes will be an accustomed hunt for any previous detections or for checking online if the sample has been analyzed by other researchers [14].

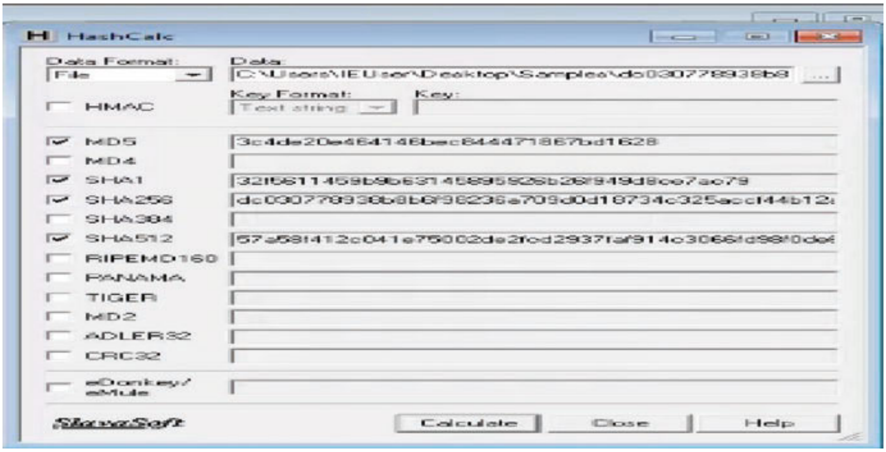


Fig. 9.4 HashCalc

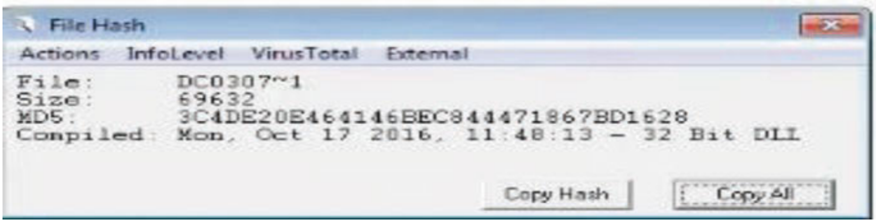


Fig. 9.5 File Hash

4.5 Hashcalc Tool

This tool is employed to induce the hashes of the malware file as hashes are unique in order that it is often wont to detect the file employing a database of hashes. By inputting the file into hashcalc we are able to immediately retrieve the MD5 and SHA-1 hashes of the file. During this tool, we are able to select the input format of the file also and also we get options to induce all kinds of hashes (Fig. 9.4).

If we are using my hash that is pre-installed within the flare VM we are able to simply right-click on the unpacked malware file and click on get hashes and that we are introduced to a window with the MD5 hash and other information about the file (Fig. 9.5).

Fig. 9.6 VirusTotal

Fig. 9.7 Detailed view of virus in VirusTotal

4.6 VirusTotal

This automated malware analysis website can be used after getting the hashes from hashcalc to induce even more information about the malware [11]. On the website, we get the choice to stick the file hash and hunt for the relevant information. It will tell us if other engines detected the file as malware or not and that we get all types of knowledge about the detected malware from several engines. within the information section, the user will see the file type and every one the assorted names it won't to have (Figs. 9.6 and 9.7).

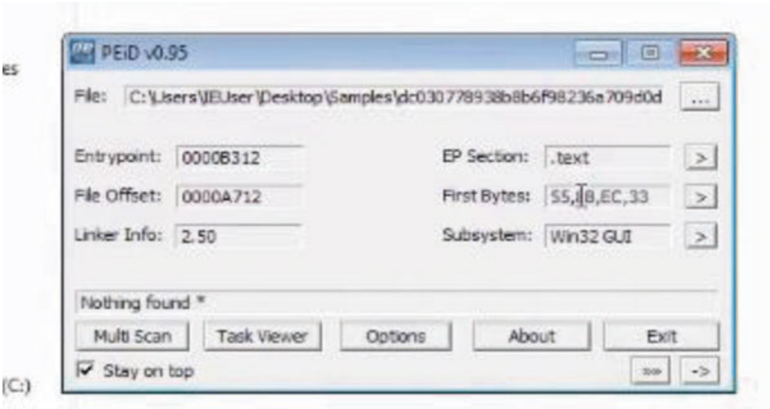


Fig. 9.8 Paid

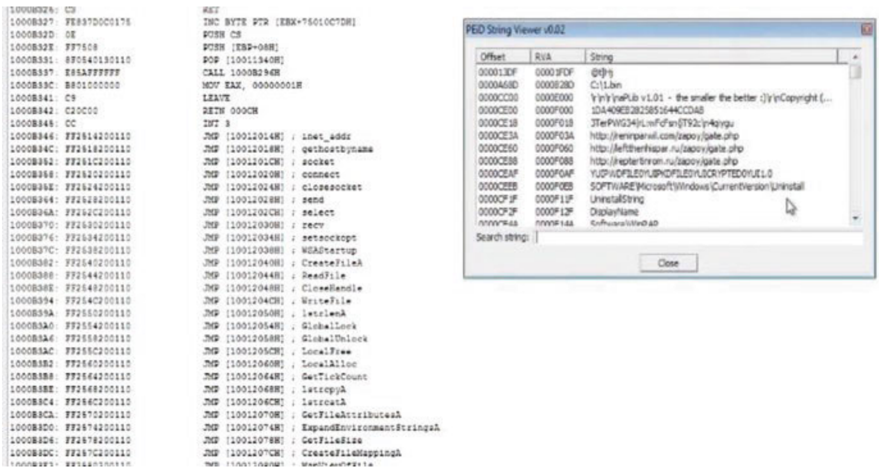


Fig. 9.9 String view

4.7 *Tools Utilized*

In the flare VM the paid tool is employed to urge the strings as output in a very panel. All it requires is an unpacked malware which is fed as an input (Fig. 9.8).

If we glance at the output then we will see that we have the offset and the actual strings that are present within the malware (Fig. 9.9).

On the flare, if we right-click on the unpacked file and choose to get strings then the string command utility tool will immediately give us the ASCII strings that exist within the file (Fig. 9.10).

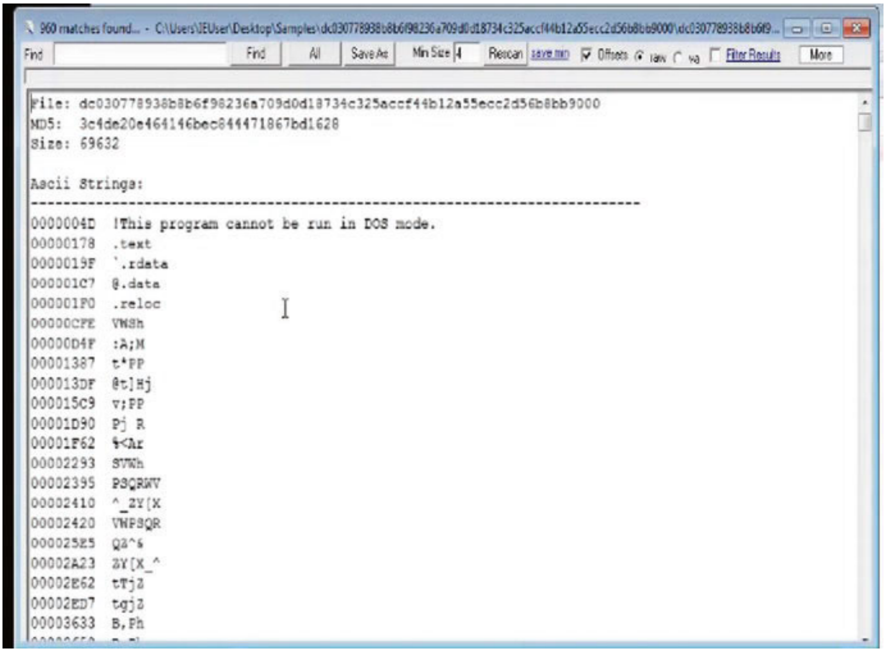


Fig. 9.10 Strings matching

On the flare VM by using the PowerShell we simply type strings and that we can see that we are able to get every kind of string through the command utility. this is often a manual method to induce the strings (Fig. 9.11).

Through the PowerShell program line, we are able to save the strings to a specified location on the VM and might later open the file to appear at the strings. As we open the txt file, we are ready to see that the minimum number of characters is 6 which is very important because it clears away all the rubbish strings that are present to confuse the analyzer [15]. Additionally, within the txt file, we are able to see that there's some URL present meaning that there's a backdoor connection between the machine being infected and therefore the creator of the malware. within the strings, we discover that there is a registry key that indicates that the malware uninstalls itself and shows another functionality like stealing credentials which is that the main function of the malware (Fig. 9.12).

5 Packers and Unpacking

A packer may be a device that is acclimated to compress the substance of the malware. Aggressors will utilize packers to jumble the substance of the malware, this makes it troublesome to explore strings. Packers compress an executable and when

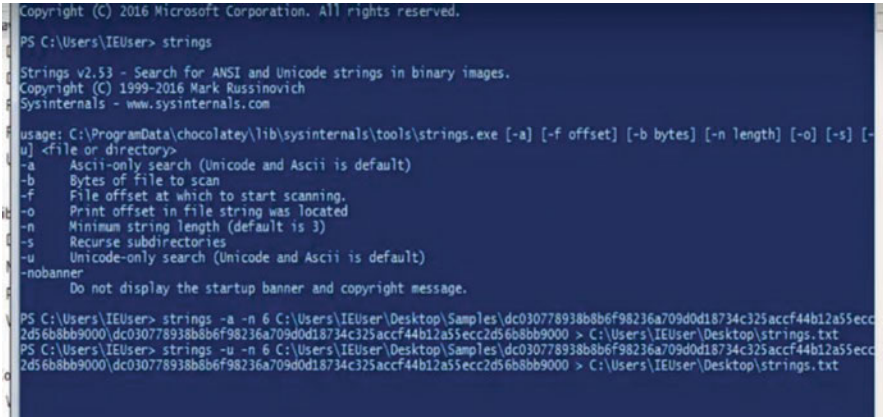
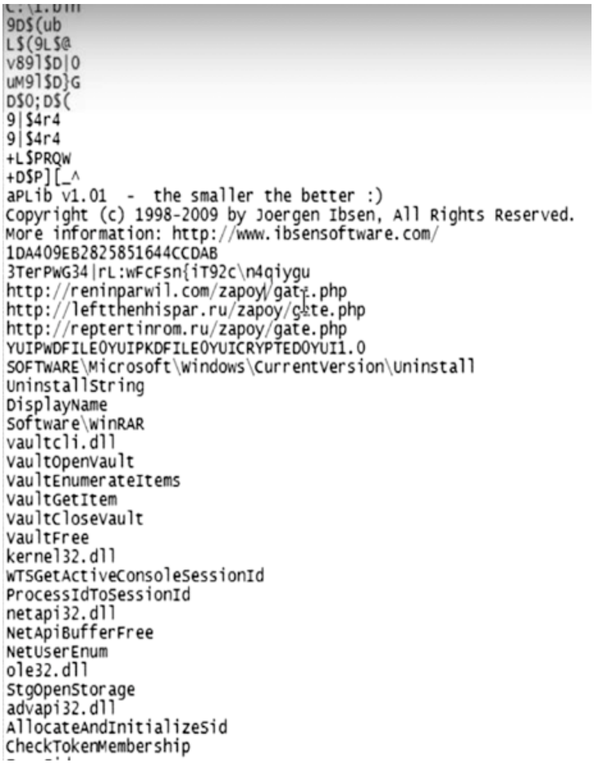


Fig. 9.11 PowerShell view

Fig. 9.12 Leftover information



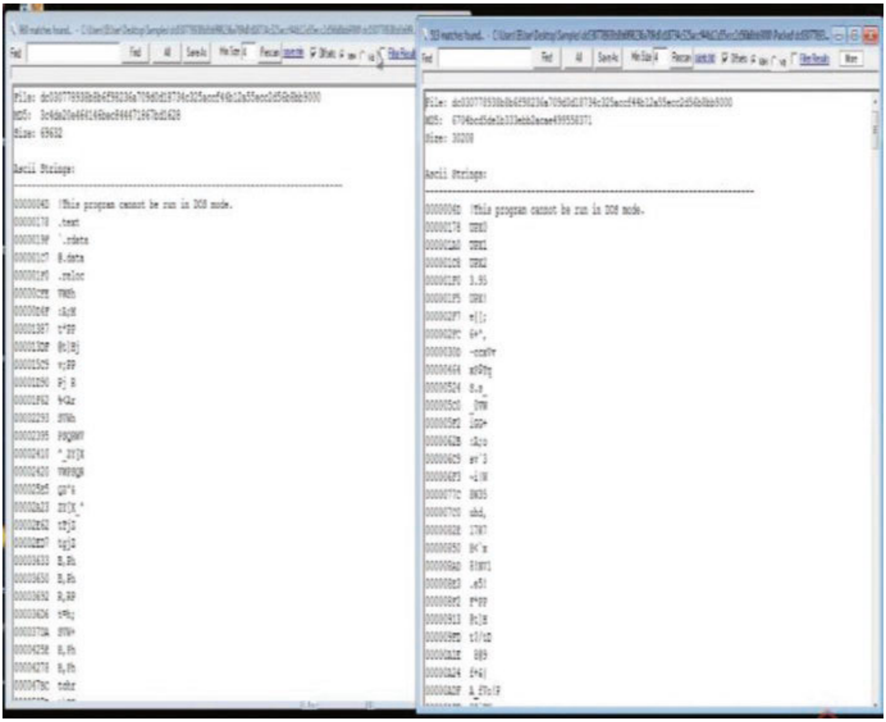


Fig. 9.13 Comparison between packed and unpacked malware

executed the stuffed executable are decompressed. this grants us to explore the beginning unloaded executable. Here, we have a pressed record that we will be utilizing to unload and see what contrasts are shown in both the adaptations. Here on the off chance that we compare both the forms we see that inside the unloaded adaptation we are clearly able to see the strings that we would like and so the registry keys though in case we take a look at the stuffed adaptation all the information is display in an incoherent arrange with few special cases (Fig. 9.13).

The PE header contains the data the OS requires to run the executable. This information is incredibly useful because it can give us more information about the functionality of the malware and the way the malware interacts with the OS. It contains all of the important and necessary information required by the OS to execute the executable. It contains information that specifies where the executable has to be loaded into memory. It contains the libraries that the executable requires to be loaded (dll). It contains information that specifies where the execution begins.

	Machine	SizeOfOptionalHeader	Characteristics	MajorLinkerVersion	MinorLinkerVersion	SizeOfCode	SizeOfInitializedData	SizeOfUninitializedDa
count	138047.000000	138047.000000	138047.000000	138047.000000	138047.000000	1.380470e+05	1.380470e+05	1.380470e+(
mean	4259.069274	225.845632	4444.145994	8.619774	3.819286	2.425956e+05	4.504867e+05	1.009525e+(
std	10880.347245	5.121399	8186.782524	4.088757	11.862675	5.754485e+06	2.101599e+07	1.635288e+(
min	332.000000	224.000000	2.000000	0.000000	0.000000	0.000000e+00	0.000000e+00	0.000000e+(
25%	332.000000	224.000000	258.000000	8.000000	0.000000	3.020800e+04	2.457600e+04	0.000000e+(
50%	332.000000	224.000000	258.000000	9.000000	0.000000	1.136640e+05	2.631680e+05	0.000000e+(
75%	332.000000	224.000000	8226.000000	10.000000	0.000000	1.203200e+05	3.850240e+05	0.000000e+(
max	34404.000000	352.000000	49551.000000	255.000000	255.000000	1.818587e+09	4.294966e+09	4.294941e+(

8 rows × 55 columns

Fig. 9.14 Dataset screenshot

6 Machine Learning and Deep Learning Algorithms for Malware Analysis

For the classification of malware, we are able to automate the method by using machine learning. Initially, we import all the desired libraries required to perform tasks like data manipulation, model construction, etc. then the file must be loaded into a data frame and therefore the separation must be provided. After the successful loading of the file, we take a glance at the dataset by using describe function. Here, we see that the dataset has 55 categories shown as columns here (Fig. 9.14).

After getting a decent idea of what the dataset sounds like we will classify the entries into legitimate files and malicious files within the training set. We even have to drop some of the columns like the Name of the file.

```
Out[15]: legitimate
         0    96724
         1    41323
         dtype: int64
```

Tree-based models utilize an arrangement of if-then rules to come up with expectations from one or more choice trees. All tree-based models will be utilized for either relapse (anticipating numerical values) or classification (anticipating categorical values). An extra-trees classifier. This course executes a meta estimator that matches an assortment of randomized choice trees (a.k.a. extra-trees) on different subsamples of the dataset and employments averaging to upgrade the prescient exactness and control over-fitting. The Additional Trees calculation works by making an outsized number of unpruned choice trees from the preparing dataset. Forecasts are made by averaging the expectation of the choice trees inside the case of relapse or utilizing larger part voting inside the case of classification. ExtraTreesClassifier fits an assortment of randomized choice trees on different subtests of the dataset and employments averaging to boost the prescient precision and control over-fitting.

The ExtraTreesClassifier is additionally wont to select the specified features useful for classifying a file as either Malicious or Legitimate 14 features are identified pro re nata by ExtraTreesClassifier. And if we take a glance at the features extracted by ExtraTreesClassifier we are going to discover that 14 features are used for the classification of the file as malware or legitimate.

Cross-validation may be a resampling method acclimated to assess machine learning models on a constrained information test. The method contains a single parameter called k that alludes to the sum of bunches that a given information test is to be part into. As such, the method is commonly called k-fold cross-validation. The objective of cross-validation is to check the model's capacity to anticipate modern information that wasn't utilized in evaluating it, to hail issues like overfitting or choice inclination and to offer some knowledge on how the show will generalize to an autonomous dataset (i.e., an obscure dataset, for illustration from a genuine issue).

Cross Approval in Machine Learning: four shapes of Cross-validation 1. Holdout Method. 2. K-Fold Cross-Validation. 3. Stratified K-Fold Cross-Validation. 4. Leave-P-Out Cross-validation. Cross Approval can be an awfully useful technique for surveying the adequacy of your show, especially in cases where you would like to moderate overfitting. It is too of utilizing in deciding the hyperparameters of your demonstrate, inside the sense that which parameters will lead to shake foot test blunder. Take the gather as a holdout or test information set. Take the remaining bunches as a preparing information set. Fit a show on the preparing set and assess it on the test set. Hold the assessment score and dispose of the model.

Cross-validation is applied to divide the dataset into random train and test subsets. The test size for the analysis purpose is 0.2 which represents what proportions of the dataset should be included within the test split. After the test and training split has been successfully done the method of building the machine learning model begins. The following models are wont to see which works best for the malware classification (Fig. 9.15).

The above-described models are compared on their accuracies and the model with the simplest accuracy is said to be used for the malware analysis purpose.

Building the below Machine Learning model

```
model = { "DecisionTree":tree.DecisionTreeClassifier(max_depth=10),
        "RandomForest":ek.RandomForestClassifier(n_estimators=50),
        "Adaboost":ek.AdaBoostClassifier(n_estimators=50),
        "GradientBoosting":ek.GradientBoostingClassifier(n_estimators=50),
        "GNB":GaussianNB(),
        "LinearRegression":LinearRegression()
}
```

Fig. 9.15 Configuration used for every model

Saving the model

```
joblib.dump(model[winner],'classifier/classifier.pkl')

['classifier/classifier.pkl',
 'classifier/classifier.pkl_01.npy',
 'classifier/classifier.pkl_02.npy',
 'classifier/classifier.pkl_03.npy',
 'classifier/classifier.pkl_04.npy',
 'classifier/classifier.pkl_05.npy',
```

Fig. 9.16 Saving the model as a pickle file

```
%run malware_test.py "/home/Downloads/Skype.exe"
The file Skype.exe is legitimate
```

To test for the malicious file, an application has been downloaded from malwr.com

```
%run malware_test.py "/home/Downloads/BCN12ui49823.exe"
The file BCN12ui49823.exe is malicious
```

Fig. 9.17 Predicting an unseen file using our model

We can get the model with the most accuracy by using the max function. The model with the utmost accuracy is gradient boosting. This model is saved for further exploration (Fig. 9.16).

To better check the accuracy and functioning of the model, we determine the false positives and false-negative rates. The lower the worth the higher the model is performing. this may be achieved by using the predict function.

The values for the false positives and false negative rates seem very low that the model is ideal for the classification task. the following stage is to load the saved model which may be achieved by joblib function and therefore the loading of features is often achieved by using the function pickle.

After successfully loading the model and therefore the features to figure on we test our model with an unseen file and see the results. For this purpose, python's pefile PE library is employed to construct and build the feature vector, and an ML model is employed to predict the category for the given file supported by the already trained model (Fig. 9.17).

Extending our exploration, a small amount further and applying Deep Learning algorithms to seek out out whether we could differentiate between malicious and non-malicious files. We have built a dataset containing both malicious and non-malicious files. we have extracted their PE file header and sections and created a dataset. Here may be a screenshot of the primary few rows (Fig. 9.18).

6.1 Artificial Neural Networks

The lion's share of neural systems is completely associated with one layer to a diverse. These associations are weighted; the upper the amount the more noteworthy impact one unit has on another, nearly like an individual's brain. since the information goes through each unit the arrange is learning more almost the data. The

	labels	F1	F2	F3	F4	F5	F6	F7	F8	F9	...	F477	F478	F479	F480	F481	F482	F483	F484	F485	F486
0	pe-malicious	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.385031	0.60	0.40	0.565036	0.054403
1	pe-malicious	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.695652	0.20	0.20	0.372974	0.030327
2	pe-malicious	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	...	0.0	0.0	0.0	0.0	0.0	0.163088	1.00	1.00	0.979375	0.203325
3	pe-malicious	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.925532	0.25	0.25	0.648750	0.000407
4	pe-malicious	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	...	0.0	0.0	0.0	0.0	0.0	0.220399	1.00	1.00	0.979375	0.203325

5 rows × 487 columns

Fig. 9.18 Dataset screenshot

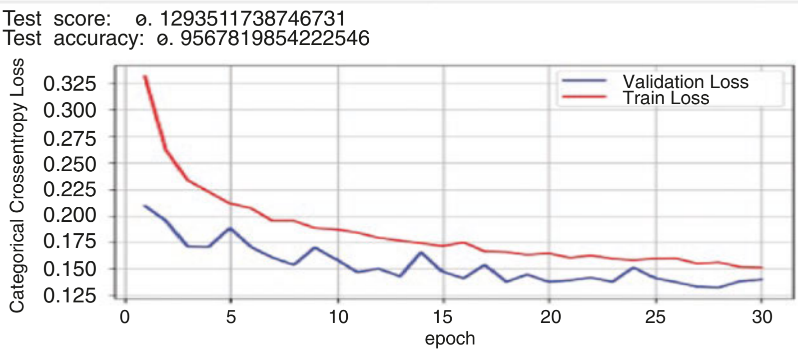


Fig. 9.19 Plot obtained for epoch vs Categorical Cross-Entropy loss

inverse side is that the yield units and this is often frequently where the organize reacts to the information that it had been given and handled (Fig. 9.19).

In arrange for ANNs to discover out, they have to have an extraordinary sum of information tossed at them called a preparing set. After you attempt to appear an ANN the way to distinguish a cat from a puppy, the preparing set would offer thousands of pictures labeled as a puppy in this manner the arrangement would start to be told. Once it is been prepared with a major sum of data, it will attempt and classify future information backed what it considers it is seeing (or hearing, wagering on the data set) all through the different units. Amid the preparing period, the machine's yield is compared to the human-given depiction of what ought to be watched. On the off chance that they are indistinguishable, the machine is approved. In the event that it is erroneous, its employments back proliferation to direct its learning—going back through the layers to change the numerical condition. alluded to as profound learning, this could be what makes a organize shrewdly.

6.2 ANN Model

We have then split the dataset into train and test and have applied the factitious neural network (ANN) on the dataset to urge a plan how the Deep learning model would work on malware analysis problems.

Fig. 9.20 ANN model used

Layer (type)	Output Shape	Param #
dense_22 (Dense)	(None, 70)	34090
batch_normalization_15 (Batc	(None, 70)	280
dropout_15 (Dropout)	(None, 70)	0
dense_23 (Dense)	(None, 70)	4970
batch_normalization_16 (Batc	(None, 70)	280
dropout_16 (Dropout)	(None, 70)	0
dense_24 (Dense)	(None, 2)	142

Total params: 39,762
Trainable params: 39,482
Non-trainable params: 280

The proposed model specifications are as follows (Fig. 9.20):

Our model performance seems to be pretty decent concerning the good applicability of deep learning models on malware analysis projects. The test accuracy that we got seems to be great. This shows that ANN is indeed useful in malware analysis problems. We will now last to explore a small amount further and try out turning malicious and benign files into images and applying a Convolutional neural network on top of it. Here we have got attached some of the samples where malicious and non-malicious files are transformed into images (Figs. 9.21 and 9.22).

They might look kind of like the oculus but we are sure that our CNN model is going to be able to find differences in both these image classes.

First, we have got taken only 256 bytes from each of our files. We have got to set that number to make sure that each one of the pictures that we get is of equal sizes no matter the parent application size. After that, we have got converted those bytes into 2D arrays. Finally, we converted those 2D arrays into images.

6.3 Convolutional Neural Networks

Convolutional neural systems are composed of numerous layers of manufactured neurons. Manufactured neurons, a harsh impersonation of their natural partners, are scientific capacities that calculate the weighted entirety of numerous inputs and yield enactment esteem. The conduct of each neuron is characterized by its weight. When nourished with the pixel values, the factitious neurons of a CNN select different visual highlights (Fig. 9.23).

Fig. 9.21 Malicious

Fig. 9.22 Non-Malicious

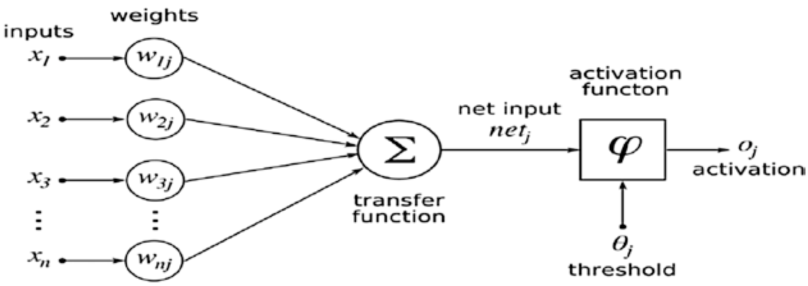


Fig. 9.23 CNN model illustration

6.4 CNN Model

The primary (or foot) layer of the CNN ordinarily recognizes essential highlights like even, vertical, and corner to corner edges. The yield of the essential layer is bolstered as input of the taking after layer, which extricates more complex highlights, like corners and combinations of edges. As you advance more profound into the convolutional neural arrange, the layers begin identifying higher level highlights like objects, faces, and more. Based on the enactment outline of the extreme convolution layer, the classification layer yields a gather of certainty scores (values

```
Epoch 20/20
7/7 [==============================] - 1s 163ms/step - loss: 0.0065 - acc: 1.0000 - val_loss: 0.2745 - val_acc: 0.9808
```

Fig. 9.24 CNN result

```
#index 0 is for non malicious files, 1 is for malicious files
loaded_model.predict(V1)
array([[0.94784486, 0.05215513]], dtype=float32)

#0 for non malicious files, 1 for malicious files
loaded_model. predict_classes (V1)
array ([0], dtype=int64)
```

Fig. 9.25 Applying our model on an unseen executable file

between and 1) that indicate how likely the picture is to have a place in a “class.” (Fig. 9.24)

The model performance seems to be great here, that too in precisely 20 epoch values. Finally, once we combine our models and test a brand-new file with it, we are able to determine whether the file is malicious or not. Here is a glimpse of it (Fig. 9.25).

In the case above, our model predicts it to be a non-malicious file. It absolutely was actually a secure file, so our model seems to be pretty decent.

7 Dynamic Analysis of Virus

To counter noxious assaults on computing frameworks want to identify malware as early as conceivable and halt it from executing its pernicious code. Whereas it is ordinarily simple to distinguish known malware, the foremost problem is dealing with an obscure coding system. to see whether an unknown/new executable is pernicious or not, it is common to utilize a master examiner who can survey the character of the executable. If the expert’s examination uncovers that the executable is malevolent, at that point a signature design (based on inactive, energetic, or cross-breed highlights) is frequently made to create location instruments with the inclination of identifying the assault inside long-standing time (counting comparative variations). In other words, examination turns “never seen some time recently malware” into a signature that can be recognized in the long run. Whereas manual investigation is amazingly solid, it’s not adaptable, not one or the other is it conceivable to utilize such investigation on each record much obliged to the overwhelming costs related to manual analysis.

Energetic examination alludes to the strategy of analyzing a code or script by executing it and watching its activities. These activities may be watched at different levels, from exceptionally cheap level conceivable (the code itself) to the framework as a full (e.g., changes made to the registry or record framework). the target of energetic examination is to appear the noxious movement performed by the executable whereas it is running, without compromising the security of the examination stage. From the protective viewpoint, there is a hazard of being contaminated by the malware whereas analyzing it powerfully, since it requires that the malware be stacked into the Slam and executed by the facilitating CPU.

Analyzing a suspicious record by inactive or energetic examination strategies can give important and profitable data with respect to a file's effect on the facilitating framework and offer assistance to decide whether the record is malevolent or not, backed by the method's predefined rules. Whereas different procedures (e.g., code muddling, energetic code stacking, encryption, and pressing) are frequently utilized by malware journalists to sidestep inactive investigation (counting signature-based antivirus apparatuses), the energetic examination is solid to those strategies and might give more noteworthy understanding with respect to the analyzed record and thus can result in way better discovery capabilities.

8 Implementation and Results

To see how srvcp.exe runs, we are able to infect the system with the specimen and observe it in Process Hacker's process listing. Behind the scene, our specimen attempted to read gus.ini within the System32 folder several times. We used a Process Monitor here to search out the precise location for infection. Initially, the method may fail, but after the thread is made, it will succeed. We are able to observe aspects of malicious processes through navigating in C:\Windows\System32\gus.ini. (Fig. 9.26)

Now we had an opportunity to look at how the malicious process interacts with the infected system. for example, we are able to see green (safe, i.e., it had been verified by the system before the infection) listens to TCP port 49476 but the red one was malicious, i.e., it absolutely was taking note of port 49473 after infecting the system. this is often one in all characteristics virus shows after successfully infecting the system (Fig. 9.27).

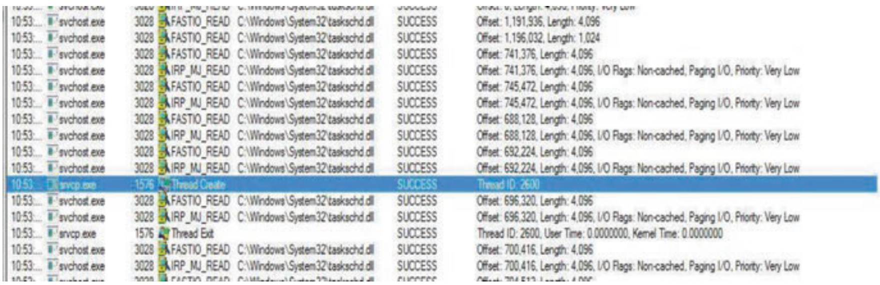


Fig. 9.26 Graphical View

Fig. 9.27 Port View

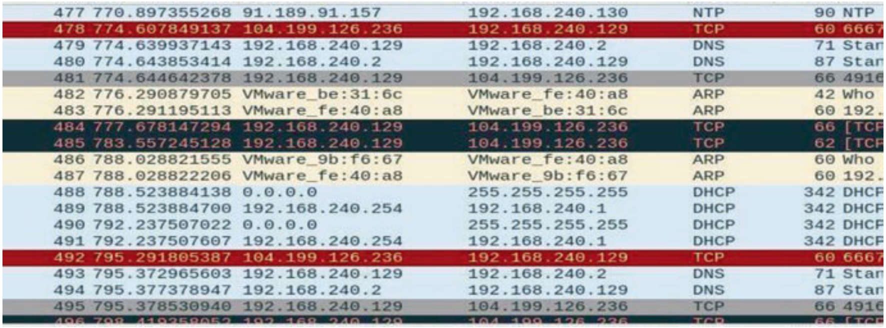


Fig. 9.28 Successful DNS query

In the case of srvcp.exe, we see our infected host(192.168.240.129) is communicating with 192.168.240.130(REMnux) on TCP port 80, sending a DNS query. Because the network is currently shared we get immediate responses (Fig. 9.28).

Below is shown, if our specimen is attempting to resolve hostname irc.mcs.net. The query is repeated after the initial request fails to induce a response because, at this stage of the analysis, we do not have a DNS server yet.

Analysis of srvcp.exe with Process Monitor and Process Hacker.

To see how srvcp.exe runs, we are able to infect the system with the specimen and observe it in Process Hacker's process listing. Behind the scene, our specimen attempted to read gus.ini within the System32 folder several times. We used a Process Monitor here to search out the precise location for infection. day by day nitially the method may fail but after the thread is made, it'll succeed. we are able to observe aspects of malicious processes through navigating in C:\Windows\System32\gus.ini. (Fig. 9.29)

Port Analysis with TCPview of srvcp.exe Now we had an opportunity to look at how the malicious process interacts with the infected system. for example, we are able to see green(safe, i.e., it had been verified by the system before infection) listens to TCP port 49476 but the red one was malicious, i.e., it absolutely was taking note of port 49473 after infecting the system. this is often one in all characteristics virus shows after successfully infecting the system (Fig. 9.30).

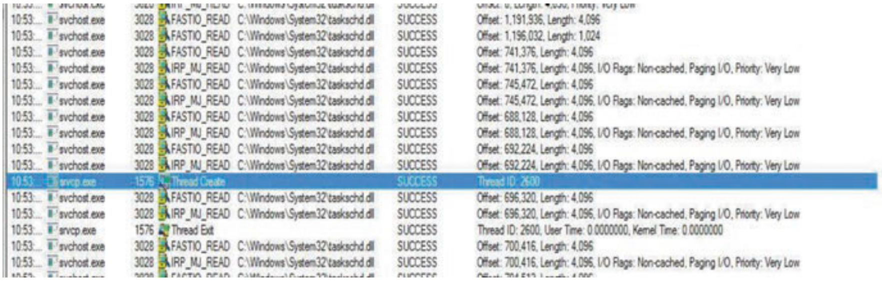


Fig. 9.29 Graphical view

Fig. 9.30 Port view

Wireshark: Infected system analysis In the case of srvcp.exe, we see our infected host(192.168.240.129) is communicating with 192.168.240.130(REMnux) on TCP port 80, sending a DNS query. Because the network is currently shared we get immediate responses (Fig. 9.31).

Below is shown, if our specimen is attempting to resolve hostname irc.mcs.net. The query is repeated after the initial request fails to induce a response because at this stage of the analysis we do not have a DNS server yet (Fig. 9.32).

Now, actual visualization is shown here. for every failed request, it attempts to attach to TCP port 6667. The server responded with an RST/ACK packet because it absolutely was not listening on the targeted port because this port is often used for IRC. It is trying to attach to Port 6667 with different ports(49183/84/85 and on) (Fig. 9.33).

This is monitored by a Network admin. They decided whether to just accept or deny those filters. for every red signals, they are notified and immediate actions are taken if required (Fig. 9.34).

Now, actual visualization is shown here. for every failed request, it attempts to attach to TCP port 6667. The server responded with an RST/ACK packet because it absolutely was not listening on the targeted port because this port is often used for IRC. It is trying to attach to Port 6667 with different ports (49183/84/85 and on) (Fig. 9.35).

This is monitored by the Network admin. They decided whether to just accept or deny those filters. For every red signal they are notified and immediate actions are taken if required.

477	770.897355268	91.189.91.157	192.168.240.130	NTP	90	NTP
478	774.607849137	104.199.126.236	192.168.240.129	TCP	60	6667
479	774.639937143	192.168.240.129	192.168.240.2	DNS	71	Stan
480	774.643853414	192.168.240.2	192.168.240.129	DNS	87	Stan
481	774.644642378	192.168.240.129	104.199.126.236	TCP	66	4916
482	776.290879705	VMware_be:31:6c	VMware_fe:40:a8	ARP	42	Who
483	776.291195113	VMware_fe:40:a8	VMware_be:31:6c	ARP	60	192.
484	777.678147294	192.168.240.129	104.199.126.236	TCP	66	[TCP
485	783.557245128	192.168.240.129	104.199.126.236	TCP	62	[TCP
486	788.028821555	VMware_9b:f6:67	VMware_fe:40:a8	ARP	60	Who
487	788.028822206	VMware_fe:40:a8	VMware_9b:f6:67	ARP	60	192.
488	788.523884138	0.0.0.0	255.255.255.255	DHCP	342	DHCP
489	788.523884700	192.168.240.254	192.168.240.1	DHCP	342	DHCP
490	792.237507022	0.0.0.0	255.255.255.255	DHCP	342	DHCP
491	792.237507607	192.168.240.254	192.168.240.1	DHCP	342	DHCP
492	795.291805387	104.199.126.236	192.168.240.129	TCP	60	6667
493	795.372965603	192.168.240.129	192.168.240.2	DNS	71	Stan
494	795.377378947	192.168.240.2	192.168.240.129	DNS	87	Stan
495	795.378530940	192.168.240.129	104.199.126.236	TCP	66	4916

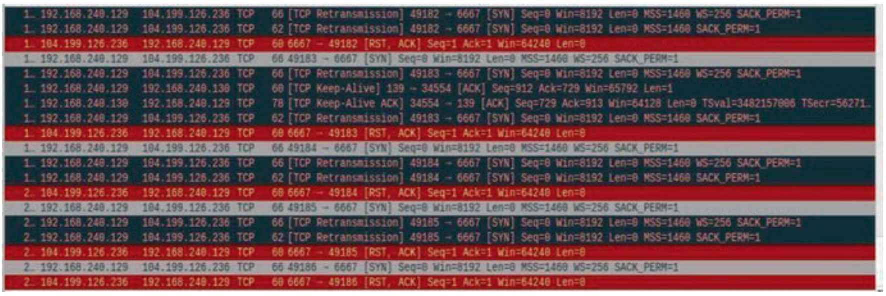
Fig. 9.31 Successful DNS query

No.	Time	Source	Destination	Protocol	Length	Info
1023	134.065196936	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1024	134.092919792	fe80::591c:9d5b:721…	fe80::91b3:9ad3:d3a…	SSDP	455	HTTP/1.1 200 OK
1025	134.231447467	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x6e2e A irc.mcs.net
1026	134.231494024	192.168.240.130	192.168.240.129	ICMP	99	Destination unreachable (Port unreachable)
1027	134.237540035	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1028	134.325839477	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1029	134.469785712	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1030	134.517302829	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1031	134.517563399	192.168.240.129	192.168.240.130	DNS	85	Standard query 0x5b73 A teredo.ipv6.microsoft.com
1032	134.612747841	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1033	134.925662398	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1034	135.050211801	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1035	135.190048664	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1036	135.378301661	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x7d5b A irc.mcs.net
1037	135.378370041	192.168.240.130	192.168.240.129	ICMP	99	Destination unreachable (Port unreachable)
1038	135.379220199	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1039	135.689181009	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1040	135.814902673	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x659c A irc.mcs.net

Fig. 9.32 Unsuccessful DNS query on Wireshark

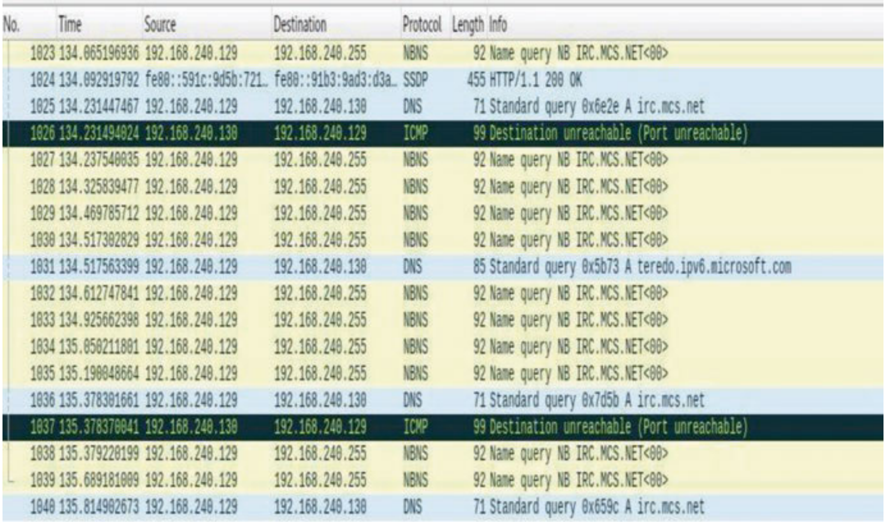
Start by importing srvcp.exe into IDA Pro. After successful importing, IDA Pro shows assembly instructions and automatically adds comments. In this code block, a function (push and call) is repeatedly called with different strings as parameters (Fig. 9.36).

Though the function in the original program probably had a meaningful name, that information was stripped away from the executable, which is why IDA Pro called it sub_4012C6. The actual work is deobfuscating the string supplied to it as a parameter. As expected, this function decoded the string supplied to it as the parameter by obtaining the length of the encoded string and iterating through it backward. While looping through characters of the encoded string, the function used the Boolean XOR operation to XOR the ordinal value of the character with the character's position from the right of the encoded string. This resulted in the deobfuscated versions of the string which the function returned when it finished executing.



1…	192.168.240.129	104.199.126.236	TCP	66	[TCP Retransmission] 49182 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1…	192.168.240.129	104.199.126.236	TCP	62	[TCP Retransmission] 49182 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
1…	104.199.126.236	192.168.240.129	TCP	60	6667 → 49182 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
1…	192.168.240.129	104.199.126.236	TCP	66	49183 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1…	192.168.240.129	104.199.126.236	TCP	66	[TCP Retransmission] 49183 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1…	192.168.240.129	192.168.240.130	TCP	60	[TCP Keep-Alive] 139 → 34554 [ACK] Seq=912 Ack=729 Win=65792 Len=1
1…	192.168.240.130	192.168.240.129	TCP	78	[TCP Keep-Alive ACK] 34554 → 139 [ACK] Seq=729 Ack=913 Win=64128 Len=0 TSval=3482157006 TSecr=56271…
1…	192.168.240.129	104.199.126.236	TCP	62	[TCP Retransmission] 49183 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
1…	104.199.126.236	192.168.240.129	TCP	60	6667 → 49183 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
1…	192.168.240.129	104.199.126.236	TCP	66	49184 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1…	192.168.240.129	104.199.126.236	TCP	66	[TCP Retransmission] 49184 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1…	192.168.240.129	104.199.126.236	TCP	62	[TCP Retransmission] 49184 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
2…	104.199.126.236	192.168.240.129	TCP	60	6667 → 49184 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
2…	192.168.240.129	104.199.126.236	TCP	66	49185 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2…	192.168.240.129	104.199.126.236	TCP	66	[TCP Retransmission] 49185 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2…	192.168.240.129	104.199.126.236	TCP	62	[TCP Retransmission] 49185 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
2…	104.199.126.236	192.168.240.129	TCP	60	6667 → 49185 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
2…	192.168.240.129	104.199.126.236	TCP	66	49186 → 6667 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
2…	104.199.126.236	192.168.240.129	TCP	60	6667 → 49186 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0

Fig. 9.33 Unwanted malicious requests



No.	Time	Source	Destination	Protocol	Length	Info
1023	134.065196936	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1024	134.092919792	fe80::591c:9d5b:721…	fe80::91b3:9ad3:d3a…	SSDP	455	HTTP/1.1 200 OK
1025	134.231447467	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x6e2e A irc.mcs.net
1026	134.231494024	192.168.240.130	192.168.240.129	ICMP	99	Destination unreachable (Port unreachable)
1027	134.237540035	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1028	134.325839477	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1029	134.469785712	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1030	134.517302829	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1031	134.517563399	192.168.240.129	192.168.240.130	DNS	85	Standard query 0x5b73 A teredo.ipv6.microsoft.com
1032	134.612747841	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1033	134.925662398	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1034	135.050211801	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1035	135.190048664	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1036	135.378301661	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x7d5b A irc.mcs.net
1037	135.378370041	192.168.240.130	192.168.240.129	ICMP	99	Destination unreachable (Port unreachable)
1038	135.379220199	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1039	135.689181009	192.168.240.129	192.168.240.255	NBNS	92	Name query NB IRC.MCS.NET<00>
1040	135.814902673	192.168.240.129	192.168.240.130	DNS	71	Standard query 0x659c A irc.mcs.net

Fig. 9.34 Unsuccessful DNS query on Wireshark

For each script run, the result of encoded and decoded values differs. So, it is difficult to visualize instantly. It can take considerable time to analyze it. But assembly is a low-level language, expect to encounter difficulties understanding the meaning of some of the more cryptic portions of the code thus we need OllyDbg to gain advanced level analysis (Fig. 9.37).

Above figure explains about the gus.ini process of srvcp.exe. Navigate to the fopen function and find references to this function. Pass it to the disassembler, which leads to location 40366A for this case. Srvcp.exe is about to open gus.ini when it reaches the breakpoint. The stack area shows parameters passed to fopen.

Parameter looks like this: "%[^=]=%[^'. Which is passed to sscanf using the push instruction at the address 4036CC. This causes parsing of string that follows the format of standard initialization files. Now a potential location where this

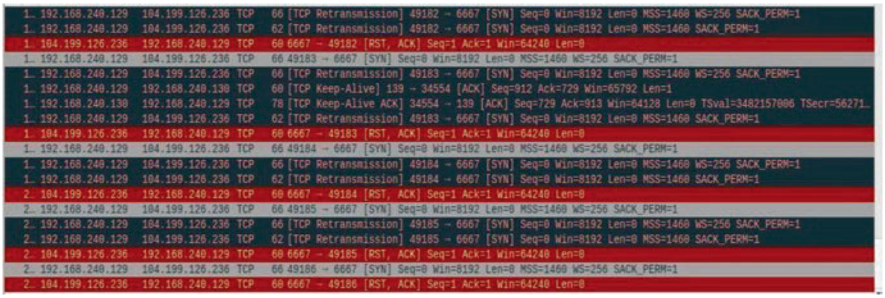


Fig. 9.35 Unwanted malicious requests

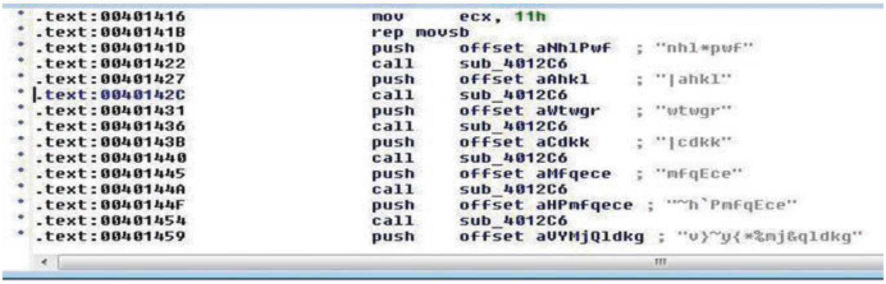


Fig. 9.36 Assembly code of srvcp.exe

transformation could occur is function 405366 which is at the address 4036BA because of a so-called sscanf function (Fig. 9.38).

After analyzing the above function and address location, the function typically returns the results of their computations. And is compared with the decoded value of gus.ini. which allows the attacker to adjust the specimen's configuration and the virus is injected. The code analysis is too vast to cover in this project so we utilized basic concepts on coding. For further explanation of the code, references are given below. To sum up, this is basic for dynamic analysis of viruses done manually.

We are analyzing worm.win32.Yaneth worm for this section.

As soon as the worm is loaded in pestudio, various sections indicating different attributes can be seen (Fig. 9.39).

On the VirusTotal section, it shows all antivirus signatures whether or not they consider the given exe is malicious or not. It is a less complicated version of online VirusTotal. Below is what the result feels like (Fig. 9.40).

One thing, needless to say, is that if the Microsoft section declares it as an endemic then it is obviously virus. because the environment on which it is executed is platform-dependent (Fig. 9.41).

The most important thing to appear for is Hashes for any exe files. this may provide plenty of knowledge like signature validation, certificates, their cryptic formations then on which may be accustomed verify their integrity. This can be very crucial to notice because all antiviruses that are out there declare any exe is

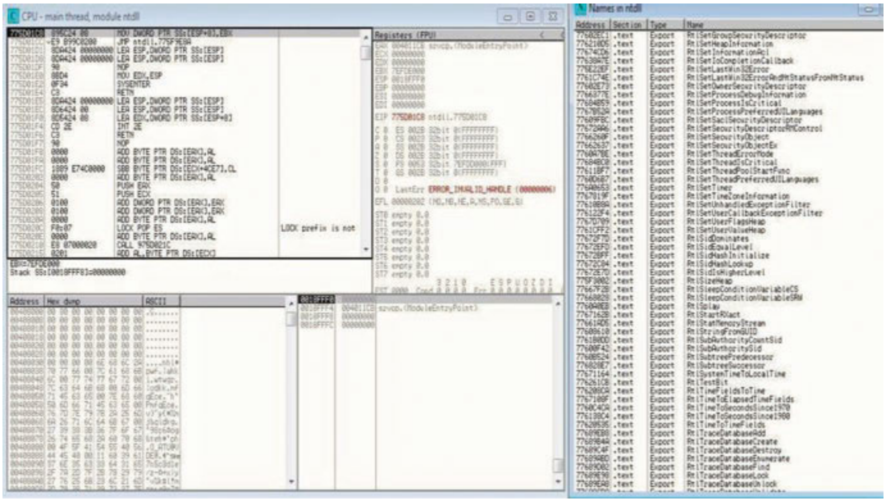


Fig. 9.37 Complete View of code

Fig. 9.38 Parameter Exploration

malicious or not on the idea of that information. All scanned viruses for this project hashes are given at the tip of the project.

Look into file info. There is a reputation UPX. What does that mean? Well, it does not mean it is executable instantly, instead, it is a container, i.e., it is visiting self-extract something and replicate within the system until it finds a target to where it had been meant to infect. to determine its working, it has to be able to run thus we will analyze it further, and here comes the importance of dynamic analysis (Fig. 9.42).

xml-id	indicator (28)	detail
1430	The file references string(s) tagged as blacklist	count: 3
1120	The file is scored by virustotal	score: 59/72
1265	The count of imports is suspicious	count: 4
1245	The file contains a blacklist section	section: UPX0
1245	The file contains a blacklist section	section: UPX1
1223	The first section is writable	section: UPX0
1225	The location of the entry-point is suspicious	section: UPX1:0x00007BE0
1631	The file contains self-modifying executable section(s)	status: yes
2215	The file contains writable and executable section(s)	count: 2
1262	The file imports anonymous function(s)	count: 1
1153	The file contains a virtualized section	section: UPX0
1019	The file contains a rich-header	status: yes
1424	The original name of the file has been found	name: Yaneth.exe
1036	The file checksum is invalid	checksum: 0x00000000
1634	The file references a group of API	api: dynamic-library, count: 2
1634	The file references a group of API	api: execution, count: 1
1633	The file references a group of hint	hint: dos-message, count: 1
1633	The file references a group of hint	hint: utility, count: 3
1633	The file references a group of hint	hint: file, count: 3
1050	The file uses Control Flow Guard (CFG) as software security defense	status: no
1100	The file opts for Data Execution Prevention (DEP) as software security defense	status: no
1102	The file opts for Address Space Layout Randomization (ASLR) as software security ...	status: no
1043	The file contains a Manifest	status: no
1106	The file opts for Stack Buffer Overrun Detection (GS) as software security defense	status: no
1040	The file contains a digital Certificate	status: no
1109	The file opts for Code Integrity (CI) a software security defense	status: no

0BDF0C30FA83B3CCF8272C78 cpu: 32-bit file-type: executable subsystem: GUI

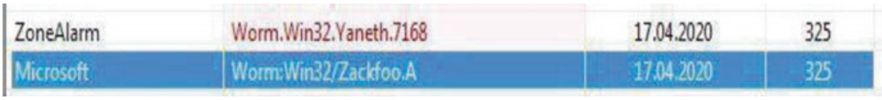
Fig. 9.39 PeStudio

engine (72/72)	score (59/72)	date (dd.mm.yyyy)
MicroWorld-eScan	Gen:Variant.Kazy.579832	17.04.2020
CMC	Generic.Win32.2dfd1aa991!MD	21.03.2019
CAT-QuickHeal	Worm.Zackfoo	17.04.2020
McAfee	Artemis!2DFD1AA99183	17.04.2020
Cylance	Unsafe	17.04.2020
Zillya	Worm.Yaneth.Win32.2	17.04.2020
K7AntiVirus	Trojan (0000cbb51)	07.04.2020
Alibaba	Worm:Win32/Yaneth.7e48960f	27.05.2019
K7GW	Trojan (004e93d71)	17.04.2020
Cybereason	malicious.99183f	16.06.2019
TrendMicro	Mal_Xed	17.04.2020
F-Prot	W32/Malware!5377	17.04.2020
TotalDefense	Win32/Zackfoo.C	17.04.2020
APEX	Malicious	16.04.2020
Avast	Win32:Yaneth [Wrm]	15.04.2020
ClamAV	Win.Worm.Yaneth-1	17.04.2020
Kaspersky	Worm.Win32.Yaneth.7168	17.04.2020
BitDefender	Gen:Variant.Kazy.579832	17.04.2020
NANO-Antivirus	Trojan.Win32.Yaneth.gxau	17.04.2020
Paloalto	generic.ml	17.04.2020
AegisLab	Worm.Win32.Yaneth.o!c	17.04.2020
Tencent	Win32.Worm.Yaneth.Hwnl	17.04.2020
Sophos	W32/Yanen-A	17.04.2020
Comodo	Worm.Win32.Yaneth.7168.A@1ysl	17.04.2020
F-Secure	Worm.WORM/Yaneth.7168.2	17.04.2020
DrWeb	Win32.HLLW.Zacker.24576	17.04.2020

Fig. 9.40 Worm Details

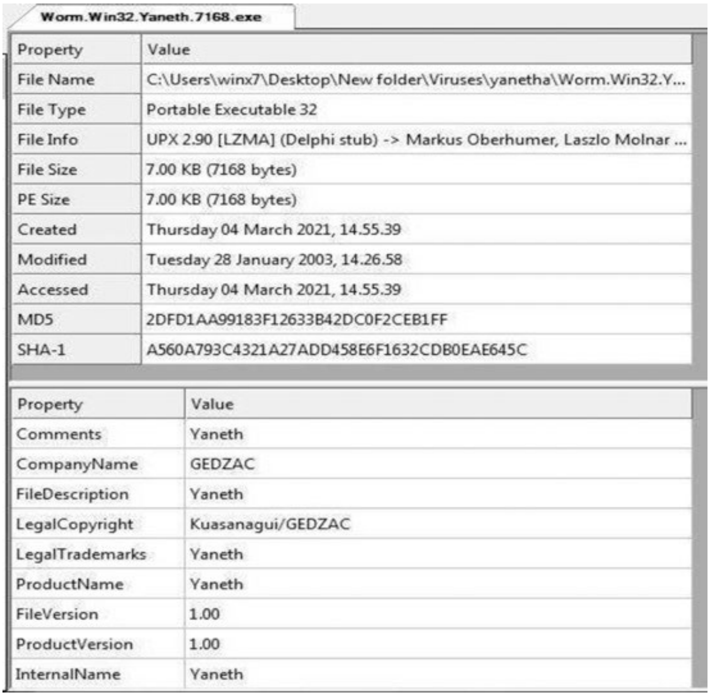
Now to investigate and visualize, start process hacker and process continues as above until the results of Wireshark. We are using the results directly here.

We are attempting to attach in tcp port 80 as requested by port 1060. needless to say, we get a reply from a client which indicated the virus has successfully injected, and on the second figure, because the request is looping an excessive amount of, Wireshark is applying filters to reject those requests. While worms try to still connect with other possible ports that most of them are going to be rejected. This makes systems suspicious and tries to eliminate them (Fig. 9.43).



ZoneAlarm	Worm.Win32.Yaneth.7168	17.04.2020	325
Microsoft	Worm:Win32/Zackfoo.A	17.04.2020	325

Fig. 9.41 Microsoft Warning



Worm.Win32.Yaneth.7168.exe

Property	Value
File Name	C:\Users\winx7\Desktop\New folder\Viruses\yanetha\Worm.Win32.Y...
File Type	Portable Executable 32
File Info	UPX 2.90 [LZMA] (Delphi stub) -> Markus Oberhumer, Laszlo Molnar ...
File Size	7.00 KB (7168 bytes)
PE Size	7.00 KB (7168 bytes)
Created	Thursday 04 March 2021, 14.55.39
Modified	Tuesday 28 January 2003, 14.26.58
Accessed	Thursday 04 March 2021, 14.55.39
MD5	2DFD1AA99183F12633B42DC0F2CEB1FF
SHA-1	A560A793C4321A27ADD458E6F1632CDB0EAE645C

Property	Value
Comments	Yaneth
CompanyName	GEDZAC
FileDescription	Yaneth
LegalCopyright	Kuasanagui/GEDZAC
LegalTrademarks	Yaneth
ProductName	Yaneth
FileVersion	1.00
ProductVersion	1.00
InternalName	Yaneth

Fig. 9.42 CFF Explorer

Now to verify their signature, both shots of Unicode must match, i.e., if they differ at any point it willl be confirmed something has been altered, tampered or changed. This tool is employed to shoot the system's registry to test if any system files are altered after virus/worms injection. We are taking two locations directly. one among the system32 files and another of yanetha worms (Fig. 9.44).

Now we add three files on both locations (face, MSN, snoopy) and execute them on any one of them (Fig. 9.45).

Look at the keys and value of both shots, they are changed means files are infected on both locations. Even a small change can make a large difference. This is often the explanation why hashes are a vital factor to search out the integrity of any files.

Now let us compare to urge more idea what has been changed here. Below is shown what number keys are altered. It is impossible to feature all changes so we use a straightforward one here (Fig. 9.46).

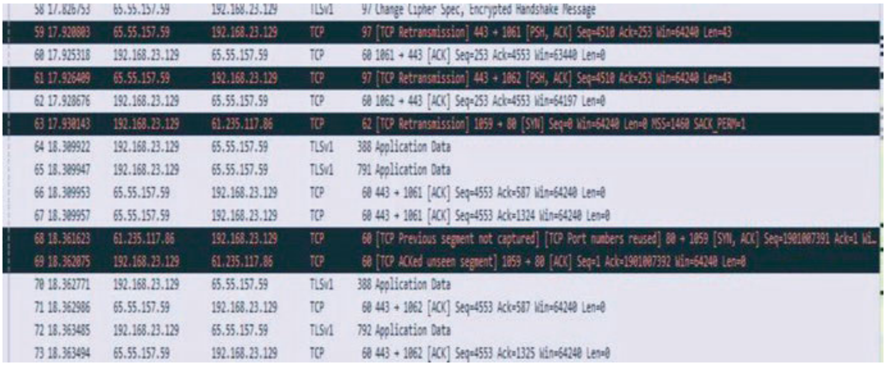
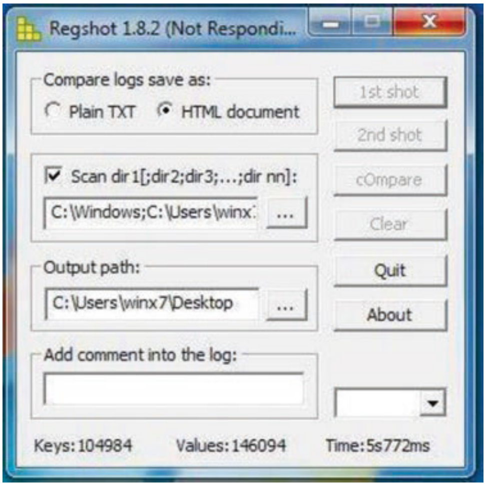


Fig. 9.43 Wireshark Analysis

Fig. 9.44 First Regshot



Below are the values. It indicates that original cryptic values are tampered and it targeted HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009\Counter. Target was system files, no wonder why Microsoft has marked this as a system virus which was shown in pe studio (Fig. 9.47).

Remember we had added some extra files before the primary shot and executed them during the second shot. Below is the quantity of files added and after executing what percentage of them are changed. this is often the most effective method to grasp whether any malicious items have been injected into the device and hence are utilized by many security and malware analysts (Fig. 9.48).

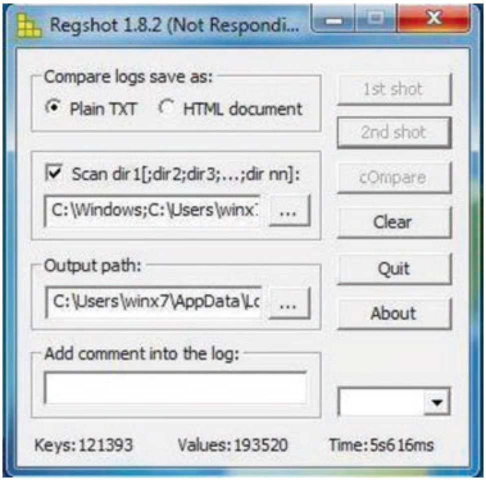
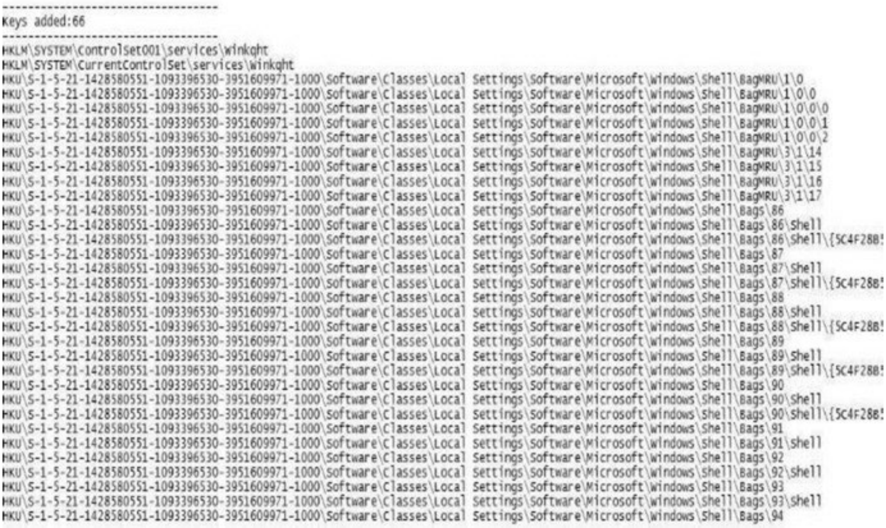


Fig. 9.45 Second Regshot



```
----------------------------------
Keys added:66
----------------------------------
HKLM\SYSTEM\ControlSet001\services\Winkqht
HKLM\SYSTEM\CurrentControlSet\services\Winkqht
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0\0
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0\1
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0\2
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\3\1\14
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\3\1\15
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\3\1\16
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\3\1\17
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\86
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\86\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\86\Shell\{5C4F28B5
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\87
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\87\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\87\Shell\{5C4F28B5
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\88
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\88\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\88\Shell\{5C4F28B5
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\89
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\89\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\89\Shell\{5C4F28B5
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\90
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\90\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\90\Shell\{5C4F28B5
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\91
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\91\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\92
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\92\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\93
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\93\Shell
HKU\S-1-5-21-1428580551-1093396530-3951609971-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell\Bags\94
```

Fig. 9.46 Alteration Study

9 Conclusion and Future Work

From the above research, we came to a conclusion that in malware analysis if the static and hybrid analysis is done together provides a much better understanding of how the malware is developed and works which can be used to develop systems to detect the malware and prevent it from infecting any systems.

Fig. 9.47 Hashes Alteration

Fig. 9.48 Pre/Post Analysis

References

1. Internet Security Threat Report, Volume 22, Symantec (April 2017)
2. Yin, H., & Song, D. (2013). Automatic malware analysis: An emulator based approach. *Springer-Briefs in Computer Science*. https://doi.org/10.1007/978-1-4614-5523-37
3. Salehi, Z., Ghiasi, M., & Sami, A. (May 2012). A miner for malware detection based on API functioncalls and their arguments, In: Artificial Intelligence and Signal Processing (AISP), 16th CSI International Symposium on, pp. 563–568.
4. Uppal, D., Sinha, R., Mehra, V., & Jain, V. (September 2014). Malware detection and classification based onextraction of api sequences, In: International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2337–2342.
5. Tian, R., Islam, R., Batten, L., & Versteeg, S. (2010). Differentiating malware from cleanware using behavioural analysis, Malicious and Unwanted Software (MALWARE). *5th International Conference, 5*(5), 23–30.

6. Distler, D. (December 14, 2007). *Malware analysis: An introduction*, SANS Institute.
7. Ahmadi, M., Dmitry, U., Stanislav, S., Mikhail, T., & Giorgio, G. (2016). Novel feature extraction, selection and fusion for effective malware family classification. In *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy* (pp. 183–194). ACM.
8. Kohavi, R. (1995). *The power of decision tables* (pp. 174–189). Machine learning: ECML-95.
9. Kawaguchi, N., & Omote, K. (2015). Malware function classification using APIs in initial behavior. In *Information Security (AsiaJCIS), 10th Asia Joint Conference* (pp. 138–144). IEEE.
10. Qi, Y. Random Forest for bioinformatics, http://www.cs.cmu.edu/
11. Hansen, S. S., Thor Mark Tampus, L., Matija, S., & Jens Myrup, P. (2016). An approach fordetection and family classification of malware based on behavioral analysis. In *Computing, Networking and Communications (ICNC), International Conference* (pp. 1–5). IEEE.
12. Hong, J., Park, S., & Kim, S. W. (2016). On exploiting static and dynamic features in malware classification. In *International Conference on Big Data Technologies and Applications* (pp. 122–129). Springer. 2016 Nov 17.
13. Ranveer, S., & Hiray, S. (2015 Jan 1). Comparative analysis of feature extraction methods of malware detection. *International Journal of Computer Applications, 120*(5).
14. Pirscoveanu, R. S., Steven Hansen, S., Thor, M. T. L., Matija, S., Jens Myrup, P., & Alexandre, C. (2015). Analysis of malware behavior: Type classification using machine learning. In *Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), International Conference* (pp. 1–7). IEEE.
15. Gupta, S., Sharma, H., & Kaur, S. (2016 Dec 14). Malware characterization using windows API calls sequences. In *International Conference on Security, Privacy, and Applied Cryptography Engineering* (pp. 271–280). Springer.

Chapter 10
Automated Weather Monitoring Station Based on IoT for Smart Cities

Shaifali M. Arora and Mishti Gautam

Abstract In everyday life, weather conditions play a major role. Collection, monitoring, and analysis of data about the different parameters of the weather are necessary to plan various activities in day-to-day life. The weather conditions are required to be monitored for numerous reasons, like the dependency of agriculture, aerial, and marine transport services on the weather; detection of air quality and particulate matter for the health of humans and the environment; to ensure a safe working environment in industries; to predict and forecast climatic phenomena, etc. For ages due to the unavailability of accurate forecast data, due to irregular measurement and analysis of weather conditions, many natural calamities and disasters take place resulting in the loss of millions of lives. Data collected from satellites is for a larger geographical area and not for a pinpointed area at the ground level let us say for a city and therefore in some cases some mismatching of data may occur. If data from the ground level are provided as an added aid to the Meteorological department along with the data collected through various other means to perform analysis there are chances of better prediction and forecasting of natural phenomena. The dissertation is a solution to overcome these limitations.

In today's world, some major areas of application of a smart, real-time, efficient, low-cost, accurate, low-power, portable, high-speed, Internet of Things (IoT)-based Weather Monitoring station are: Airport operations, Coastal area weather detection, Construction of high rising structures, Agricultural greenhouse, and warehouse condition monitoring, Air Pollution, Solar-based technological industries. Therefore, the need of the hour is to design an Automated Weather Monitoring System which will enable enhanced data collection in real-time for different parameters, such as light intensity, humidity, temperature, air quality, and wind speed without the intervention of humans. This Weather Monitoring Station circuit will be designed to provide an automatic monitoring mechanism to authorities and for the people who pass by the location at which the station is installed. For this purpose, competitive strategy tools and equipment are required to design hardware to fetch the required data and provide it for analysis.

S. M. Arora (✉) · M. Gautam
Maharaja Surajmal Institute of Technology, Janakpuri, New Delhi, India
e-mail: shaifali04@msit.in

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_10

This chapter describes the model designed for this automatic monitoring, and the main component used in this model is Raspberry Pi, which will control all the sensors and upload the real-time data collected using the sensors to a cloud and display the same on an LCD screen/panel installed onsite. The Weather Monitoring Station prototype uses a Light intensity sensor (LDR), Anemometer, Temperature sensor, and Air Quality detection sensor, the continuous analog readings of which will be converted to digital using an ADC IC MCP3208.

1 Introduction

Weather conditions have a very important part in everyday life. Monitoring of weather is very important for predicting the present climate as well as observing and finding changes in climate to generate data to predict future changes in our environment. The current scenario for monitoring national air quality have limited scope because of the time lag in recording and reporting of the data that results in a lag in planning for real-time solution. Also, Data collected from satellites is for a larger geographical area and not for a particular pinpointed area at the ground level and therefore in some cases some mismatching of data may occur. If data from the ground level are provided as an added aid to the Meteorological department along with the data collected through various other means to perform analysis there are chances of better prediction and forecasting of natural phenomena, that is where this research comes into play. Also, the weather monitoring system that is adopted traditionally does not take measurements in real-time, and recordings are done at set intervals. But the scenario has changed with the deployment of IoT for measurements. This enabled the system to collect real-time data and that too continually, so now the plan of action can be implemented without delay based upon the current or real-time data. [4–6]

The research will help the civic authorities and local people to keep check on the atmospheric condition. A model “Automated Weather Monitoring System based on IoT” has been presented in this chapter**.** The main component used in this model is the Raspberry Pi 3 B+, which controls all the sensors and uploads the real-time data collected using the sensors to a cloud and displays it on an LCD for the local people near the areas to see so that they can take precautions if they observe any change in atmospheric conditions from the ideal conditions. The Raspberry Pi 3 is a microcomputer that will control all the major tasks in the proposed circuit. The sensors used in this model are Anemometer, Air Quality Index level detection sensor (MQ135), Temperature sensor (DHT11), and Light intensity sensor (LDR).

2 Literature Survey

Weather monitoring is not something new. The weather sensors have been used for decades. But the only problem was that the data couldn’t be collected continually and could be reported at central weather monitoring stations continually. But with the development of cloud computing, IoT, and Electronics now such tedious tasks of monitoring weather are simplified. Monitoring of temperature manually to now automatically has become possible because of advanced sensors. This has given a new direction in temperature monitoring weather for the environment or healthcare. Researchers are continually working in the field to understand the climatic changes before any tragedies take place.

Weather monitoring started many years back. Urban et al. [6] analyzed the aspects of different weather types by comparing the results of measurements of air temperature obtained using automatic weather stations (AWS) to the measurements obtained from glass thermometers.

Saktaya et al. [8] proposed the Artificial Neural Networks (ANN)-based approach for forecasting the weather. The researchers adopted supervised learning NN-based model and used a training algorithm to adjust the bias and weights of various neurons in the network. In this research, the researchers were able to predict the temperatures by using ANN and data from various Meteorological Stations.

Sushmitha [9] proposed a model in which the temperature and humidity data of a particular region/ location has been collected using the temperature and humidity sensors. The collected data are transmitted to the cloud using Raspberry pi. The data which are stored in the cloud are transformed into various types of files for further processing and analysis of weather conditions. The correlation finding approach is adopted for analysis and future predictions. The ARIMA model has been adopted in this research.

Dolara [12] et al presented the expansion of forecast models for a wind farm reducibility with a 24 h horizon with an aim to obtain accurate wind power predictions by using feed-forward artificial neural networks. Different forecasting models have been developed and presented in this work and for each of them, the best architecture is researched by means of sensitivity analysis, modifying the main parameters of the artificial neural network.

J Cowie [15] in his research paper highlighted the use of machine learning for weather forecasting by NCAR. The first automated engine designed for forecasting weather was Dynamic Integrated forecasting (DICast®) Many companies are still using it for many applications. Surface transportation, wildland fire forecasting, and artificial intelligence are some of the applications used by NCAR where DICast has been included.

Work presented in [7, 10, 11, 14] is toward the weather prediction model for smart cities. Kathiravan et al [19] proposed a javascript framework based on a web application named AngularJS for the forecasting of the weather of any area, any city in the world.

N Ahmed et al. [21] proposed an economic and portable mini-weather station, which can be marketable as a local device and also ensures scalability, has combo features, and is convenient for Bangladeshi farmers as well as for developing counties. In this project, microcontroller Arduino Uno R3 has been used to control the weather station. To read analog data from various sensors like UV Index and UV-A Lamp Monitoring using UV-B sensor, measuring soil moisture using moisture sensor, DS1820 sensor, air temperature and air humidity using sensor DHT11 sensor, MQ2 Gas Sensor, using HC-05 Bluetooth device, and Barometric Pressure BMP085 is done. The main feature of this station is that it does not need any internet, except a self-generated Bluetooth network is required. Considering various seasonal weather in Bangladesh, a mini-weather station has been designed that accurately measures all parameters from local weather status.

Nitin et al [23] proposed a forecasting system for weather based on machine learning. To overcome the difficulties of the weather forecast, there is a need for improvement in weather prediction methods. Nation's economy and the living conditions of people their lives are affected to a great extent by these predictions. In this work, machine learning algorithms and data analytics are used to predict weather conditions. The main attraction of this work is that a cost-effective solution had been proposed.

Anubha et al [24] proposed an IoT-based model for automated weather report generation and prediction system that used a machine learning approach. The main techniques used in this work are machine learning, artificial intelligence; feature extraction; Internet of Things; image classification; Internet; image segmentation; healthcare; medical image processing; cloud computing; and social networking (online).

Sai Yeshwanth et al [26] asserted a solution to a traditional weather monitoring system that used the latest technologies like embedded systems, cloud computing, and internet of things to improve efficiency by decreasing its usage. The main feature of this project is that this project has been successfully used and implemented in Neemrana, Rajasthan.

Neha Kumari et al [27] proposed an IoT-based weather monitoring system in which real-time weather parameters are collected and then these data are stored in a cloud. A specific webpage is designed to display the collected data. These stored data are further used for weather forecasting. The weather parameters that were used in this project were precipitation, temperature, dew point, humidity, light intensity, smoke percentage, and air pressure.

Kalvin et al [28] proposed a model for monitoring climatic conditions of a particular place. In his technique, weather parameters like CO Level using sensors, temperature, and humidity are measured. Sensors perceive changes in environmental conditions and send them to the end-users for making different types of analysis. To connect this system to the whole world IoT is used.

Vaishnavi et al [29] presented a weather monitoring system that is based on Arduino, the proposed model basically monitors the weather parameters of the environment. The proposed model mechanically collects the data for climatic changes in the environment. The paper technique focused on the weather conditions and is able to forecast the weather with no human error.

Vorasit et al [30] fabricated an imaging system using Raspberry Pi-based. A microclimate sensor has been integrated into the system to estimate the growth of the crop in a particular spring. This system has been designed and programmed in such a way that it captures the crop images automatically for all seasons. The captured images were helpful for extracting various traits of a plant, and this way with various weather conditions these effects can be analyzed.

Ashwitha et al. [31] in their work proposed recording the pertinent data, its classification with proper visualization techniques perceivable everywhere in the globe. IoT has been associated with the proposed technique as it is a progressive approach to associate objects and connecting the whole world of objects to the internet. K-nearest neighbor classification algorithm has been used in the proposed technique for performing the processing on real-time data which is obtained from the various sensors connected for collecting real-time data and to generate practical prediction mechanism. Natural parameters such as atmospheric temperature, humidness, and wind pressure have been continually recorded to make wide data analytics and to draw helpful predictive decisions.

Y. NarsimhaRao et al [32] proposed a weather reporting system using IoT that consists of multiple devices for recording data for temperature, humidity, and air pollution. In the proposed model, parameters related to the environment, such as temperature, rainfall, the density of carbon dioxide, and humidity in the air, are measured with sensors. All the information from sensors is collected on the Arduino Uno processor and sent to the cloud for further processing. Also, data security issues on the cloud are addressed.

D. D. Khandelwal [33] proposed a monitoring system for weather that used both open-source software and hardware. The main aim of the proposed work was to collect environmental data from the remote territory of the Himalayas and that too at a very low cost.

Based upon the extensive literature survey a model has been proposed that uses Raspberry Pi as a processor and transfers live data to the cloud.

3 Proposed Model

Raspberry Pi 3 has been used in the prototype of Automated Weather Monitoring Station based on IoT that acts as the brain of this model, data from all the sensors is provided to it. Based upon this data, it keeps checking on all the changes and activities in the atmosphere. All the sensors are connected to the Raspberry pi using the GPIO pins of the microcontroller. These sensors sense the required information and pass it back to the Raspberry pi from where the Raspberry pi displays it on the 16*2 LCD Display and send the data to cloud platform Thingspeak.com. Block Diagram of the proposed model is presented in Fig. 10.1, the connection of all the sensors and the corresponding output is shown here.

3.1 Raspberry Pi 3 B+

Raspberry Pi [34] is a small computer on a single board at low power consumption and a cheaper price. If peripherals like a mouse, Keyboard, display are connected to this small board, it will function as a mini PC. This processor is widely used for

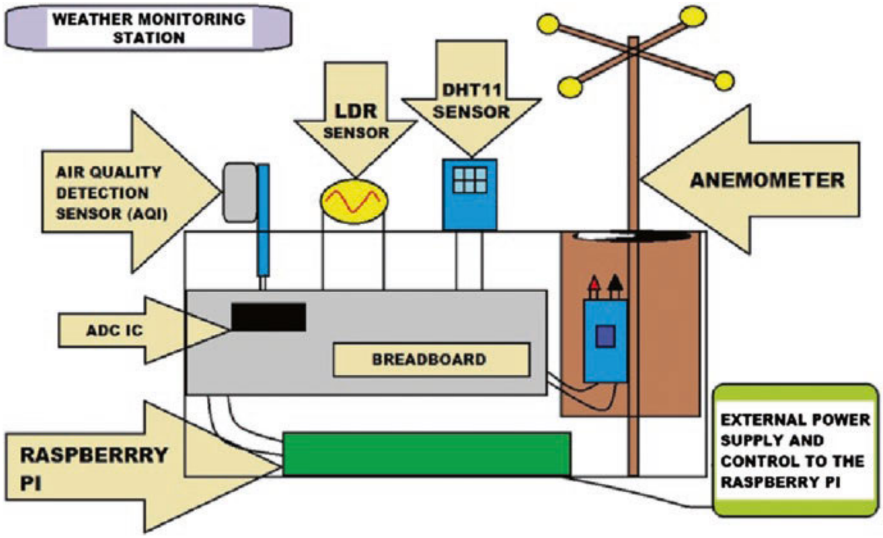


Fig. 10.1 Model of automated weather monitoring station

applications like IoT-based applications, real-time Image/Video Processing-based applications. and Robotics applications. The only drawback of Raspberry Pi is that it is slower in comparison to a laptop or desktop.

Raspberry Pi supports Debian-based Raspbian OS and NOOBS OS which is efficiently optimized for use and has GUI. It also includes various tools for Browsing, office, Python programming, games playing and development, etc.

Main features of the Raspberry Pi 3 that is used in our model are—it is a 64-bit quad-core processor running at 1.4GHz, Bluetooth 4.2/BLE, dual-band 2.4GHz and 5GHz wireless LAN, PoE capability via a separate PoE HAT, and faster Ethernet. The dual-band wireless LAN on the board allows the board to be designed into end products with low cost and early time to market.

Raspberry Pi is a compact computer, it provides easy access to GPIOs and other on-chip hardware for developing an application. Ease of access of GPIOs makes it convenient to connect and control devices like motors, LED, sensors, etc.

Other features of Raspberry Pi 3 are- on-chip GPU, onboard SDRAM which varies from 256 MB to 1 GB, SPI, I2C, I2S, and UART-and ARM-based Broadcom Processor. The speed of the CPU of Raspberry Pi may vary from 700 MHz to 1.2 GHz. Raspberry Pi provides on-chip modules (Fig. 10.2).

Raspberry Pi-3 is used as the main component, the controller, in this research. It is programmed to enable all the sensors and collect real-time data from them to upload on the cloud platform and display it on the 16*2 LCD display. Using the information that is appearing on the cloud platform and LCD, the Meteorological department can predict more accurate weather forecasts and it will also help people living nearby the System to keep a check on the various atmospheric

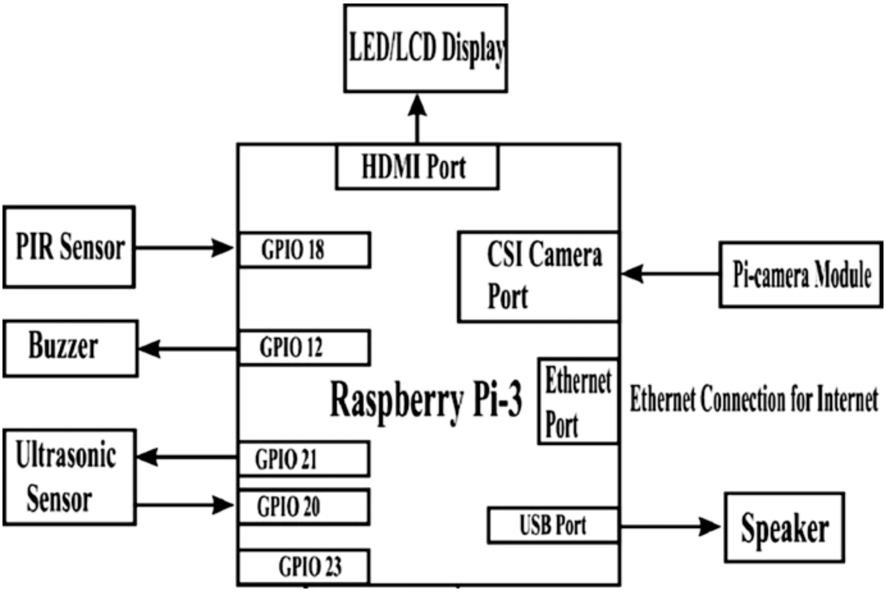


Fig. 10.2 Block diagram of Raspberry Pi

conditions which may affect them like AQI, wind speed, sunlight intensity, temperature, and humidity.

3.2 ADC MCP3208 IC

ADC MCP3208 IC is a 12-bit Analog-to-Digital (A/D) Converter with a sample and holds circuitry onboard. The MCP3208 is a programmable A/D converter that provides either eight single-ended inputs or four pseudo-differential input pairs. In this ±1 LSB Differential Nonlinearity (DNL) has been specified. It can communicate with other devices by using a simple serial comm. interface which is compatible with the SPI protocol. The voltage range over which the IC can operate is 2.7V–5.5V.

Here, ADC MCP-3208 is used to convert the analog output of the MQ-135 AQI Sensor to digital so that it can be easily read by the Raspberry Pi and provide an accurate reading on the cloud platform and LCD display, which will be accessible by the civic meteorological department and locals, respectively.

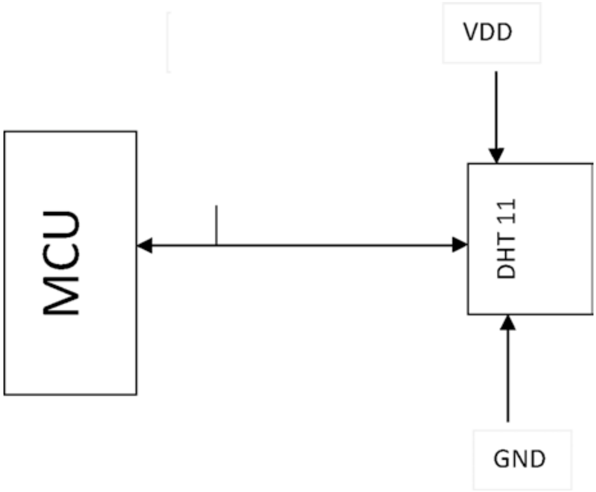


Fig. 10.3 Connection diagram of DHT

3.3 DHT-11 Temperature and Humidity Sensor

DHT11 is a widely used sensor that gives a digital signal output of the temperature and humidity. Digital modules are used in the sensor to impound the temperature and humidity. It provides excellent stability and high reliability. A high-performance 8-bit microcontroller, NTC temperature measurement device are connected on chip. The DHT11 sensor can be easily interfaced with various microcontrollers (Fig. 10.3).

The output is provided in serial. The interfacing or connection diagram of DHT11 with MCU, there are open-source libraries available for it that can be readily used.

In this model, DHT-11 Sensor (Temperature and Humidity Sensor) is used to measure the temperature and humidity of the atmosphere nearby where the weather station is installed with the help of the pre-built libraries downloaded from Adafruit.

3.4 Anemometer

An anemometer is a device that is used for the measurement of wind speed and air pressure. These are important instruments for the study of weather patterns.

In an anemometer, there is a vertical rod to which three or four cups are attached. When the wind blows, the cups rotate, and because of which the rod spins. The speed of the spinning of the rod depends on the speed of the wind. The anemometer counts the number of rotations of the rod which is therefore used to estimate the wind speed. Wind speed is usually averaged over a short period of time because of gusts and lulls.

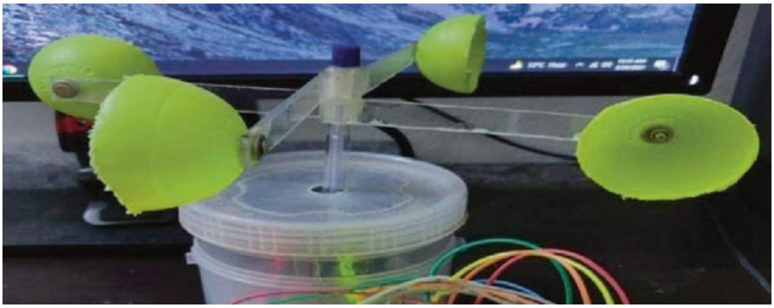


Fig. 10.4 Self-made Anemometer

Anemometers are most widely used in all the weather monitoring stations, from warm equatorial regions to the frigid Arctic. The anemometer used in this model has been made from scratch from materials like balls, an old pen, an old box, and the main electronic component which has been used in the anemometer is an IR sensor which helps in counting the number of rotations the anemometer has made to calculate the wind speed (Fig. 10.4).

3.4.1 IR Sensor

There are two types of IR sensors: Active and Passive. In our model, active sensors are used. In the case of active sensors, infrared radiation is both emitted and detected at one end only these sensors have two parts: one is a light-emitting diode (LED) and the other is a receiver. When an object approaches the IR sensor, the IR light transmitted from the LED is reflected back from the object and then this reflected light is detected by the receiver LED. Active IR sensors behave as proximity sensors, and they are most widely for the detection of obstacles.

3.5 MQ-135 Sensor

MQ series Gas sensors are the most appropriate sensors for measuring or detecting any particular gas. These sensors are cheap and easily available. **MQ135** is the most widely used gas sensor, which is available just as the sensor or as a module both. If the purpose is to only measure PPM, gas a module can be used because it has on-chip op-amp comparator and a digital output pin.

In the proposed model, we have used the MQ135 to check the Air Quality Index and display the PM levels to the locals and meteorological department which is helpful for everyone as it is the greatest concern nowadays as pollution levels are rising with each passing day.

3.6 Light Intensity Sensor

LDR is Light-Dependent Resistor. To make a light intensity sensor, an LDR sensor is used. The value of resistance of LDR changes with the light. When the intensity of light is high, then the value of resistance is low, and in darkness or low intensity, the value of resistance is high, in the range of Mega-ohms.

The Light-Dependent Resistor (LDR) is a special type of Resistor, it is polarity independent and can be connected in any direction.

In the presented model, the LDR sensor is used for light intensity detection by measuring the intensity of sunlight.

3.7 LCD Display

LCD modules are very commonly used in most embedded projects because of their cheap price, availability, and programmer-friendly nature.

3.8 ThingSpeak Online Cloud Platform

ThingSpeak is an open-source application designed for the IoT. API is used to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a LAN. ThingSpeak enables the creation of sensor logging applications, location tracking applications, and a social network of things with status updates.

The important feature of ThingSpeak is it can be integrated with various already available software's like MATLAB. The main advantage of ThingSpeak is that uploaded data can be easily analyzed, visualized using MATLAB without the need of purchasing a license. With ThingSpeak live data can be aggregated, visualized, and analyzed on the cloud.

4 Working of Presented Model

To verify whether the hypothesis was working or not a prototype model was prepared. Working the whole model is not on the complex side it is rather easy. Primarily, when the power is supplied to the circuit Raspberry Pi turns on and the code is run. Then it provides the required voltage and current to the different sensors for their effective and proper working. As all the sensors get turned on, every sensor individually starts working and detecting the weather parameters and conditions. Thereafter, as soon as it detects the environmental condition it sends that real-time data to the Raspberry pi and after calculations, it displays the different readings on the LCD as well as on the cloud platform in real-time. The Anemometer starts rotating when the wind blows and the raspberry starts to count the number of rotations using the IR Sensor. It then calculates the wind speed and displays it.

DHT11 temperature and humidity sensor senses the temperature and the humidity and sends data to the raspberry pi for further computations. MQ135 Gas sensor is the sensor that requires an analog to digital converter IC as this sensor provides the output in analog format. Therefore, to read the data from it we require ADC MCP3208 IC to convert the analog reading from the MQ135 to digital and send it to the raspberry pi, and then it displays it on the LCD display and the cloud platform.

Light Intensity Sensor uses an LDR sensor whose resistance value is low or negligible when the intensity of light is high and resistance is high in the range of Mega-ohms when there is darkness or the intensity of light is low. These data are given to the Raspberry Pi which converts it into the reading of Sunlight intensity and displays it on the LCD display and on the cloud platform (Figs. 10.5, 10.6, 10.7, 10.8, 10.9, 10.10 and 10.11).

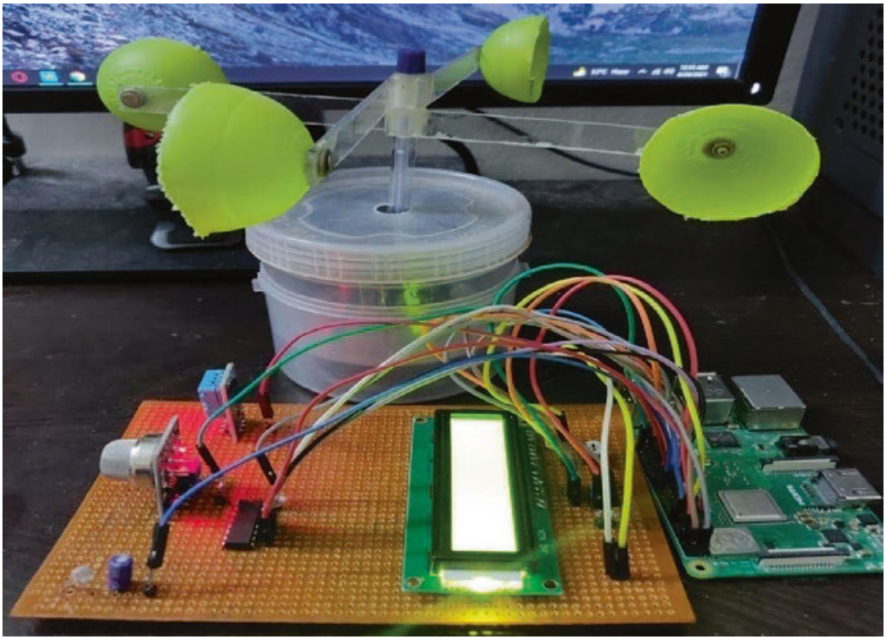


Fig. 10.5 Prototype model

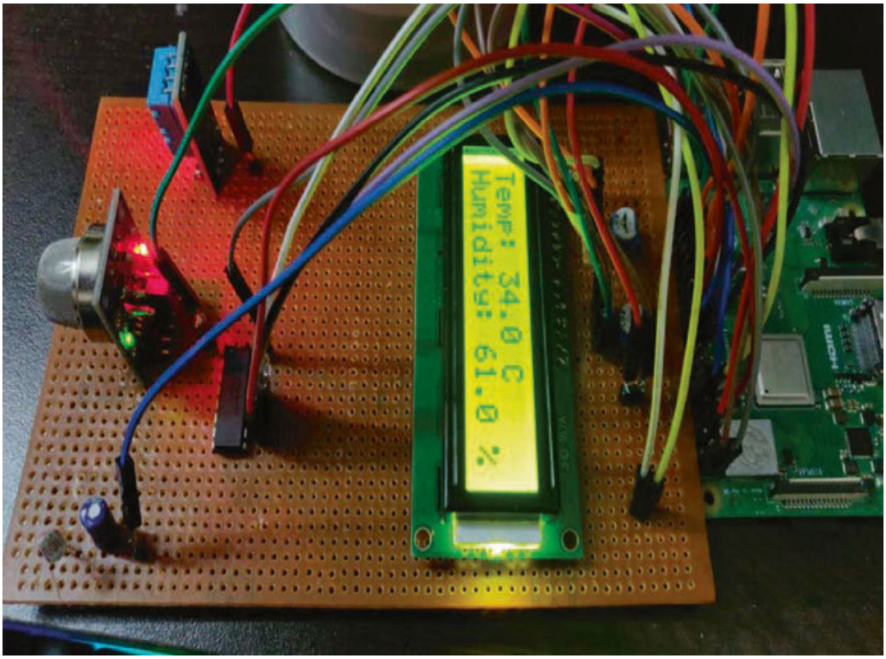


Fig. 10.6 Prototype model

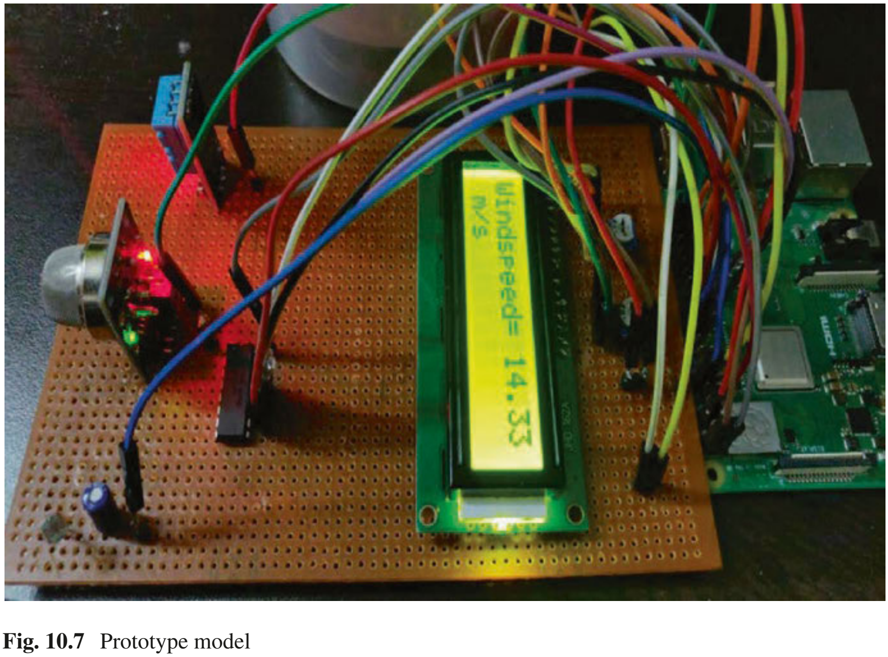


Fig. 10.7 Prototype model

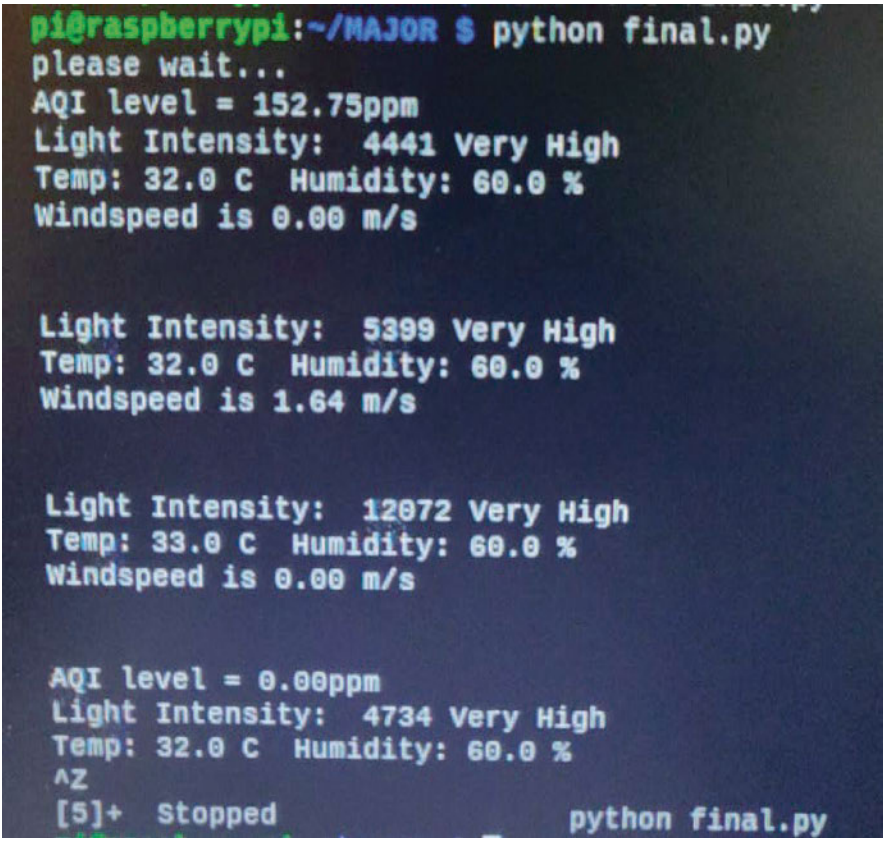


Fig. 10.8 Output of Raspberry Pi

5 Future Scope

The research hypothesis is currently limited to the sensors and raspberry pi. It has a huge dependency on internet connection and therefore to limit this dependency of external internet connection we can include a Wi-Fi module that can provide the raspberry pi with permanent internet.

In addition to this, there is room for improvement using machine learning technologies since from the ThingSpeak platform one can export the data uploaded on it in an excel file consisting of all the real-time data. It can be converted to .csv files or downloaded in .csv format and used for analysis, prediction, and forecasting of weather conditions. In the future, it may even be possible to include this feature in our project and use it to predict any mishaps before them happening.

It also has a future scope of advancement where we can upload the data on cloud and access it back using a mobile application so that without opening the cloud

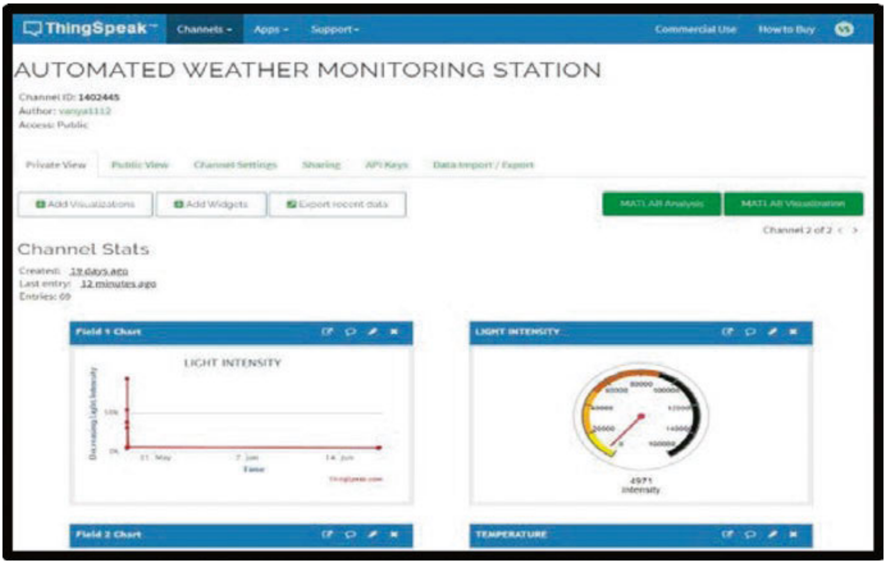


Fig. 10.9 Output at cloud platform

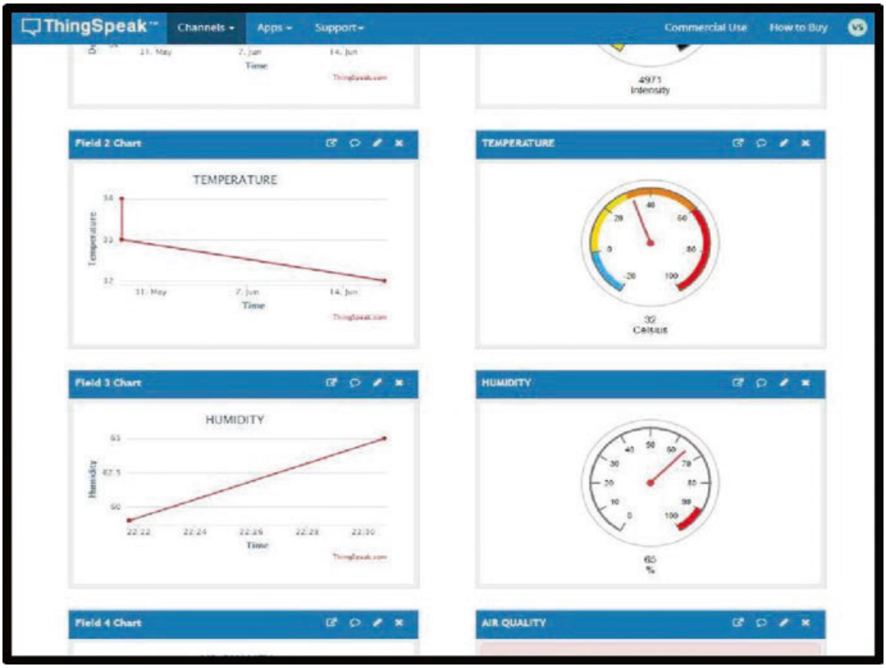


Fig. 10.10 Output at cloud platform

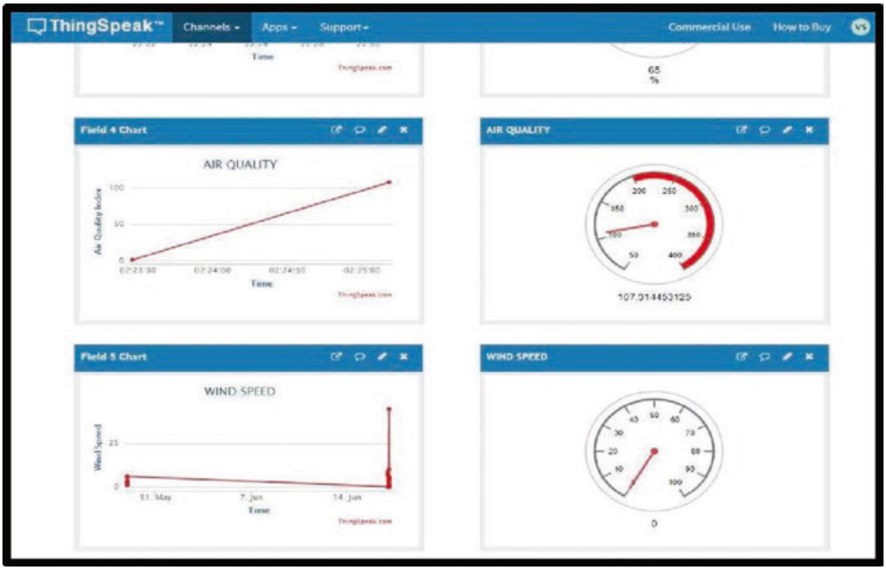


Fig. 10.11 Output at cloud platform

platform one can keep a watch on it and get live updates of environmental conditions from the comfort of their homes.

6 Conclusion

Climate monitoring is a crucial process since rapid differences in atmospheric condition causes an impact on the diverse community, budgetary, substantial aspects along with safeness, fitness, nutrition consumption, travel. Hence, the acquisition of appropriate weather data is necessary to produce the right end result. Detecting, monitoring, and forecasting weather parameters are very important for everyday life and to take preventive measures and precautions from upcoming natural disasters. Therefore, there is great usage and application of this hypothesis research. The display placed with the Automatic weather monitoring station in a locality helps the locals to get information about their surroundings by knowing the temperature, wind speed, sunlight intensity, air quality index, and humidity. All these parameters help them by making them aware about at which time in the day there is minimum sunlight intensity as well as minimum temperature and the air quality index which indicate how much the air outside is polluted and is it good to breathe or how much it affects the locals if they inhale the polluted air of that extent. The real-time data collected and provided to the meteorological department will help them to forecast weather of the particular area as the data provided to them was for a specific area while weather forecasting using satellite data is for a very large patch of the area which is also not accurate sometimes and for some places and mismatching of data with ground-level equipment may occur. This research has the potential to improve the existing methods and implement a smart, real-time, efficient, low-cost, accurate, low-power, portable, high-speed, IoT-based Automated Weather Monitoring Station.

References

1. Wang, N.-Y., & Chen, S.-M. (2009). Temperature prediction and TAIFEX forecasting based on automatic clustering techniques and two-factors high- order fuzzy time series. *Expert Systems with Applications, 36*(2), 2143–2154.
2. Kusmierek-Tomaszewska, R., Zarski, J., & Dudek, S. (2012). Meteorological automated weather station data application for plant water requirements estimation. *Computers and Electronics in Agriculture, 88*, 44–51.
3. Abbatea, S., Avvenutia, M., Carturan, L., & Cesarinia, D. (2013). Deploying a communicating Automatic Weather Station on an Alpine glacier. *The 3rd international Workshop on Sensor Networks for Intelligence Gathering and Monitoring (SNIGM) Procedia Computer Science, 19*, 1190–1195.
4. Zanella, S., Member, N., Bui, A., Castellani, L. V., & Zorzi, M. (2014). Internet of Things for smart cities. *IEEE Internet of Things, 1*(1), 22–32.
5. Brunelli, D., Minakov, I., Passerone, R., & Rossi, M. (2014). POVOMON: An Ad- hoc Wireless Sensor Network for indoor environmental monitoring. *2014 IEEE Workshop on Environmental Energy and Structural Monitoring Systems Proceedings, 1-6.*
6. Urban, G. (2015). An attemp to assess the results of air temperature measurements from automatic weather stations in comparison to glass thermometer measurements in the context of weather types. *Bulletin of Geography Series, 9*, 67–79.
7. Shah, J., & Mishra, B. (2016). IoT enabled environmental monitoring system for smart cities. *2016 International Conference on Internet of Things and Applications (IOTA)*, 383–388. https://doi.org/10.1109/IOTA.2016.7562757
8. Suksri, S., & Kimpan, W. (2016). Neural network training model for weather forecasting using Fireworks Algorithm. In *Proceedings of the 2016 international computer science and engineering conference (ICSEC).*
9. Sushmitha Kothapalli, & S.J. Totad,"A real time weather forecasting and analysis," pp. 1567-1570, 2017.
10. Munandar, A., Fakhrurroja, H., Rizqyawan, M. I., & Ratama, R. P. (2017). Design of weather monitoring system based on mobile application sing automatic weather station. *IEEE Transaction*, 44–46.
11. Ahvenniemi, H., Huovila, A., Pinto-Seppä, I., & Airaksinen, M. (**2017**). What are the differences between sustainable and smart cities. *Cities, 60*, 234–245.
12. Dolara, A., Gandelli, A., Grimaccia, F., Leva, S., & Mussetta, M. (2017). Weather-based Machine Learning technique for Day-Ahead Wind power forecasting. *IEEE*, 207–208.
13. Khan, M. S., Woo, M., Nam, K., & Chathoth, P. K. (**2017**). Smart city and smart tourism: A case of Dubai. *Sustainability, 9*, 2279.
14. Singh, R., et al. (2018). An imperative role of sun trackers in photovoltaic technology: A review. *Renewable and Sustainable Energy Reviews, 82*, 3263–3278.
15. Cowie, J., Linden, S., Mccandless, T., Kosovic, B., & Alessandrini, S. (2018). Machine learning for Applied Weather Prediction. *IEEE*, 276–277.
16. Rajab, H., & Cinkelr, T. (2018). IoT based Smart Cities. *2018 International Symposium on Networks, Computers and Communications (ISNCC)*, 1–4. https://doi.org/10.1109/ISNCC.2018.8530997

17. Nsabagwa, M., Mugume, I., Kasumba, R., & Muhumu, J. (2018). Condition monitoring and reporting framework for wireless network-based automatic weather stations", pp. 1, 8
18. Kiani, F., & Seyyedabbasi, A. (2018). Wireless sensor network and internet of things in precision agriculture. *International Journal of Advanced Computer Science and Applications, **9***(6), 99–103.
19. Srinivasan, K., Nema, A., Huang, C., & Ho, T. Y. (2018). Weather forecasting application using web-based-model-view-whatever framework. *IEEE*, 1–2.
20. Hasan, M., Anik, M. H., & Islam, S. (2018). Microcontroller based smart home system with enhanced appliance switching capacity. *2018 Fifth HCT Information Technology Trends (ITT)*, 364–367.
21. Ahmad, N., Uday, T. I. R., Islam, M. T., Patoary, R., Billah, M. M., Ahmed, N., & Tithi, F. S. (2019). *Portable mini-weather station for agricultural sector of rural area in Bangladesh*. Algorithms Intell Syst.
22. Sánchez-Corcuera, R., Nuñez-Marcos, A., Sesma-Solance, J., Bilbao-Jayo, A., Mulero, R., Zulaika, U., Azkune, G., & Almeida, A. (**2019**). Smart cities survey: Technologies, application domains and challenges for the cities of the future. *International Journal of Distributed Sensor Networks*, 15.
23. Singh, N., Chaturvedi, S., & Akhter, S. (2019). Weather forecasting using Machine Learning Algorithm. In *Proceedings of the 2019 international conference on signal processing and communication (ICSC)*.
24. Parashar, A. (2019) IoT based automated weather report generation and prediction using machine learning. In: 2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT), Jaipur, India, pp. 339–344
25. Javidroozi, V., Shah, H., & Feldman, G. (2019). Urban computing and smart cities: Towards changing city processes by applying enterprise systems integration practices. *IEEE Access, 7*, 108023–108034.
26. Chaganti, S. Y., Nanda, I., & Pandi, K. R. (2020). Cloud-based weather monitoring system. In *Advances in data science and management* (pp. 235–245). Springer.
27. Kumari, N., Gosavi, S., & Nagre, S. S. (2020). Real-time cloud based weather monitoring system. In *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*. IEEE.
28. Kavin, R., Lakshmi, K., Rani, S. S., & Rameshkumar, K. (2020). Weather monitoring system using internet of things. In *Proceedings of the 2020 6th international conference on advanced computing and communication systems (ICACCS)*.
29. Lakhara, V., Kurade, P., Pawar, T., Chougule, A., & Asurlekar, M. (2021). Real time weather monitoring system implementation based on Internet of Things. *Available at SSRN*, 3866819.
30. Ashwitha, A., & Latha, C. A. (2021). Smart weather acquisition, analysis and alert system. *Annals of the Romanian Society for Cell Biology, 25*(6), 11289–11298.
31. Sangjan, W., et al. (2021). Development of a raspberry pi-based sensor system for automated in-field monitoring to support crop breeding programs. *Inventions, 6*(2), 42.
32. Narasimha Rao, Y., et al. (2020). Providing enhanced security in IoT based smart weather system. *Indonesian Journal of Electrical Engineering and Computer Science, 18*(1), 9–15.
33. Khandelwal, D. D., & Singhal, M. (2021). Developing a low-cost weather monitoring system for data-sparse regions of the Himalayas. *Weather, 76*(2), 60–64.
34. Pi, Raspberry. (2015). Raspberry pi 3 model b. [Online]. https://www.raspberrypi.org

Chapter 11
Energy Harvesting for Sustainability

Parul Agarwal, M. Afshar Alam, Sheikh Mohammad Idrees, Ajay Vikram Singh, and Joel J. P. C. Rodrigues

Abstract The future era utilizes IoT in order to provide long-lasting energy and ensure its benefits and optimal usage in smart cities. But IoT device usage may not ensure sustainability and may rely on non-ambient sources of energy, most commonly the batteries, which can be quickly deployed but require periodic battery replacement. Energy harvesting provides solutions by utilizing ambient energy, which is ubiquitous and shall enable green communications. Energy harvesting is the collection of ambient energy in small amounts to power wireless devices. This proves promising where usage of batteries is impractical, say body sensor networks, in particular. It converts available energy to electrical energy which can be used later. It shall provide self-sustaining energy from the environment's ambient sources. This paper presents a significant focus on energy and energy harvesting methods that rely on solar, thermal, kinetic, and Radio Frequency. Also, Thermal electricity

P. Agarwal · M. A. Alam
Department of Computer Science and Engineering, Jamia Hamdard, New Delhi, India
e-mail: pagarwal@jamiahamdard.ac.in; aalam@jamiahamdard.ac.in

S. M. Idrees (✉)
Department of Computer Science (IDI), Norwegian University of Science and Technology, Trondheim, Norway

A. V. Singh
Amity Institute of IT, Amity University, Noida, UP, India
e-mail: avsingh1@amity.edu

J. J. P. C. Rodrigues
College of Computer Science and Technology, China University of Petroleum (East China), Qingdao, China

Instituto de Telecomunicações, Covilhã, Portugal
e-mail: joeljr@ieee.org

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*, Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_11

is one of the most crucial aspects in the field of bioelectricity. It has been found through the literature surveys that thermoelectric is found to be the most sustainable form of electricity. The implications and reasons for harvesting energy are discussed. This paper derives the conclusion that energy harvesting is one of the crucial aspects which needs to be widely adopted in order to save the environment and achieve sustainability.

Keywords Sensors · Ambient · Smart cities · Energy · Electricity · RF · Solar · Thermal · Kinetic

1 Introduction

Energy harvesting is found to be one of the emerging methods, which is also a viable technique for electronic devices for pulling sufficient energy from the surrounding environment. In one of the studies conducted, it has been stated that the surrounding environment for the energy materials comprises solar energy, thermal power, the energy of the wind, salinity gradients, as well as kinetic energy, which is also referred to as ambient energy [1]. Further, these forms of energy are converted into an electrical form for storing the power. This desirable technology is found to have significant potential in serving as one of the best alternative supplies of power for batteries, which are universally small, mobile as well as independent wireless devices for electricity, the ones that have their implications in wearable electronics as well as sensor networks (wireless) [2]. With respect to energy harvesting, it can be said that it is one of the most significant topics, which includes methods as well as materials that are used for converting mechanical energy, waste heat, as well as magnetic energy into electricity. In order to transform these forms of energy, photovoltaics, as well as thermoelectrics, are some of the major materials that are required along with some emerging technologies in converting them to electricity. This present study emphasizes the field of energy as well as harvesting materials. In this regard, this present study has significantly emphasized the various concepts of energy harvesting, which helps the industries in many ways.

With respect to the ever-growing demand for energy, the increasing prices for energy as well as various concerns regarding the environment, such as global warming, compel toward having a sustainable view and using smart technologies [3] in the context of having cleaner as well as more sustainable sources of energy. Also, IoT and its devices require a constant energy supply for proper functioning in smart environments [4]. This calls for concern in smart cities and ensuring an uninterruptible energy source. Harvesting thermal energy is one of the best methods for capturing the heat that is available abundantly in this environment. This trapped heat is further converted for making a usable form of energy, such as the mechanical form of energy or the electrical form of energy. In the context of this heat energy, it can be further stated that it is available in two major forms, which are (i) Waste Heat and (ii) Nature heat. It is further reported that more than half of the energy that is produced from several renewable [5] as well as nonrenewable sources across the world is mostly rejected within the environment, and most of the heat is rejected in the form of waste heat. Apart from this, it has also been found that natural resources, such as volcanic heat, geothermal heat as well as solar heat are some of the sources that provide enormous heat and are untapped. Further, in this context, thermal energy harvesting is one of the cost-effective techniques that help in recovering heat (wasted) as well as using heat (natural) in order to generate electricity that can be further revolutionized for the generation of the renewable form of energy. On the basis of the recorded temperature, it can be further said that heat is majorly categorized into three main categories. (i) High-grade, which ranges from *"650 °C or higher;"* (ii) medium grade, which ranges from "200 *°C to 277 °C;"* (iii) low-grade, which is lower than 2*77 °C* [6]. High as well as medium-grade heat can be recovered easily and on the other hand, low-grade heat constitutes more than 50% of the total waste heat and is one of the most difficult grades of waste heat to recover. When the thermal energy harvesters are taken into consideration, it is found that these have significant advantages over overheating engines (traditional). With respect to thermal harvesters of energy, it is found that these incur a very less operational cost. In addition, these are also found to be utilizing the thermal energy, which is freely available in the environment, such as wasted heat as well as natural heat. These forms of heat are further converted into a useful form of energy and are then utilized for carrying out useful work. When the thermal harvesters of heat are compared with the heat engines (traditional), it is found that these harvesters can be used at low temperatures. In addition, it can be further said that these harvesters are the most suitable techniques for several applications, such as sensor nodes, power stations, and for ensuring stability [7] and many more. It has also been found that these harvesters are devices in a solid form that do not produce any kind of harmful emission and these are also found to have the least cost in terms of maintenance. These harvesters also possess a low efficiency in terms of conversion of energy, which is usually between 1% and 10%. However, these are also observed as self-sustainable, as it helps them in making themselves appropriate in order to provide power in remote locations or to the locations that are difficult to be reached. It has been found that devices such as RTGs *"Radioisotope Thermoelectric Generators"* are used by NASA for more than 300 years and in this context, it has been observed that not even a single thermocouple has created any complication in the context of producing power or energy [8].

This paper has been divided into four sections. Section 2 discusses several harvesting techniques: concepts, and types that can prove to be beneficial in the present and future scenarios. Section 3 discusses the applications of these techniques with respect to smart cities. The conclusion is presented in Sect. 4.

2 Energy Harvesting

2.1 Concept of Energy Harvesting

The concept of energy harvesting is that lost energy during the creation and output process through vibration, light, heat, and movement will be harvested and increase efficiency rates by creating self-sustaining technologies. Piezoelectric harvesting, thermoelectric harvesting, and pyroelectric harvesting are some techniques to harvest the lost energy [10, 11].

2.2 Concept of Energy Harvesting Materials

When energy harvesting is taken into consideration, it is found that it holds significant importance in several fields. In this context, it has been observed that the harvesting of energy requires several types of methods as well as materials for converting the mechanical form of energy to the electrical form of energy. For transforming mechanical, heat, or magnetic energy into electrical, photovoltaics [12] along with thermoelectrics are required. It is further opined that innovative materials play a very vital role in the context of the development of *"energy harvesting materials,"* which include single crystals, ceramics, polymers as well as composites along with the developed technologies and materials [13], such as piezoelectric, thermoelectric, pyroelectric as well as magnetic materials.

2.3 Types of Energy Harvesting

With respect to the harvesting of energy, the following are some of the major common sources for ambient energy:

- Light energy
- Thermal energy
- Vibrational energy
- RF energy

2.3.1 Harvesting Solar Energy

Solar energy is the most commonly occurring form of energy. A detailed study related to it is discussed in [14]. With respect to solar energy, it is found that this form of energy can be harvested with the help of photovoltaic cells, which are also known as PV cells. In this regard, it can be stated that these PV cells are used for converting light energy, which is directly derived from the sun, into electrical form [15]. This principle is known as the photovoltaic effect. These effects are further defined as a process, where photons excite electrons into a higher form of energy and hence cause the generation of electric current. With respect to these PV cells, it is found that these PV cells are majorly categorized into four categories, which are:

- Thin-film cells
- Single as well as multi-junction sells
- Emerging technologies of PV
- Crystalline Si cells

With respect to these cells, it can be said that these cells are a bit expensive. Hence, instead of implementing these PV cells, LEDs (light-emitting diodes) can be implemented for harnessing solar energy as well as providing energy for the devices of low-power having an IoT edge. LEDs can work as power harvesters as they absorb photons. They are cost-efficient through lack the ability to absorb light in an optimized manner [16].

2.3.2 Harvesting Vibrational Energy

In order to harvest the kinetic energy, piezoelectric transducers are used. In this process, piezoelectric transducers are used for producing electricity with the help of kinetic energy. This kinetic energy is present in the form of sounds, movements, as well as vibrations [17].

In this process, it is found that the transducer transforms kinetic energy into *AC Current,* which is further processed into a specific form and then stored in a battery or a super-capacitor. Some of the implications of piezoelectric harvesting are as follows:

- Pressure sensors on the tires of cars: In this context, it can be said that the piezoelectric harvesting sensors are found to be installed within the car tires. These sensors help in monitoring the pressure of air present in the tire as well as demonstrate this particular information on the dashboard of the car.
- Remote control units (battery-less): With respect to the remote-control units, it has been found that with the help of piezoelectric transducers, the remote-control unit uses the force put on the buttons into energy which powers the IR signal of the remote.
- Floor tiles of piezoelectric: With reference to the tiles, it can be said that the pavements are lined with floor tiles made up of piezoelectric, which further convert the kinetic energy collected by the steps of pedestrians into the electrical form of energy. This converted energy is further used for powering several numbers of applications which are used for displaying as well as a ticketing system

2.3.3 Harvesting Thermal Energy

With reference to the electrical systems, it can be said that all the electrical systems produce heat. In this context, this emission of heat accounts for a significant proportion of degenerated energy. Therefore, in this context, it can be said that the thermoelectric harvesting of energy is usually based on a principle which is known as *the Seebeck effect* [18]. Further, with reference to the *Seebeck effect*, it can be said that it is the phenomenon where the difference in temperature at a particular junction between two *semiconductors* produces a particular voltage. In the context of the *thermoelectric system of harvesting,* it is found that it usually consists of a *thermoelectric generator* which further contains several *thermocouples*. *"Thermocouples"* are found to be connected in a particular series of a source of heat. This source of heat can be any source such as an engine, a water heater, solar panel, and many more. Further, it is found that the energy that is generated by this system is directly proportional to the difference in the temperature [19] as well as the size of the *thermoelectric generator*. In the context of this thermoelectric generator, it is found that this particular harvesting has its implication in powering sensor nodes (wireless) with respect to the industries as well as other environments that have a high temperature, where a significant amount of heat is being lost.

Energy harvesting is the process to reduce the energy generating sources, such as vibration, mechanical stress, temperature gradient, and light and convert to produce relatively low-power levels. Energy harvesting has received increased attention in the last few years due to the rapid increase in the population as well as an increase in energy demand. Energy harvester materials are the best alternative energy sources for low-power applications as the energy harvested is small. The best thing about energy harvesting materials is they support sustainability by promoting environmentally friendly technologies. The devices that use energy harvesting reduce the carbon dioxide emissions that make the devices indispensable to achieve sustainability. Energy harvesting technology is applied in the Internet of things (IoT) and wireless sensor networks (WSN) to eliminate the need for conventional batteries and network-based energy.

2.3.4 RF Energy Harvesting

With reference to radio frequency (RF) energy, it can be said that it is the energy that is present all around the environment. The Waves of RF [20] are found to be broadcasted into an environment with the help of mobile phones as well as some other electronic devices. In the context of these devices, it has been found that it has become a crucial part of the lives of humans. In the context of the technology used for harvesting RF energy, it is found to have a significant number of reasons. In this regard, it has been found that it has a mass appeal and it is found to be quite simple. Today, humans are living in a place where the transmitters are increasing at a significant rate. The waves of RF are found to be present everywhere and these waves represent free energy that is required to be tapped. Several devices such as Wi-Fi routers, microwave ovens, laptops, and many more are some of the other sources that produce RF energy waves that are required to be harnessed.

In the context of harvesting wireless energy, it has been found that it is one of the most trusted harvesting technologies due to its simplicity as well as ease of implementation. Waves of RF are collected with the help of an antenna that causes a potential difference in the length of this wave. This particular potential difference further creates a charge that is carried forward to move across the antenna's length. Further, an integrated circuit of RF is enabled to transform the mechanical energy and then store it in a capacitor. Apart from this, it has been found that the harvesting of RF energy has several types of applications [20, 21]. First and foremost, it has been found that it can be applied in close range of a transmitter. With respect to RF energy, it can be further said that it is used in devices that require low power. Other than this the RF energy has also found its application in the automation of building, industrial control as well as structural monitoring.

Table 11.1 depicts these harvesting techniques and discusses them with respect to a few parameters.

The above table identifies that though solar energy offers the best efficiency whether indoors or outdoors, it suffers from restricted daytime utilization. In terms of inherent characteristics, RF harvesting has several advantages to offer.

Smart cities are a reality. The devices are connected, the components of smart cities communicate, and are intelligent. The technological advances in several areas enable efficient and reliable services. These cities depend mainly on IoT which uses sensors for data collection. Though sensors can be grid-powered, the need for autonomous or wearable sensors will prove to be cutting-edge. The next section explores the major component of Smart cities: IoT and focuses on the use of energy harvesting techniques for powering it.

3 Energy Harvesting and Smart Cities

IoT also known as Internet of things and its variants include IoE (Internet of everything), and others is a technology that has enabled devices to be connected with people and all components of any network to be connected with each other for effective communication. Thus, empowering 2C's: connection and communication. It was introduced in 2009, with the purpose of establishing a connection of Radio Frequency with the internet [39].

IoT majorly comprises sensors, gateways, processors, and applications. Sensors collect the data, and gateways transfer processed data to an appropriate location. Bluetooth, Wi-Fi, etc., are responsible for the transportation and form the communication standards. The applications are responsible for the utilization of the collected data. The major application area of IoT includes Smart cities. In particular, it finds applications in smart transportation, smart healthcare, energy management, and smart buildings. For Wireless sensor nodes, the sources of power may be either renewable or nonrenewable. The nonrenewable sources of energy consist mainly of

Table 11.1 Study of various energy harvesting techniques

Type of energy harvesting	Derived from	Overall efficiency	Features	Applications	References
Solar energy	Photovoltaic cells, Photodiodes, LED'S	May vary between 3% and 35% depending on indoors or outdoors.	Functionally available only through daytime and depends on weather.	Satellites, calculators, lighthouses, space probes, power lighting, in smart agriculture for pumping water, solar panels on rooftops in smart cities, and smart villages for utilizing energy in cost-effective manner.	[22–23]
Thermal energy	Thermal electric generators (TEG)	0.8%–7% (depends on the industrial or human purpose).	With minute temperature changes, TEG'S exhibit low efficiency.	Homes for baking, cooking, cooling drying, and at industries for manufacturing.	[24, 25]
Vibrational energy	Piezoelectric transducers, movement of a coil.	10%–40% (depends on machine or human motion).	Energy harvested is restricted by bandwidth.	Operating helicopters, trains and for detecting fault in bearings at industries using smart sensors.	[26–32]
RF energy	RF sources like TV, Radio, FM, GSM, 3G, 4G, 5G, CDMA.	0.02%–25%(depending on its source).	Independent of environmental factors like temperature, humidity, weather, etc. Available day and night.	RFID tags for identification, recharging of several devices, smart sensors for homes and other application areas, smart lighting, WSN for building management.	[33–38]

batteries. But batteries are costly and have a negative environmental impact. Thus, arises the need for energy harvesting using renewable sources of energy available in abundance and those that offer negligible environmental pollution (if any). Several IoT-based autonomous devices can be powered using Photovoltaic cells, which are capable of supporting device-based Artificial Intelligence [40]. Solar energy can also be utilized effectively for powering IoT end node devices as discussed in [41, 42]. Also, harvested energy management has been studied in [43]. Vibrational energy using ReVibe [44] can be used for harvesting energy to be utilized for powering sensors. Medical monitoring systems rely on radio receivers, wristwatches, etc. We understand that generating energy from batteries is short-termed. Thus, we need to rely upon and look for alternatives in front of renewable sources for the same. This paper [45] discusses these techniques for biomedical research and throws light on the associated challenges.

Also, medical devices, which are implantable, require a battery for it to operate. Due to dead batteries, these devices are surgically replaced. The solution lies in energy harvesting from various sources of energy. The human body itself is a huge source and the main sources are the thermal and kinetic forms. These are derived from body heat, the motion of the arms, legs, and other body parts. In [46], the authors discuss these harvesters to power the medical devices. The parameters of analysis are size, power, and its applications. They conclude that a valid and logical solution for powering the batteries is energy harvesting techniques. As discussed, the major sources of energy of the human body are thermal and kinetic energy, so in [47], the authors scavenge energy from human movements for powering the wearable devices. They developed an energy harvester attached to the human knee that generates 1.6MW of power, and this is sufficient for powering small medical or electronic devices which may include embedded technologies, health monitoring devices or wearables, or GPS devices (for patient tracking). This leads the path for more self-powered wearable devices using the energy harvesting techniques discussed in this chapter. The future of smart healthcare would rely on remote patient monitoring, connected healthcare centers, connected ambulances, and all communicating through wearables or sensors. Another paper [48] throws light on the same aspect for medical sensors.

Not only does it have its applications for smart healthcare but also these energy harvesting methods can be exploited for smart homes, smart buildings, and smart transportation as well. Smart homes are in, and they collect data from sensors and control the environment of the home. They provide security and access control mechanisms, all related to the home [49]. The benefit they offer is saving time and lowering energy consumption [50]. IoT integrates the infrastructure using wireless technologies rather than physical connections. Integrating the system [51], depends on different features. The antenna system forms a crucial part of the smart devices, and factors to be considered for enhancing the IoT performance are the antenna size, gain [52, 53], and the radiation pattern. RF harvesting has gained importance for biomedical devices. The environment around us is full of radio waves ranging from 300 MHz to 3000 MHz. These waves have their applications in Mobiles, broadcasting, satellites, and Wi-Fi communications [54]. Abundantly available energy can be easily harvested, reused, and stored for IoT applications. In [55], a Zigbee-based wireless network for controlling the environment in a smart home is proposed. An embedded gateway bridges the Zigbee with the internet. In [56], the authors design an efficient, reliable, and flexible, real-time sensor network that monitors the wellness of a resident of the home. The nodes of the network track the usage and movements and depending on these parameters defines the wellness or the well-being of an individual. They extend the same approach for smart building designs and are associated with the performance of the system. Others for smart buildings can be studied [57, 58].

Smart transportation or the Intelligent Transportation system, commonly known as the ITS, comprises smart techniques that would improve the transportation for effectiveness in terms of travel time, costs associated, fuel-efficient utilization, and all these advantages are achieved in a sustainable manner. A major component of this is the roads as they are a means of transportation from one place to another. Roads made of plastic waste are already in and have been adopted in several countries. Besides, using them to harvest energy is new research and has been focused upon recently [59, 60]. Solar roads [61], as the name suggests, uses solar energy for harvesting and can be utilized for providing power to the street lights, signals, and signages. This harvested energy can also be stored and utilized using the power grid. This concept shall surely change the way energy would be utilized for the smart transportation sector. Solar roads use PV modules, which are placed on the surface of the road to capture light. This is helpful in lighting up the street dividers as well as for melting snow in cold places. Several countries that have adopted this technique are the USA, the UK, France, Italy, and others. Roads using piezoelectric devices for generating electrical energy are known as piezoelectric roads [62, 63]. Thus, we have seen that smart cities rely on Wireless networks and IoT for the operation of sensors, and wearables and have vast applications for smart healthcare, smart homes, smart buildings, and smart transportations. The future of smart cities shall witness a new era of optimum utilization of these techniques for an enhanced life and a sustainable future.

Though these techniques are useful, the sensors and wireless networks are prone to security threats that include replay, eavesdropping, denial of service, jamming, and others. More literature related to these can be explored [64–69].

4 Conclusion

We realize that IoT has exploded in a significant way and the coming years, sensors are estimated to range in billions. Industrial IoT is majorly aimed at carrying out the automated processes as well as making those processes more efficient. However, those devices as well as sensors are deployed and are required to be powered by batteries which would result in posting several other problems. These sensors are further found to be deployed in areas where it is crucial to reach and hence replacing those batteries is also a difficult task. However, it has also been found that replacing batteries also represents a significant cost which is limiting as well as also observed as a significant issue by the adopters. Therefore, in this context, it can be said that energy harvesting is required for eliminating such issues. When the energy is taken into consideration, it is found that it is available in abundant sources within the environment. This energy is further used for transforming into an electrical form of energy such that a variety of circuits can be powered. Energy harvesting is helpful in places where it is difficult to have a nearby source of power. Apart from that, it also eliminates the need for frequent replacement of batteries as well as the running wires at the end of the applications. Since smart cities and its applications in education, healthcare, transportation, and others rely on IoT heavily, energy harvesting using light, heat, vibrations, and RF shall have many future prospects to explore.

A few of these applications have already been explored in this chapter. These techniques are not only cost-effective, ubiquitous but are a sustainable phenomenon too which shall be helpful in reducing the carbon footprint of this planet.

Acknowledgment This work is partially funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the Project UIDB/50008/2020; and by Brazilian National Council for Scientific and Technological Development – CNPq, via Grant No. 313036/2020-9.

References

1. Madruga, S. (2021). Modeling of enhanced micro-energy harvesting of thermal ambient fluctuations with metallic foams embedded in Phase Change Materials. *Renewable Energy, 168*, 424–437.
2. Li, L., Xu, J., Liu, J., & Gao, F. (2018). Recent progress on piezoelectric energy harvesting: structures and materials. *Advanced Composites and Hybrid Materials, 1*(3), 478–505.
3. Nižetić, S., Ned, D., Papadopoulos, A., & Rodrigues, J. J. P. C. (September 2019). Smart technologies for promotion of energy efficiency, utilization of sustainable resources and waste management. In *Journal of Cleaner Production* (Vol. 231, pp. 565–591). Elsevier., ISSN: 0959-6526. https://doi.org/10.1016/j.jclepro.2019.04.397.
4. Muhammad Ammad, Munam Ali Shah, Saif ul Islam, Carsten Maple, Abdullah A. Alaulamie, Joel J. P. C. Rodrigues, Shafaq Mussadiq, Usman Tariq, A novel fog-based multi-level energy-efficient framework for IoT-enabled smart environments", *IEEE Access Journal*., ISSN (Print): 2169-3536, ISSN (Online): 2169–3536, 8, July 2020, pp. 150010–150026, https://doi.org/10.1109/ACCESS.2020.3010157.
5. https://www.nrdc.org/stories/renewable-energy-clean-facts. Accessed: 13.6.2021.
6. https://www.interreg-central.eu/Content.Node/CE-HEAT/Low-grade-waste-heat-utilization-in-the-European-Union.html accessed: 23.6.2021.
7. Saleem, B., Badar, R., Judge, M. A., Manzoor, A., Islam, S. U., & Rodrigues, J. J. P. C. (2021). Adaptive recurrent NeuroFuzzy control for power system stability in smart cities. In *Sustainable Energy Technologies and Assessments Journal* (Vol. 45, p. Paper Id: 101089). Elsevier., ISSN: 2213-1388. https://doi.org/10.1016/j.seta.2021.101089
8. Radioisotope Thermoelectric Generator (RTG). Available online: https://solarsystem.Nasa.Gov/rps/rtg.Cfm. (accessed on 23 June, 2021).
9. The Energy Flow Chart Released by Lawrence Livermore National Laboratory. Available online: https://flowcharts.llnl.gov/content/assets/images/charts/energy/energy_2011_world.png (accessed on 23 June, 2021)
10. Kiziroglou, M. E., & Yeatman, E. M. (2012). In J. A. Kilner, S. J. Skinner, S. J. C. Irvine, & P. P. Edwards (Eds.), *Materials and techniques for energy harvesting* (pp. 541–572). Woodhead Publishing Series in Energy, Functional Materials for Sustainable Energy Applications, Woodhead Publishing. ISBN 9780857090591.

11. Gould, C., & Edwards, R. (2016). *Review on micro-energy harvesting technologies* (pp. 1–5). 2016 51st International Universities Power Engineering Conference (UPEC). https://doi.org/10.1109/UPEC.2016.8114023
12. Koech, R. K., Kigozi, M., Bello, A., Onwualu, P. A., & Soboyejo, W. O. (2019, August). Recent advances in solar energy harvesting materials with particular emphasis on photovoltaic materials. In *2019 IEEE PES/IAS PowerAfrica* (pp. 627–632). IEEE.
13. Tong, C. (2019). Emerging materials for energy harvesting. *Introduction to Materials for Advanced Energy Systems*, 719–817.
14. Lee, S., Vandiver, M., Viswanathan, B., & Subramanian, V. (2012). Harvesting solar energy using inexpensive and benign materials. In W. Y. Chen, J. Seiner, T. Suzuki, & M. Lackner (Eds.), *Handbook of climate change mitigation*. Springer. https://doi.org/10.1007/978-1-4419-7991-9_32
15. https://www.energy.gov/eere/solar/solar-photovoltaic-cell-basics. Accessed: 24 June, 2021.
16. Moayeri Pour, G., & Leon-Salas, W. D. (2014). Solar energy harvesting with light emitting diodes. *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, 1981–1984. https://doi.org/10.1109/ISCAS.2014.6865551
17. https://www.uwsp.edu/cnr-ap/KEEP/nres633/Pages/Unit1/Section-B-Two-Main-Forms-of-Energy.aspx. Accessed 24 June, 2021
18. Verma, G., & Sharma, V. (**2019**). A novel thermoelectric energy harvester for wireless sensor network application. *IEEE Transactions on Industrial Electronics, 66*, 3530–3538.
19. Hou, L., Tan, S., Zhang, Z., & Bergmann, N. W. (**2018**). Thermal energy harvesting WSNs node for temperature monitoring in IIoT. *IEEE Access, 6*, 35243–35249.
20. Tran, L. G., Cha, H. K., & Park, W. T. (2017). RF power harvesting: A review on designing methodologies and applications. *Micro and Nano Systems Letters, **5***, 14. https://doi.org/10.1186/s40486-017-0051-0
21. Shinohara, N., & Kawasaki, S. (2009). Recent wireless power transmission technologies in Japan for space solar power station/satellite. In: *2009 IEEE radio and wireless symposium*, pp 13–15
22. Fei, F., Zhou, S., Mai, J. D., et al. (2014). Development of an indoor airflow energy harvesting system for building environ- ment monitoring. *Energies, 7*, 2985–2986.
23. Musiani, D., Lin, K., & Rosing, T. S. (April 2007). Active sensing plat- form for wireless structural health monitoring. In: *Proceedings of the 6th international symposium on information processing in sensor networks (IPSN'07)*, Cambridge, MA, 25–27, pp.390–398. New York: IEEE.
24. Vladimir, L. (2013). Thermoelectric energy harvesting of human body heat for wearable sensor. *IEEE Sensors Journal, 13*, 2284–2287.
25. Vullers, R. J. M., Schaijk, R., Visser, H. J., et al. (2010). Energy harvesting for autonomous wireless sensor networks. *IEEE Solid-State Circuits Magazine, 2*, 29–38.
26. Discenzo, F. M., Chung, D., & Loparo, K. A. (2006). Pump condition monitoring using self-powered wireless sensors. *Journal of Sound and Vibration, 40*, 12–15.
27. Yang, Z., & Zu, J. (2016). Toward harvesting vibration energy from multiple directions by a nonlinear compressive- mode piezoelectric transducer. *IEEE/ASME Transactions on Mechatronics, 21*, 1787–1789.
28. Wang, L., & Yuan, F. G. (2008). Vibration energy harvesting by magnetostrictive material. *Smart Materials and Structures, 17*, 045009.
29. Zucca, M., & Bottauscio, O. (2012). Hysteretic modeling of electrical micro-power generators based on Villari effect. *IEEE T Magnet, 48*, 3092–3095.
30. Ylli, K., Hoffmann, D., Willmann, A., et al. (2015). Energy harvest- ing from human motion: exploiting swing and shock excitations. *Smart Materials and Structures, 24*, 1–3.
31. Hadas, Z., Vetiska, V., Vetiska, J., et al. (2016). Analysis and efficiency measurement of electromagnetic vibration energy harvesting system. *Microsystem Technologies, 22*, 1768–1772.
32. Tan, Y. K. (2013). *Energy harvesting autonomous systems: design, analysis, and practical implementation* (pp. 37–62). CRC Press Taylor & Francis Group.

33. Liu, V., Parks, A., Talla, V., et al. (2013). Ambient backscatter: Wireless communication out of thin air. *ACM SIGCOMM Computer Communication Review, 43*, 39–50.
34. Vyas, R. J., Cook, B., Kawahara, Y., et al. (2013). E-WEHP: A batteryless embedded sensor platform wirelessly powered from ambient digital-TV signals. *IEEE T Microw Theory, 61*, 2491–2505.
35. Pravizso, A. & Agrawal, J. P. (May 2014). RF energy harvesting. In: *Proceedings of the 64th electronic components and technol- ogy conference*, Orlando, FL, 27–30. New York: IEEE.
36. Bouchouicha D, Latrach M, Dupont F, et al. An experimental evaluation of surrounding RF energy harvesting devices. In: *Proceedings of the 2010 European microwave conference (EuMC)*, Paris, 28–30 September 2010, pp.1381–1384. New York: IEEE.
37. Kenneth G, Chemishkian S, Hull JJ, et al. Feasibility of wireless sensors using ambient 2.4 GHz RF energy. In: *Proceedings of the 2012 IEEE sensors*, Taipei, Taiwan, 28–31 October 2012, pp.1–4. New York: IEEE.
38. SAS, I. (2012). *Ansys mechanical apdl theory reference*. ANSYS, Inc..
39. Ashton, K. (**2009**). That 'Internet of Things' thing. *RFID Journal, 22*, 97–114.
40. Michaels, H., Rinderle, M., Freitag, R., Benesperi, I., Edvinsson, T., Socher, R., Gagliardi, A., & Freitag, M. (**2020**). Dye-sensitized solar cells under ambient light powering machine learning: Towards autonomous smart sensors for the internet of things. *Chemical Science, 11*, 2895–2906.
41. Ram, S.K., Das, B.B., Swain, A.K., Mahapatra, K.K. Ultra-Low Power Solar Energy Harvester for IoT Edge Node Devices. In Proceedings of the 2019 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS), Rourkela, India, 16–18 December 2019; pp. 205–208.
42. Ram, S.K., Sahoo, S.R., Sudeendra, K., Mahapatra, K. Energy efficient ultra low power solar harvesting system design with mppt for iot edge node devices. In Proceedings of the 2018 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS), Hyderabad, India, 17–19 December 2018; pp. 130–133.
43. Bedier, M., Karami, A., Galayko, D., Basset, P. Autonomous energy management interface for electrostatic series-parallel charge pump vibrational energy harvester. In Proceedings of the 2017 15th IEEE International New Circuits and Systems Conference (NEWCAS), Strasbourg, France, 25–28 June 2017; pp. 385–388.
44. ReVibe Energy. Available online: http://revibeenergy.com/productsservices/ (accessed on 24 June, 2021).
45. Romero, E., Warrington, R. O., & Neuman, M. R. (2009). Energy scavenging sources for biomedical sensors. *Physiological Measurement, 30*(9).
46. Andrea, C., Alessandro, D., Emilio, S., & Mauro, S. (2014). Kinetic and thermal energy harvesters for implantable medical devices and biomedical autonomous sensors. *Measurement Science and Technology, 25*.
47. American Institute of Physics. "Harvesting energy from the human knee: Researchers develop a way to power small devices by walking." ScienceDaily. www.sciencedaily.com/releases/2019/07/190717122600.htm (accessed June 30, 2021).
48. Gljušćić, P., Zelenika, S., Blažević, D., & Kamenar, E. (**2019**). Kinetic energy harvesting for wearable medical sensors. *Sensors, 19*, 4922. https://doi.org/10.3390/s19224922
49. Okba, A., Takacs, A., & Aubert, H. (2018). Compact flat dipole rectenna for IoT applications. *Progress In Electromagnetics Research C, 87*, 39–49.
50. Shafique, K., Khawaja, B. A., Khurram, M. D., et al. (2018). Energy harvesting using a low-cost rectenna for internet of things (IoT) applications. *IEEE Access, 6*, 30932–30941.
51. Zeng, M., Li, Z., Andrenko, A. S., Zeng, Y., & Tan, H. Z. (2018). A compact dual-band rectenna for GSM900 and GSM1800 energy harvesting. *International Journal of Antennas and Propagation, 2018*., Article ID 4781465, 9 pages.
52. Lu, L., Jiao, Y. C., Weng, Z. B., Zhang, L., Cui, C. Y., & Wang, R. Q. (2018). High-efficiency circularly polarized dielectric resonator antenna array fed by the cavity-backed SIW. *IEEE Antennas and Wireless Propagation Letters, 17*(7), 1145–1148.

53. Mao, X., Li, K., Zhang, Z., & Liang, J. (September 2017). Design and implementation of a new smart home control system based on Internet of Things. In *2017 International Smart Cities Conference (ISC2)* (pp. 1–5).
54. Nguyen, N. H., Bui, T. D., Le, A. D., et al. (2018., Article ID 1692018, 9 pages). A novel wideband circularly polarized antenna for RF energy harvesting in wireless sensor nodes. *International Journal of Antennas and Propagation, 2018*.
55. Gao, Y., Qin, Z., Zhang, R., Zhang, W., Duan, Y., & Li, Z. (2018). Research on data collection design based on Zigbee wireless technology smart home system. *IOP Conference Series: Materials Science and Engineering, 452*, article 042057.
56. Ghayvat, H., Mukhopadhyay, S., Gui, X., & Suryadevara, N. (2015). WSN- and IoT-based smart homes and their extension to smart buildings. *Sensors, 15*(5), 10350–10379. https://doi.org/10.3390/s150510350
57. Shah, J., & Mishra, B. (2016). Customized IoT enabled wireless sensing and monitoring platform for smart buildings. *Procedia Technology, 23*, 256–263.
58. Alsuhli, G., & Khattab, A. (February 2019). A fog-based IoT platform for smart buildings. In *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)* (pp. 174–179).
59. *On Road Energy Harvesting – EU POWERAMP*. https://cordis.europa.eu/result/rcn/92180_ en.html
60. Diamond, K. (2009). *Climate Change, Sustainable Development, and Ecosystems Committee Newsletter*. American Bar Association.
61. Williams, L. (2018) Is it the end of the road for asphalt and concrete?'. *IET Engineering & Technology Magazine*. https://eandt.theiet.org/content/articles/2018/09/is-it-the-end-of- the-road-for-asphalt-and-concrete/
62. Hill, D., Agarwa, A., & Tong, N. (2014). *Assessment of piezoelectric materials for roadway energy harvesting*. Final Report, Energy R&D Division, California Energy Commission.
63. Kokkinopoulos, A., Vokas, G., & Papageorgas, P. (2014). Energy harvesting implementing embedded piezoelectric generators – The potential for the Attiki Odos traffic grid. *Energy Procedia, **50***, 1070–1085. https://doi.org/10.1016/j.egypro.2014.06.126
64. Liu, Q., Yildirim, K. S., Pawełczak, P., & Warnier, M. (**2016**). Safe and secure wireless power transfer networks: Challenges and opportunities in RF-based systems. *IEEE Communications Magazine, 54*, 74–79.
65. Nguyen, T. G., So-In, C., & Ha, D. B. (**2017**). Secrecy performance analysis of energy harvesting wireless sensor networks with a friendly jammer. *IEEE Access, 5*, 25196–25206.
66. Shakhov, V., Nam, S., Choo, H. Flooding attack in energy harvesting wireless sensor networks. In Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication, Kota Kinabalu, Malaysia, 17–19 January 2013; pp. 1–5.
67. Knorn, S., & Teixeira, A. (**2019**). Effects of Jamming Attacks on a Control System with Energy Harvesting. *IEEE Control Systems Letters, 3*, 829–834.
68. Luo, Z., Wang, W., Xiao, J., Huang, Q., Jiang, T., & Zhang, Q. (**2018**). Authenticating on-body backscatter by exploiting propagation signatures. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technologies, 2*, 1–22.
69. Di Mauro, A., Papini, D., & Dragoni, N. (2012). *Security challenges for energy-harvesting wireless sensor networks* (pp. 422–425). PECCS.

Chapter 12
A Review of Machine Learning Models in Renewable Energy

Anuj Gupta, Kapil Gupta, and Sumit Saroha

Abstract Renewable energy is gradually being used to offset the impact of climate change and global warming. Various forecasting techniques have been introduced to enhance renewable energy prediction ability. The objective of this research is structured as follows: Firstly, this study examines machine learning methods for forecasting renewable energy resources. Secondly, this survey demonstrates the process implemented in the machine learning model for forecasting the performance of the machine learning model. Thirdly, the analyses of renewable energy forecasting models were conducted on the basis of mean absolute percentage error and correlation coefficient. Finally, at the conclusion of this study, several possible future work opportunities were identified.

Keyword Machine learning models · Forecasting · Evaluation metrics · Renewable energy

1 Introduction

1.1 Renewable Energy

With the rapid rise of global industrialization, it has become evident that continued use of fossil fuel will not only expedite the depletion of fossil fuel reserves but will also degrade the atmosphere, affecting human health and raising the risk of global warming [1]. Among the different resources of energy, renewable energy is

A. Gupta (✉) · K. Gupta
Maharishi Markandeshwar (Deemed to be University), Mullana-Ambala, India

S. Saroha
Guru Jambheshwar University of Science and Technology, Hisar, India

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_12

naturally available on the earth surface such as wind energy, solar energy, tides etc. Renewable energy has drawn interest due to its characteristics of sustainability and zero emission of CO_2 but the main challenge of the renewable energy sector is to provide uninterrupted energy supply in the near future. Sustainable energy systems would be able to solve some of today's most challenging energy issues such as increasing energy supply, providing reliability, and resolving regional energy crises [2]. However, the generation of electric energy is sporadic and unpredictable due to the high volatility and dynamic properties of renewable energy. Therefore, Researchers are still to grappling with question on how to deal with the erratic characterstics of renewable energy. Accurate prediction of energy improves the efficiency of energy systems. So several machine learning models were developed for accurate forecasting of renewable energy. In addition, various ensemble models were developed to enhance the performance of machine learning models. Various time periods were used, such as minutes, hours, days, and weeks depending on the intent of prediction. Various evaluation metrics were used to check the effectiveness of the forecasting machine used in renewable energy predictions [3].

1.2 Machine Learning

Machine Learning models have been widely used in numerous fields during the last few decades, including data processing, artificial intelligence, optimization, and statistics. Machine learning models attempt to find a relation between dependent and independent techniques which were used by researchers to improve the prediction accuracy of forecasted models [4]. Machine learning models consist of three methods: supervised, unsupervised, and reinforcement methods. The supervised method involves learning of the model using a data set and reference pattern is also provided for accomplishing a job, while the unsupervised learning is a self-learning technique. The reinforcement method learns from feedback and past experience to enhance the accuracy of models. Based on these basic learning methods various mechanisms and theories were discussed in the literature [5]. Due to the rapid growth of the machine learning field, deep learning emerges as a subfield of machine learning. Deep learning is a set of analytical machine learning techniques for learning feature hierarchies which are often focused on artificial neural network (ANN) [6]. Deep learning is a modern substitute for machine learning; we have a variety of structured and unstructured data in various forms and aspects from every region of the world. Structural data can be easily drowned out, while unstructured data could take decades to provide relevant information. Deep learning, often known as big data, is a strategy for dealing with massive amounts of data taken from various mediums, such as social media, online platforms, e-commerce, internet engine search, and so on. This abundance of data is smoothly accessible. It can be shared through fintech applications such as mobile payment applications etc. [7].

1.3 *Study of Machine Learning Techniques to Renewable Energy*

This section discusses the literature on machine learning techniques to forecast renewable energy. Table 12.1 presents a list of some recent studies related to machine learning techniques applying to renewable energy. Voyant et al. presented a review on solar radiation forecasting using machine learning techniques. According to the author, standalone models such as artificial neural networks, linear regression, and support vector machines performed well in the forecasting field, while the hybrid model is a viable way to increase the performance of prediction models [8]. Das et al. reviewed how to forecast solar power and analysis of evaluation metrics in machine learning models. The author reported that support vector machine and neural network-based models are very famous in the forecasting field [9]. Zendehboudi et al. presented a review on the hybrid support vector machine (SVM) model and outperformed the standalone SVM model performance. The main focus of the author is to forecast solar and wind energy using machine learning models [3]. Ahmed and Khalid et al. conducted a review on energy policy, market strategies, and storage and dispatch of energy and its reliability. This type of review is very helpful for understanding the government schemes about renewable energy and forecasting system design and its operation [10]. Mosavi et al. focused on the implementation and categorization of models in energy systems. In energy system application, hybrid models outperform classical machine learning models [11]. Bermejo et al. presented a review on solar, wind, and hydraulic energy and also described forecasting of energy using machine learning models and how we increase its reliability [12]. Wang et al. presented a forecasted model using deep learning techniques. The author applies preprocessing techniques to increase the accuracy of the forecasted model [1]. Melit et al. conducted a review on machine learning techniques. According to the author, deep learning techniques and numerical weather forecasts with extracting features used to generate long-term photovoltaic power generation; and for determining the time dependence information in forecasting the performance of long short-term memory networks, convolution neural networks, and recurrent neural networks were better [13].

Table 12.1 Literature study of applying machine learning techniques to renewable energy

Ref. no.	Name of author	Sources
[8]	Voyant et al. [2017]	Solar
[9]	Das et al. [2018]	Solar
[3]	Zendehboudi et al. [2019]	Solar, wind
[10]	Ahmed and Khalid et al. [2019]	Solar, wind
[11]	Mosavi et al. [2019]	Solar, wind, hydropower
[12]	Bermejo et al. [2019]	Solar, wind, hydropower
[1]	Wang et al. [2019]	Solar wind
[13]	Mellit et al. [2020]	Solar

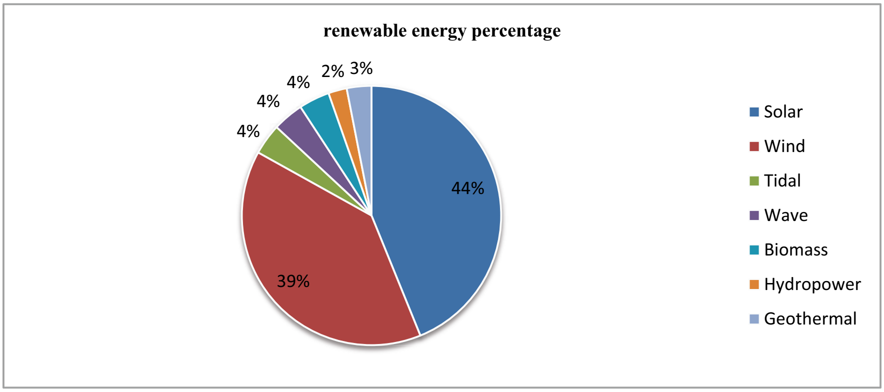


Fig. 12.1 Pie chart depicting percentage of seven different renewable energy sources

2 Renewable Energy Forecasting Using Machine Learning Models

Renewable Energy is highly beneficial to the environment due to zero emission of CO_2. However, due to climatic variability, the accurate forecasting is a very complex process for researchers, but machine learning models provide an efficient way to address this problem. Machine learning models consist of four steps. In the first step, original data is collected. In the second step, preprocessing techniques are applied to refine the input data as per requirement. The third step involves the selection of a model for forecasting and in the last step verification of the trained model is carried out. Figure 12.1 represents a pie chart consisting of seven types of renewable energy resources and one important thing to be noticed is that both wind and solar energy percentages are nearly the same while the other renewable energy resources such as wave, tidal, biomass, hydropower, and geothermal are less than five percent in this study.

2.1 Current Status of Machine Learning Models in Renewable Energy System

Table 12.2 highlights the papers on the subject of using machine learning models in alternative energy resources from 2017. In the renewable energy system, techniques come under three categories: mathematical model, artificial intelligence approach, and ensemble model [14]. In wind energy forecasting mathematical models came in the early phase [15, 16]. Recently, researchers used an artificial intelligence approach to forecast wind energy such as artificial neural network [17–31], support vector machine [14–16], long short-term memory neural network [32–38], adaptive neuro fuzzy inference system [39], and random forest classification. In addition,

Table 12.2 Energy sources, strategies, and techniques in the forecast of sustainable Energy

Energy Source	Methods	Techniques
Solar energy	Mathematical approach	ARIMA [52, 53]
	Artificial intelligence approach	GRU [54];RF [55, 57];SVR RF [56];GBR [58]; LR,DT,SVM,ANN [59];ANN,GBR,SVM,RF [60];ANN [61, 65, 107];CNN [66];DNN [104];DNN,RNN,LSTM [67];ML&SHM,GRU,LSTM [68];LSTM [69, 71, 72];LSTM,GRU [70];CBNQR [75]; GPR [91];RDANN [94];ELM [105];static and dynamic ensemble [106]
	Ensemble approach	WD-Ensemble [73]; IMFOA-SVM [74];SVM-PSO [76];CBA,ANN,SVM [77];EEMD-LSSVR [95];Least absolute shrinkage and selection operator, LSTM [96]; PSO-ELM [99];Artificial bee colony-empirical models [101]
Wind energy	Mathematical	PIS [15]
	Artificial intelligence approach	GPR,SVR,ANN [14]; Xgbr, SVR,RF [16];SVR,ANN,GB [40];RF [41]; MLP [17];DNN,PCA [18];FFANN [19];EDCNN [20, 27];LRNN.SVR [21];CNN [22, 30];DNN [23, 24];Stacked auto encoder-BP [26];PDCNN [28];IRBFNN [29];GP [31];LSTM [32];ILSTM [33];LSTM-ANN [34];Auto LSTM [36]; SWLSTM [37];ELSTM [38]
	Ensemble approach	ANFIS,PSO,GA [39];Improved dragonfly algorithm-SVM [42];DBN with GA [43]; Type-2 FNN-PSO [44]; MO and LA-LSSVM [45]; CEEMDGWO [98];VMD-BSELM [109];SELM [110];VMD-SSA-LSTM-ELM [111];EEMD-DBM [46];ELM-ICEEMD [47];BMA and EL [48];SBRFR [49];KPCA-CVR-COR [50];WPD-LSTM [51];EWT,RNN [112]
Wave	Artificial intelligence approach	FIS,ANN [115];GPR [116]
	Ensemble approach	ICEEMD-ELM [84];BOGGA-ELM [85]
Hydropower	Artificial intelligence approach	BLR [108]
	Ensemble approach	GWO-ANFIS [78];BSDP [113]
Biomass	Artificial intelligence approach	LR,DTR,SVM [80];GBRT [81];DTR,MP [82]
	Ensemble approach	SVM-SA [83];PSO-SVM [114]
Tidal	Artificial intelligence approach	Wavelet-SVR [86]

(continued)

Table 12.2 (continued)

Energy Source	Methods	Techniques
	Ensemble approach	EEMD-LSSVM [89, 93];MHS [102]
Geothermal	Artificial intelligence approach	LSTM [89]
	Ensemble approach	TDNN [90]

ensemble methods were implemented to increase the accuracy of the forecasted model in which they used preprocessing techniques and optimization algorithms with machine learning models [42–45]. A number of preprocessing techniques exist in the literature such as empirical mode decomposition, ensemble empirical mode decomposition, wavelet transform, and wavelet packet decomposition which was used to overcome the effect of noise from input data and increase the performance of wind speed forecasting [46, 47]. One important thing is to be noticed from the literature that the Bayesian method showed more accurate results than the hybrid wind speed prediction model [48, 49] [50]. Implemented a methodology for predicting wind speed using a dynamic integration approach. In this, the nonlinear characteristics were extracted from the original data using the principal component analysis method. The results of the proposed models outperform other forecasted models. An ensemble method consists of a deep learning model: long short-term memory neural network and convolutional neural network with wavelet packet decomposition preprocessing technique which was implemented to forecast wind speed and is capable of delivering satisfactory prediction outcomes [51]. In solar forecasting method prediction can be done on different-2 time scales. The most popular statistical model was the autoregressive integrated moving average [52, 53], artificial neural network [61–65] and support vector machine [54–60] come under the categories of machine learning models while deep neural network consists of long short-term memory [64, 67–72] and convolution neural network [66]. The hybrid model is made up of a combination of more than one model with preprocessing technique for multistep-ahead prediction [71–77]. An ensemble model of adaptive neuro fuzzy inference system and a gray wolf optimization technique was used to forecast hydropower [78]. Biomass and hydropower are two clean energy sources that can help to fulfill the world energy demand with zero emission of carbon dioxide [79]. In the literature, various machine learning models such as support vector machine regression, linear regression model, k-nearest neighbor regression models reported that they are capable of modeling biogas products without additional modification [80, 81]. Various researchers used four models: gradient boosting, support vector regression, selection tree regression, and neural network to forecast Co, Co_2, and H_2 during biogas production and high heating voltage output [82]. Among all four models, gradient boosting and selection tree regression generate better forecasting performance. A new ensemble model used a combination of neural network and simulated annealing optimization techniques to investigate the baking procedure for implementing a forecasted model of the high heating value (HHV) in biomass [83]. The outcome of this combination outperforms other forecast models. For the prediction of wave generation, an ensemble model named complete ensemble empirical mode decomposition-Empirical learning model (CEEMD-ELM) model was developed to forecast wave heights [85]. For the forecasting of tidal power generation, an artificial intelligence approach based on a support vector regression model with wavelet transform was implemented to predict the tide velocity [86]. In literature, various hybrid models were proposed to forecast the tidal speed and direction with improved accuracy. An ensemble method used a combination of wavelet neural network model and Fourier series method. The results of the developed model improved prediction accuracy [87]. The forecasting approach was implemented based on the least square support vector machine and ensemble mode decomposition to enhance the quality of tidal speed and direction [88]. In Geothermal energy forecasting, [89] researchers used the Long short term memory neural network (LSTM) encoding and decoding techniques. These techniques learn from the past geothermal energy output and make forecasts about future geothermal energy production. The performance of the encoder and the time duration of the LSTM with standardized periods and sequential lengths were used to make geothermal energy generation predictions [90]. Explore geothermal steam generators to use previous data to get the correct estimation of the sensors installed along with the steam generators in systems. The result revealed that in every case the time-dependent neural network model outperformed other traditional regression indicators. Table 12.2 shows that solar and wind seem to be the two most commonly examined renewable energies and artificial intelligence approach and ensemble models are the two most commonly used technologies.

ARIMA: Auto Regressive Integrated Moving Average; GRU: Gated Recurrent Unit; RF: Random Forests; SVR: Support Vector Regression; GBR: Gate Boosted Regression; LR: Linear Regression; DT: Decision Trees; SVM: Support Vector Machine; ANN: Artificial Neural Network; CNN: Convolution Neural Network; DNN: Deep Neural Network; RNN: Recurrent Neural Network; LSTM: Long Short-Term Memory neural network; CBNQR: Copula Base Nonlinear Quantile Regression; GPR: Gaussian Process Regression; RDANN: Regime dependent artificial neural network; ELM: Extreme Learning Machine; WD-Ensemble: Wavelet Decomposition Ensemble; IMFOA-SVM: Improve moth-Flame Optimization Algorithm-Support Vector Machine; SVM-PSO: Support Vector Machine- Particle Swarm Optimization; CBA: Cluster Based Approach; EEMD-LSSVR: Ensemble Empirical Mode Decomposition-Least Square Support Vector Regression; PIS: Physics Informed Statistical; xGBR: xGBoost regression; MLP: Multilayer Perceptron; PCA: Principal Component Analysis; FFANN: Feed Forward artificial neural network; EDCNN: Efficient deep CNN; PDCNN: Predictive Deep CNN; IRBFNN: Improved radial basis function neural network; ILSTM: Improved Long Short-Term Memory neural network; SWLSTM: Shared weight LSTM; ELSTM: Ensemble LSTM; ANFIS: Adaptive Neuro Fuzzy Inference system; DBN: Deep Belief Network; MO and LA: Multi objective and Lion algorithm; CEEMDGWO:

Complete EEMD grey wolf optimization; VMD-BSELM: Variation Mode Decomposition- Backtracking Search regularized ELM; SELM: Stacked extreme learning machine; SSA: Singular Spectrum analysis; DBM: Deep Boltzman Machine; ICEEMD: Improved EEMD; BMA: Bayesian Model Averaging; EL: Ensemble Learning; SBRFR: Sparse Bayesian based Robust Functional Regression; KPCA-CVR-COR: Kernel Principal Component Analysis-Core Vector Regression-Competition Over Resource; WPD: Wavelet Packet Decomposition; FIS: Fuzzy Inference System; BOGGA: Bayesian Optimization Grouping Genetic Algorithm; BLR: Bayesian Linear Regression; BSDP: Bayesian Stochastic Dynamic Programming; MHS: Modified Harmony Search; TDNN: Time- Dependent Neural Network.

2.2 Data Preprocessing Techniques

There are four types of modeling methods used in energy forecasting: data preprocessing, deciding the learning parameter of the model, learning and examining the model, and prediction issue [45]. The data collected from multiple sources is primarily in raw format and lacks important qualities that would allow it to be accurate. As a result, the data must be preprocessed, which includes scaling up or down the input measurement, cleaning, and defining the input data according to the standards [48, 73]. The machine learning model employed various preprocessing techniques to measure the performance of the forecasting model as listed in Table 12.3. In the case of data splitting, a reliable machine learning model is created by dividing input data into three parts: learning sample, verification data set, and testing sample [48, 82, 91]. The kernelized Mahalanobis distance (kWMD) method can be used to determine the similarity between two different samples [61]. Data decomposition divides the original data into many sub-datasets which are used to increase the performance of the forecasted model [92]. Data discretization is a technique for

Table 12.3 Decomposition techniques employed by the machine learning models for renewable energy prediction

Ref. no.	Decomposition technique	Total paper used
[48, 82, 91]	Data splitting	3
[61]	Kernelized Mahalanobis distance method	1
[18, 20, 25, 45, 47, 48, 50, 51, 61, 69, 73, 84, 86, 95, 98, 105, 109, 111, 112]	Data decomposition	19
[57, 93–96]	Data discretization	5
[36]	Feature selection	1
[14, 25, 31, 32, 42, 48, 56]	Data normalization	7
[17]	Data standardization	1
[97]	Data transformation	1

transforming statistical information into finite data, which is particularly useful when predicting renewable energy using weather data [93–96]. The encoding categorical features method cleans up the original data by removing impurities and duplication and creating more effective features to represent the relationship between issues and modelling techniques [57]. Feature selection is a strategy for finding suitable significant factors and deleting extra features [36]. To boost processing efficiency, high-frequency signals and crosstalk were compressed into a smaller range using Data Normalization. For example, when forecasting wind speed, weather parameters must be normalized [14, 25, 31, 32, 42, 48, 56]. Data standardization is a technique for converting data of various sizes into a single size [17]. Data transformation is the process of changing the position of data. A popular data transformation approach is the Markov chain process [97]. Tables 12.4 and 12.5 represent the decomposition technique used by the machine learning model for renewable energy forecasting based on different renewable energy resources and different variables.

2.3 Selection of Machine Learning Model Parameters

The accuracy of the projected model is influenced by the input parameters chosen. In literature, various researchers work on the selection of input parameters for the forecasted model. To predict hydropower generation, Majid Dehghani et al. used grey wolf optimization (GWO) approach to select relationship parameters of the adaptive network fuzzy inference system (ANFIS) model. ANFIS forecasting efficiency can be greatly enhanced by GWO [78]. Wu et al. used a combination of ELM model and grey wolf optimization to increase wind speed prediction [98]. In forecasting solar radiation, Liu et al. employed the firefly algorithm (FFA) to assess specifications of SVM models. The developed SVM-FFA model results outperform other forecasted approaches [75]. Various researchers also used particle swarm optimization (PSO) to select the parameters in solar and wind forecasting [39, 44, 76, 99, 100]. Fan et al. used a combination of the SVM + PSO + BAT algorithm to forecast solar radiation. Particle swarm optimization and BAT algorithm were used for specifying the parameters for the SVM model [76]. To forecast global solar radiation, Demircan et al. developed an empirical regression model using artificial

Table 12.4 Evaluation of machine learning model performance by more than 11 studies

ES EM	Solar	Wind	Tidal	Wave	Geothermal	Biomass	Hydropower	Total
MSE	2	7	0	1	1	0	0	11
NRMSE	8	3	0	0	0	0	0	11
CC	8	7	1	2	1	1	0	20
RMSE	35	30	1	3	0	2	2	71
MAPE	11	22	0	1	0	0	2	36
MAE	15	24	1	2	1	2	0	45

Table 12.5 MAPE value by various studies for renewable energy resources

Energy source	MAPE (%)/Ref.NO.
Solar energy	17.7[52],2.4[[104],7.4[70],2.8[74],1.3[77] 0.2[91],4.7[96],7.4[105],2.7[106],5.8[107],1.4[108]
Wind energy	8.1[41],1.6[19],2.4[20],3.3[26],6.5[28],3.8[29],17.1[38], 8.6[14],0.8[14],2.3[98],2.2[109],5.1[110],2.2[111],1.7[46], 7.6[47],6.0[48],6.2[49],15.1[50],14.4[51],3.0[112],3.7[113],7.5[114]
Biomass	0.99[80],0.93[81],0.90[83]
Tidal	0.9[86],1.7[88],2.0[102]

bee colony (ABC) algorithms. To make a prediction, the forecasting model used periods and angles of sunshine. To pick parameters of empirical regression models, artificial bee colony algorithms were used [101]. To predict the HHV of biomass, Garca Nieto et al. introduced a new model of support vector machine and simulated annealing (SA). The SA was used to choose SVM parameters, and the suggested technique shows good results in terms of forecasting [83]. Lin et al. employed a hybrid model named as deep belied network genetic algorithm to forecast wind speed. Genetic algorithm was used for specifying parameters of the deep belief network. Ant lion algorithm was used by Li and Jin to determine the parameters of an extreme learning machine [43]. In the ocean-wave forecasting model, Cornejo-Bueno et al. utilized Bayesian optimization to determine ELM parameters. Zameer et al. designed a new neural network-based short-term wind-energy prediction approach using genetic programming for parameter estimation [31]. Papari et al. using an optimization technique named modified harmony search optimization to specify parameters of support vector regression. This combination shows better performances in comparison to other models [102].

The parameter selection of the machine learning model used for renewable energy projection is shown in Fig. 12.2.

3 Performance Evaluation of Forecasted Model

This part explains the results obtained by the forecasting model using evaluation metrics. Table 12.4 shows the evaluation of forecasted model's accuracy by more than 11 studies. It shows that most researchers used Mean absolute error, mean absolute percentage error, root mean square error to measure the effectiveness of the forecasted model. Table 12.5 represents the measurement of the forecasted model using only MAPE. As per Lewis [103] if the MAPE value lies under 10% it shows a highly reliable forecast. There is another measurement criterion used to determine the accuracy of a model which is the correlation coefficient (R^2). R^2 used to determine the correlation between measured and predicted value. Table 12.6 represents the status of 23 studies using R^2 as error metrics. Table 12.7 shows the performance of eight studies that used MAPE and R^2 simultaneously. It describes that if value of

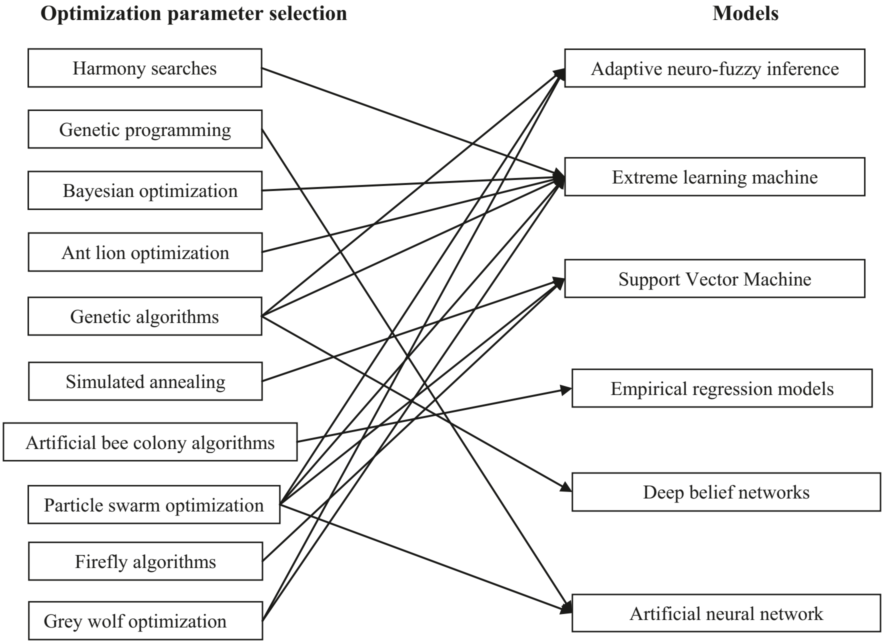


Fig. 12.2 Machine Learning Parameter Selection metaheuristics for renewable energy forecast

Table 12.6 R^2 value by various studies for renewable energy resources

Energy sources	R^2/Ref. No.
Solar energy	0.70[57],0.98[67],0.99[74],0.9[75],0.93[76],0.99[91], 0.96[105]
Wind energy	0.99[14],0.99[16],0.99[37],0.87[38],0.95[42],0.99[113],0.98[114]
Biomass	0.99[80],0.93[81],0.90[83]
Geothermal	0.8[115]
Wave	0.78[85]
Hydropower	0.7[78]

Table 12.7 MAPE and R^2 Value by various metrics

Energy source	Reference No.	R^2	MAPE (%)
Wind	[38]	0.87	17.1
Wind	[42]	0.95	8.6
Solar	[74]	0.99	2.8
Biomass	[81]	0.93	3.7
Solar	[91]	0.99	0.2
Solar	[105]	0.96	7.4
Wind	[113]	0.99	3.7
Wind	[114]	0.98	7.5

the correlation coefficient is high then the MAPE value is lower which represents the more accurate and reliable renewable energy forecasting model.

ES = Energy Sources; EM = Evaluation Metrics; MSE = Mean Square Error; NRMSE = Normalized Root Mean Square Error; CC = Correlation Coefficient; RMSE = Root Mean Square Error; MAPE = Mean Absolute Percentage Error; MAE = Mean Absolute Error.

4 Conclusion

The usage of renewable energy is growing every day in order to reduce greenhouse gas emissions, but forecasting renewable energy is a difficult issue. The use of machine learning models to anticipate renewable energy in various ways was investigated in this paper. In the first section, the machine learning model is categorized into three approaches: mathematical model, artificial intelligence approach, and ensemble model. Among these three, the artificial intelligence approach and ensemble model are used by the majority of researchers. In the second case, we describe the various preprocessing techniques to increase the performance of the forecasted machine was given. Thirdly, we describe the various parameter selection techniques for the machine learning model in the renewable energy system and finally forecasting model's effectiveness is measured using a variety of evaluation indicators. For future prospects, some observation is described as follows: In this review, it can be seen that most researchers are working on the forecasting of solar and wind energy but other renewable energy sources such as tidal, geothermal, biomass, and wave could also play an important role for future work. Secondly, preprocessing techniques enhance the performance of the forecasted model. However, this subject has received little consideration thus far. Thus, the exploration of preprocessing techniques with machine learning models may be a future research subject. Thirdly, appropriate selection of parameters also affects the accuracy of the forecasting machine. Thus, to increase the performance of the machine learning model the use of a new coronavirus optimization technique for parameter selection may be set up for future encouraging work for researchers.

References

1. Wang, H. Z., Lei, Z. X., & Zhang, X. (2019). A review of deep learning for renewable energy forecasting. *Energy Conversion and Management, 198*, 111799.
2. Olabi, A. G. (2017). Renewable and energy storage system. *Energy, 136*, 1–6.
3. Zendehboudi, A., Baseer, M. A., & Saidur, R. (2018). Application of support vector machine models for forecasting solar and wind energy resources: A review. *Journal of Cleaner Production, 199*, 272–285.
4. Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica, 31*, 249–268.

5. Qiu, J., Wu, Q., Ding, G., Xu, Y., & Feng, S. (2016). A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing, 2016*, 67.
6. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G., et al. (2017). Recent advances in convolutional neural networks. *Pattern Recognition, 1*, 1–24.
7. Amasyali, K., & El-Gohary, N. M. (2018). A review of data-driven building energy consumption prediction studies. *Renewable and Sustainable Energy Reviews, 81*, 1192–1205.
8. Voyant, C., Notton, G., Kalogirou, S., Nivet, M.-L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable Energy, 105*, 569–582.
9. Das, U. K., Tey, K. S., Seyedmahmoudian, M., Mekhilef, S., Idris, M. Y. I., Deventer, W. V., Horan, B., & Stojcevski, A. (2018). Forecasting of photovoltaic power generation and model optimization: A review. *Renewable and Sustainable Energy Reviews, 81*, 912–928.
10. Ahmed, A., & Khalid, M. (2019). A review on the selected applications of forecasting models in renewable power systems. *Renewable and Sustainable Energy Reviews, 100*, 9–21.
11. Mosavi, A., Salimi, M., Ardabili, S. F., Rabczuk, T., Shamshirband, S., & Varkonyi-Koczy, A. R. (2019). State of the art of machine learning models in energy systems, a systematic review. *Energies, 12*, 130.
12. Bermejo, J.F.; Fernandez, J.F.G.; Polo,F.O.;Marquez, A.C.A review of the use of artificial neural network models for energy and reliability prediction. A study of the solar PV, hydraulic and wind energy sources. Applied Sciences 2019, 9, 1844.
13. Mellit, A., Massi Pavan, A., Ogliari, E., Leva, S., & Lughi, V. (2020). Advanced methods for photovoltaic output power forecasting: A review. *Applied Sciences, 10*, 487.
14. Sharifzadeh, M., Sikinioti-Lock, A., & Shah, N. (2019). Machine-learning methods for integrated renewable power generation: A comparative study of artificial neural networks, support vector regression, and Gaussian process regression. *Renewable and Sustainable Energy Reviews, 108*, 513–538.
15. Howland, M. F., & Dabiri, J. O. (2019). Wind farm modeling with interpretable physics-informed machine learning. *Energies, 12*, 2716.
16. Demolli, H., Dokuz, A. S., Ecemis, A., & Gokcek, M. (2019). Wind power forecasting based on daily wind speed data using machine learning algorithms. *Energy Conversion and Management, 198*, 111823.
17. Howland, M. F., & Dabiri, J. O. (2019). Wind farm modeling with interpretable physics-informed machine learning. *Energies, 12*, 2716.
18. Khan, M., Liu, T., & Ullah, F. (2019). A new hybrid approach to forecast wind power for large scale wind turbine data using deep learning with tensorflow framework and principal component analysis. *Energies, 12*, 2229.
19. Ma, Y.-J., & Zhai, M.-Y. (2019). A dual-step integrated machine learning model for 24h-ahead wind energy generation prediction based on actual measurement data and environmental factors. *Applied Sciences, 9*, 2125.
20. Mujeeb, S., Alghamdi, T. A., Ullah, S., Fatima, A., Javaid, N., & Saba, T. (2019). Exploiting deep learning for wind power forecasting based on big data analytics. *Applied Sciences, 9*, 4417.
21. Fischetti, M., & Fraccaro, M. (2019). Machine learning meets mathematical optimization to predict the optimal production of offshore wind parks. *Computers and Operations Research, 106*, 289–297.
22. Harbola, S., & Coors, V. (2019). One dimensional convolutional neural network architectures for wind prediction. *Energy Conversion and Management, 195*, 70–75.
23. Srpak, D., Havaš, L., Skok, S., Polajžer, B. (2019). Reducing wind power forecast error based on machine learning algorithms and producers merging. In *Proceedings of the 2019 IEEE international conference on environment and electrical engineering and 2019 IEEE industrial and commercial power systems Europe*, Genoa, Italy, 11–14 June 2019.

24. Yu, R., Liu, Z., Li, X., Lu, W., Ma, D., Yu, M., Wang, J., & Li, B. (2019). Scene learning: Deep convolutional networks for wind power prediction by embedding turbines into grid space. *Applied Energy, 238*, 249–257.
25. Mujeeb, S., Javaid, N., Gul, H., Daood, N., Shabbir, S., Arif, A. (2019) Wind power forecasting based on efficient deep convolution neural networks. In *Proceedings of the 3PGCIC Conference, Antwerp*, Belgium, 7–9 November 2019.
26. Jiao, R., Huang, X., Ma, X., Han, L., & Tian, W. (2018). A model combining stacked auto encoder and back propagation algorithm for short-term wind power forecasting. *IEEE Access, 6*, 17851–17858.
27. Huang, C. J., & Kuo, P. H. (2018). A short-term wind speed forecasting model by using artificial neural networks with stochastic optimization for renewable energy systems. *Energies, 11*, 2777.
28. Zhu, Q., Chen, J., Zhu, L., Duan, X., & Liu, Y. (2018). Wind speed prediction with spatio–temporal correlation: A deep learning approach. *Energies, 11*, 705.
29. Chang, G. W., Lu, H. J., Chang, Y. R., & Lee, Y. D. (2017). An improved neural network-based approach for short-term wind speed and power forecast. *Renewable Energy, 105*, 301–311.
30. Wang, H. Z., Li, G. Q., Wang, G. B., Peng, J. C., Jiang, H., & Liu, Y. T. (2017). Deep learning based ensemble approach for probabilistic wind power forecasting. *Applied Energy, 188*, 56–70.
31. Zameer, A., Arshad, J., Khan, A., & Raja, M. A. Z. (2017). Intelligent and robust prediction of short term wind power using genetic programming based ensemble of neural networks. *Energy Conversion and Management, 134*, 361–372.
32. Eze, E. C., & Chatwin, C. R. (2019). Enhanced recurrent neural network for short-term wind farm power output prediction. *Journal of Applied Sciences, 5*, 28–35.
33. Yu, R., Gao, J., Yu, M., Lu, W., Xu, T., Zhao, M., Zhang, J., Zhang, R., & Zhang, Z. (2019). LSTM-EFG for wind power forecasting based on sequential correlation features. *Future Generation Computer Systems, 93*, 33–42.
34. Cardona, J. L., Howland, M. F., & Dabiri, J. O. (2019). Seeing the wind: Visual wind speed prediction with a coupled convolutional and recurrent neural network. *arXiv*, arXiv:1905.13290.
35. Cali, U., & Sharma, V. (2019). Short-term wind power forecasting using long-short term memory based recurrent neural network model and variable selection. *International Journal of Smart Grid and Clean Energy, 8*, 103–110.
36. Zhang, J., Jiang, X.; Chen, X., Li, X.; Guo, D., Cui, L. (2019, April 1) Wind power generation prediction based on LSTM. In Proceedings of the 4th International Conference on Mathematics and Artificial Intelligence, Chegndu China, pp. 85–89.
37. Zhang, Z., Ye, L., Qin, H., Liu, Y., Wang, C., Yu, X., Yin, X., & Li, J. (2019). Wind speed prediction method using shared weight long short-term memory network and gaussian process regression. *Applied Energy, 247*, 270–284.
38. Chen, J., Zeng, G., Zhou, W., Du, W., & Lu, K. (2018). Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization. *Energy Conversion and Management, 165*, 681–695.
39. Khosravi, A., Machado, L., & Nunes, R. (2018). Time-series prediction of wind speed using machine learning algorithms: A case study Osorio wind farm, Brazil. *Applied Energy, 224*, 550–566.
40. Feng, C., Cui, M., Hodge, B.-M., & Zhang, J. (2017). A data-driven multi-model methodology with deep feature selection for short-term wind forecasting. *Applied Energy, 190*, 1245–1257.
41. Lahouar, A., & Slama, J. B. (2017). Hour-ahead wind power forecast based on random forests. *Renewable Energy, 109*, 529–541.
42. Li, L. L., Zhao, X., Tseng, M. L., & Tan, R. R. (2020). Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. *Journal of Cleaner Production, 242*, 118447.

43. Lin, K. P., Pai, P. F., & Ting, Y. J. (2019). Deep belief networks with genetic algorithms in forecasting wind speed. *IEEE Access, 7*, 99244–99253.
44. Sharifian, A., Ghadi, M. J., Ghavidel, S., Li, L., & Zhang, J. (2018). A new method based on type-2 fuzzy neural network for accurate wind power forecasting under uncertain data. *Renewable Energy, 120*, 220–230.
45. Li, R., & Jin, Y. (2018). A wind speed interval prediction system based on multi-objective optimization for machine learning method. *Applied Energy, 228*, 2207–2220.
46. Santhosh, M., Venkaiah, C., & Kumar, D. M. V. (2019). Short-term wind speed forecasting approach using ensemble empirical mode decomposition and deep Boltzmann machine. *Sustainable Energy Grids and Networks, 19*, 100242.
47. Wang, L., Li, X., & Bai, Y. (2018). Short-term wind speed prediction using an extreme learning machine model with error correction. *Energy Conversion and Management, 162*, 239–250.
48. Wang, G., Jia, G., Liu, J., & Zhang, H. (2020). A hybrid wind power forecasting approach based on Bayesian model averaging and ensemble learning. *Renewable Energy, 145*, 2426–2434.
49. Wang, Y., Wang, H. B., Srinivasan, D., & Hu, Q. H. (2019). Robust functional regression for wind speed forecasting based on sparse Bayesian learning. *Renewable Energy, 132*, 43–60.
50. Sun, S., Qiao, H., Wang, S., & Wei, Y. (2017). A new dynamic integrated approach for wind speed forecasting. *Applied Energy, 197*, 151–162.
51. Liu, H., Mi, X., & Li, Y. (2018). Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network. *Energy Conversion and Management, 166*, 120–131.
52. Atique, S.; Noureen, S.; Roy, V.; Subburaj, V.; Bayne, S.; Macfie, J. (2019). Forecasting of total daily solar energy generation using ARIMA: A case study. In *Proceedings of the 2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*, Las Vegas, NV, USA, 7–9 January 2019; pp. 114–119.
53. Alsharif, M. H., Younes, M. K., & Kim, J. (2019). Time series ARIMA model for prediction of daily and monthly average global solar radiation: The case study of Seoul, South Korea. *Symmetry, 11*, 240.
54. Aslam, M., Lee, J.-M., Kim, H.-S., Lee, S.-J., & Hong, S. (2019). Deep learning models for long-term solar radiation forecasting considering microgrid installation: A comparative study. *Energies, 13*, 147.
55. Rana, M., & Rahman, A. (2020). Multiple steps ahead solar photovoltaic power forecasting based on univariate machine learning models and data re-sampling. *Sustainable Energy Grids and Networks, 21*, 100286.
56. Bajpai, A., Duchon, M. (2019) A hybrid approach of solar power forecasting using machine learning. In Proceedings of the 3rd international conference on smart grid and smart cities, Berkeley, CA, USA, 25–28 June 2019.
57. Kim, S. G., Jung, J. Y., & Sim, M. K. (2019). A two-step approach to solar power generation prediction based on weather data using machine learning. *Sustainability, 11*, 1501.
58. Torres-Barrán, A., Alonso, Á., & Dorronsoro, J. R. (2019). Regression tree ensembles for wind energy and solar radiation prediction. *Neurocomputing, 326*, 151–160.
59. Saloux, E., & Candanedo, J. A. (2018). Forecasting district heating demand using machine learning algorithms. *Energy Procedia, 149*, 59–68.
60. Feng, C., & Zhang, J. (2018). Hourly-similarity based solar forecasting using multi-model machine learning blending. *arXiv*, arXiv:1803.03623.
61. Zambrano, A. F., & Giraldo, L. F. (2020). Solar-irradiance forecasting models without on-site training measurements. *Renewable Energy, 152*, 557–566.
62. Khandakar, A., Chowdhury, M. E. H., Khoda Kazi, M., Benhmed, K., Touati, F., Al-Hitmi, M., & Gonzales, J. S. (2019). Machine learning based photovoltaics (PV) power prediction using different environmental parameters of Qatar. *Energies, 12*, 2782.

63. Dorado-Moreno, M., Navarin, N., Gutierrez, P. A., Prieto, L., Sperduti, A., Salcedo-Sanz, S., & Hervas-Martinez, C. (2020). Multi-task learning for the prediction of wind power ramp events with deep neural networks. *Neural Networks, 123*, 401–411.
64. Rodríguez, F., Fleetwood, A., Galarza, A., & Fontán, L. (2018). Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control. *Renewable Energy, 126*, 855–864.
65. Leva, S., Dolara, A., Grimaccia, F., Mussetta, M., & Sahin, E. (2017). Analysis and validation of 24 hours ahead neural network forecasting for photovoltaic output power. *Mathematics and Computers in Simulation, 131*, 88–100.
66. Sun, Y., Venugopal, V., & Brandt, A. R. (2019). Short-term solar power forecast with deep learning: Exploring optimal input and output configuration. *Solar Energy, 188*, 730–741.
67. Camila, C. J., Cardemil, J. M., Droguett, E. L., & Behzad, M. (2020). Assessment of deep learning techniques for prognosis of solar thermal systems. *Renewable Energy, 145*, 2178–2191.
68. AlKandari, M., & Ahmad, I. (2019). Solar power generation forecasting using ensemble approach based on deep learning and statistical methods. *Applied Computing and Informatics*.
69. Chen, J., Yu, J., Song, M., & Valdmanis, V. (2019). Factor decomposition and prediction of solar energy consumption in the United States. *Journal of Cleaner Production, 234*, 1210–1220.
70. Wen, L. L., Zhou, K. L., Yang, S. L., & Lu, X. H. (2019). Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting. *Energy, 171*, 1053–1065.
71. Qing, X., & Niu, Y. (2018). Hourly day-ahead solar irradiance prediction using weather forecasts by lstm. *Energy, 148*, 461–468.
72. Abdel-Nasser, M., & Mahmoud, K. (2017). Accurate photovoltaic power forecasting models using deep LSTM-RNN. In *Neural computing and applications* (pp. 1–14). Springer.
73. Monjoly, S., André, M., Calif, R., & Soubdhan, T. (2017). Hourly forecasting of global solar radiation based on multiscale decomposition methods: A hybrid approach. *Energy, 119*, 288–298.
74. Lin, G.-Q., Li, L.-L., Tseng, M.-L., Liu, H.-M., Yuan, D.-D., & Tan, R. (2020). An improved moth-flame optimization algorithm for support vector machine prediction of photovoltaic power generation. *Journal of Cleaner Production, 253*, 119966.
75. Liu, Y., Zhou, Y., Chen, Y., Wang, D., Wang, Y., & Zou, Y. (2020). Comparison of support vector machine and copula-based nonlinear quantile regression for estimating the daily diffuse solar radiation: A case study inChina. *Renewable Energy, 146*, 1101–1112.
76. Fan, J., Wu, L., Ma, X., Zhou, H., & Zhang, F. (2020). Hybrid support vector machines with heuristic algorithms for prediction of daily diffuse solar radiation in air-polluted regions. *Renewable Energy, 145*, 2034–2045.
77. Torabi, M., Mosavi, A., Ozturk, P., Varkonyi-Koczy, A., & Istvan, V. (2018). A hybrid machine learning approach for daily prediction of solar radiation. In *Recent advances in technology research and education* (pp. 266–274). Springer.
78. Dehghani, M., Riahi-Madvar, H., Hooshyaripor, F., Mosavi, A., Shamshirband, S., Zavadskas, E. K., & Chau, K.-W. (2019). Prediction of hydropower generation using grey wolf optimization adaptive neuro-fuzzy inference system. *Energies, 12*, 289.
79. Shayan, E., Zare, V., & Mirzaee, I. (2018). Hydrogen production from biomass gasification; a theoretical comparison of using different gasification agents. *Energy Conversion and Management, 159*, 30–41.
80. Ozbas, E. E., Aksu, D., Ongen, A., Aydin, M. A., & Ozcan, H. K. (2019). Hydrogen production via biomass gasification, and modeling by supervised machine learning algorithms. *International Journal of Hydrogen Energy, 44*, 17260–17268.
81. Samadi, S. H., Ghobadian, B., & Nosrati, M. (2019). Prediction of higher heating value of biomass materials based on proximate analysis using gradient boosted regression trees method. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*.

82. Elmaz, F., Yücel, O., & Mutlu, A. Y. (2019). Predictive modeling of biomass gasification with machine learning-based regression methods. *Energy, 191*, 116541.
83. García Nieto, P. J., García-Gonzalo, E., Sánchez Lasheras, F., Paredes-Sánchez, J. P., & Riesgo Fernández, P. (2019). Forecast of the higher heating value in biomass torrefaction by means of machine learning techniques. *Journal of Computational and Applied Mathematics, 357*, 284–301.
84. Ali, M., & Prasad, R. (2019). Significant wave height forecasting via an extreme learning machine model integrated with improved complete ensemble empirical mode decomposition. *Renewable and Sustainable Energy Reviews, 104*, 281–295.
85. Cornejo-Bueno, L., Garrido-Merchán, E. C., Hernández-Lobato, D., & Salcedo-Sanz, S. (2018). Bayesian optimization of a hybrid system for robust ocean wave features prediction. *Neurocomputing, 275*, 818–828.
86. Kavousi-Fard, A., & Su, W. (2017). A combined prognostic model based on machine learning for tidal current prediction. *IEEE Transactions on Geoscience and Remote Sensing, 55*, 3108–3114.
87. Michael, D., Thomas, A., & Adcock, A. (2018). Prediction of tidal currents using Bayesian machine learning. *Journal of Ocean Engineering, 158*, 221–231.
88. Safari, N., Ansari, O.A., Zare, A., Chung, C.Y. (2017). A novel decomposition-based localized short-term tidal current speed and direction prediction model. In *Proceedings of the IEEE Power & Energy Society General Meeting*, Chicago, IL, USA, 16–20 July 2017.
89. Gangwani, P., Soni, J., Upadhyay, H., & Joshi, S. (2020). A deep learning approach for modeling of geothermal energy prediction. *Computer Science and Information Security*, 62–65.
90. Baruque, B., Porras, S., Jove, E., & Calvo-Rolle, J. L. (2019). Geothermal heat exchanger energy prediction based on time series and monitoring sensors optimization. *Energy, 171*, 49–60.
91. Rohani, A., Taki, M., & Abdollahpour, M. (2018). A novel soft computing model (Gaussian process regression with K-fold cross validation) for daily and monthly solar radiation forecasting (Part: I). *Renewable Energy, 115*, 411–422.
92. Liu, H., Tian, H. Q., & Li, Y. F. (2015). Four wind speed multi-step forecasting models using extreme learning machines and signal decomposing algorithms. *Energy Conversion and Management, 100*, 16–22.
93. Hamed, H. H. A. (2019). A novel approach for harmonic tidal currents constitutions forecasting using hybrid intelligent models based on clustering methodologies. *Renewable Energy, 147*, 1554–1564.
94. McCandless, T. C., Dettling, S., & Haupt, S. E. (2020). Comparison of implicit vs. explicit regime identification in machine learning methods for solar irradiance prediction. *Energies, 13*, 689.
95. Sun, S., Wang, S., Zhang, G., & Zheng, J. (2018). A decomposition-clustering-ensemble learning approach for solar radiation forecasting. *Solar Energy, 163*, 189–199.
96. Wang, Y., Shen, Y., Mao, S., Chen, X., & Zou, H. (2018). LASSO & LSTM integrated temporal model for short-term solar intensity forecasting. *IEEE Internet of Things Journal, 6*, 2933–2944.
97. Fatih Onur, H., & Serttas, F. (2016). A novel hybrid (Mycielski-Markov) model for hourly solar radiation forecasting. *Renewable Energy, 108*, 635–643.
98. Wu, C., Wang, J., Chen, X., Du, P., & Yang, W. (2019). A novel hybrid system based on multi-objective optimization for wind speed forecasting. *Renewable Energy, 146*, 149–165.
99. Feng, Y., Hao, W., Li, H., Cui, N., Gong, D., & Gao, L. (2020). Machine learning models to quantify and map daily global solar radiation and photovoltaic power. *Renewable and Sustainable Energy Reviews, 118*, 109393.
100. Galván, I. M., Valls, J. M., Cervantes, A., & Aler, R. (2017). Multi-objective evolutionary optimization of prediction intervals for solar energy forecasting with neural networks. *Information Sciences, 418*, 363–382.

101. Demircan, C., Bayrakçı, H. C., & Keçeba¸s, A. (2020). Machine learning-based improvement of empiric models for an accurate estimating process of global solar radiation. *Sustainable Energy Technologies and Assessments, 37*, 100574.
102. Papari, B., Edrington, C. S., & Kavousi-Fard, F. (2017). An effective fuzzy feature selection and prediction method for modeling tidal current: A case of Persian gulf. *IEEE Transactions on Geoscience and Remote Sensing, 55*, 4956–4961.
103. Lewis, C. D. (1982). *Industrial and business forecasting methods*. Butterworth Scientific.
104. Leholo, S.; Owolawi, P.; Akindeji, K. Solar energy potential forecasting and optimization using artificial neural network- South Africa case study. In Proceedings of the Amity International Conference on Artificial Intelligence, Dubai, UAE, 4–6 February 2019.
105. Bouzgou, H., & Gueymard, C. A. (2019). Fast short-term global solar irradiance forecasting with wrapper mutual information. *Renewable Energy, 133*, 1055–1065.
106. Wang, Z.; Koprinska, Z.W.I.; Koprinska, I.; Troncoso, A.; Martínez-Álvarez, F. Static and dynamic ensembles of neural networks for solar power forecasting. In *Proceedings of the international joint conference on neural networks (IJCNN)*, IEEE, Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.
107. Yousif, J. H., Kazem, H. A., Alattar, N. N., & Elhassan, I. I. (2019). A comparison study based on artificial neural network for assessing PV/T solar energy production. *Case Studies in Thermal Engineering, 13*, 1–13.
108. Sapitang, M., Ridwan, W. M., Kushiar, K. F., Ahmed, A. N., & El-Shafie, A. (2020). Machine learning application in reservoir water level forecasting for sustainable hydropower generation strategy. *Sustainability, 12*, 6121.
109. Zhou, J., Sun, N., Jia, B., & Peng, T. (2018). A novel decomposition-optimization model for short-term wind speed forecasting. *Energies, 11*, 1752.
110. Luo, X., Sun, J., Wang, L., Wang, W., Zhao, W., Wu, J., Wang, J. H., & Zhang, Z. (2018). Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy. *IEEE Transactions on Industrial Informatics, 14*, 4963–4971.
111. Liu, H., Mi, X., & Li, Y. (2018). Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM. *Energy Conversion and Management, 159*, 54–64.
112. Liu, H., Mi, X. W., & Li, Y. F. (2018). Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and Elman neural network. *Energy Conversion and Management, 156*, 498–514.
113. Zhang, X., Peng, Y., Xu, W., & Wang, B. (2017). An optimal operation model for hydropower stations considering inflow forecasts with different Lead-times. *Water Resources Management, 33*, 173–188.
114. Nieto, P. G., Garcia-Gonzalo, E., Paredes-Sánchez, J. P., Sánchez, A. B., & Fernández, M. M. (2019). Predictive modeling of the higher heating value in biomass torrefaction for the energy treatment process using machine-learning techniques. *Neural Computing and Applications, 31*, 8823–8836.
115. Avila, D., Marichal, G. N., Padrón, I., Quiza, R., & Hernández, Á. (2020). Forecasting of wave energy in Canary Islands based on Artifcial intelligence. *Applied Ocean Research, 101*, 102189.
116. Shi, S., Patton, R. J., & Liu, Y. (2018). Short-term wave forecasting using gaussian process for optimal control of wave energy converters. *IFAC PapersOnLine, 51*, 44–49.

Chapter 13
Security and Privacy Threats in IoT-Enabled Smart Cities

Aditya Sam Koshy, Nida Fatima, Parul Agarwal, and Joel J. P. C. Rodrigues

Abstract Presently, the globe is going through development in smart cities as the Internet of Things (IoT) is capable of connecting with almost every object in the environment through sensors, cloud, end user devices, and user interface. This is possible because of the uprising in information technology day by day, thereafter contributing to the social and economic welfare of citizens. The Internet is becoming more integrated in daily life as managed by the Internet of Things. As IoT devices embrace novel endless opportunities to make life easier for people, they also increase the risk of data breaches, unsuspected users, and malicious attacks in the IoT framework. So preserving security and privacy from threats and attacks is an endowing challenge that is faced by IoT devices. It is known to an extent that data are more vulnerable in terms of security and privacy, because of integrated features implemented in the Internet of Things. Consequently, it is more prone to cyber threats and attacks despite the fact that it brings unbounded comfort and social security. Resolving these challenges and promising security and privacy of information against threats and attacks should be the prime priority while designing and implementing the architecture of Internet of things (IoT). However, the end user ought to trust that IoT devices and services provided by them are safe and secure. IoT devices protection needs to be considered while designing the framework, to protect it from any kind of threat and attack while keeping in mind ethics and poli-

A. S. Koshy · N. Fatima (✉) · P. Agarwal
Department of Computer Science & Engineering, Jamia Hamdard University, New Delhi, India
e-mail: pagarwal@jamiahamdard.ac.in

J. J. P. C. Rodrigues
College of Computer Science and Technology, China University of Petroleum (East China), Qingdao, China

Instituto de Telecomunicações, Covilhã, Portugal
e-mail: joeljr@ieee.org

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_13

cies which are utilized by the Internet of Things mechanization. In this chapter, privacy and security issues are discussed along with threats and attacks. Overview of the Interest of Things (IoT) is discussed in the beginning. Research work done in the past is also reviewed and examined with state of the art whereupon highlighting techniques utilized including the objectives and limitations. Additionally, this chapter covers layers of the Internet of Things (IoT) with the network of IoT and security and challenges in each layer of IoT architecture. Major security issues are also considered, followed by a discussion about Smart City Applications with their threats and solutions precisely to preserve concealment of Internet of Things (IoT) devices.

Keywords Smart city · Threats · Attacks · Layers of IoT · Smart

1 Introduction

The Internet of Things (IoT) is taking the world by storm. According to [1], the number of connected IoT devices is going to be over 25.4 billion in 2030. The sheer number tends to tell the success story of IoT as a technology and business opportunity. IoT is a network of devices that exchange data amongst themselves in order to provide better services. The IoT creates this huge network of different devices that aim to provide different services that make human lives easier. The involvement of IoT in our lives has opened up a world of opportunities, with so many possibilities to achieve a number of things. IoT finds its application in many fields such as infrastructure, traffic, agriculture, healthcare, business, and environment. In 2019, according to an estimate, 86% of the healthcare service companies were using IoT in some capacity or manner [1]. The IoT agriculture market is expected to value at $48,714 million in 2025 [2]. All these applications tend to improve our state of living, providing us with the best of the service. IoT finds a major role in data collection through various sensory devices and its analysis. All this points to the potential that IoT has for making a change that would far exceed a common mans' fantasy. It is all the more reason why big companies such as Apple, Microsoft, and Huawei. all want in on the action. According to [3], the total number of individual IoT patents held by Samsung Electronics is 9550. This reinforces the fact that all tech giants are well aware of the IoT, and its tendencies and how important it's going to be. With all the possibilities to improve the current state of many services and industries, the one prevailing issue that has always been of concern is security. It is necessary for us to understand the security aspect of the IoT. According to [4], the IoT saw a hike of 300% in the number of attacks on IoTs. This is a huge matter of concern since IoT is an incredible technology, but what good would it be if it is not safe. Researchers over the years have been working on IoT to build up its walls of defense and countermeasures to fend off attempts of attacks.

There are various approaches that have been taken to understand the threats that revolve around IoT, weaknesses, implementation vulnerabilities, design flaws, etc.

In this chapter we have taken the layered approach to understanding in-depth the prominent threats that linger around them. We discuss the functions of each of the layers and types of attacks that are common in them along with possible ways to counter them. We also discuss the major threats in various IoT applications such as Smart city and Smart traffic.

This chapter has been divided into eight sections. Sections 2 and 3 provide literature review and state-of-the-art techniques in IoT privacy and security. Section 4 describes the layered architecture of IoT. Section 5 describes different IoT threats based on its different layers. Section 6 defines different types of threats posed to the IoT network based on two different transmission technologies, which are short-range and long-range transmission technology. Section 7 provides an insight on different smart application–based threats on IoT and the conclusion is presented in Sect. 8.

2 Literature Review

Large number of researchers have focused on securing the Internet of Things (IoT) system. With IoT data breaches, and malicious activity are a common aspect. In this section, some of the previously done work are evaluated and inspected which focus on securing IoT devices from any type of threats and attacks. The author in [5] has put forward a requirement for an engineering framework for validating security and privacy. Moreover, he has suggested that data protection techniques should be amalgamated for the purpose of security. In [6], authors talked about a 4-layered IoT security framework with respect to security requirements of each of the IoT components which include sensors, network, cloud, and base stations along with security threats. Additionally, in [7], the authors examine security conditions including security threats corresponding to the Internet of Things (IoT). It has also been recommended for maintaining security, network heap needs to meet protocols which are obligatory for ensuring security addition to its heterogeneous network to be interoperable. In [8], security threats, privacy issues, and challenges are discussed by the author. For making data secure, authentication, confidentiality, and data integrity should be adopted. Furthermore, in [9], it is emphasized that grouped intrusion detection, non-repudiation prevention mechanisms like accessibility control, confidentiality, and data integrity are needed in any IoT framework. In [10], the author studies a variety of IoT devices and examines privacy and security threats in each device. For achieving security, encryption algorithms, access control, data protection policy, intrusion detection at the earliest, timely monitoring of IoT devices, confidentiality, and identification are necessary. In [11], the author discusses security and privacy challenges and reviews security issues based on sensors, services provided by IoT devices, network, and cloud together with application-layered architecture in an IoT environment. Furthermore, authors in [12] discuss that as data is collected pervasively, information can be easily hacked. As a result, vulnerability arises and this is a situation of concern for people as there is a threat of data breach.

So in the context of IoT, safeguarding the privacy of citizens is a strenuous piece of work. In [13], the authors mention that with the growth in the IoT's new sensors, devices are being introduced which collect personal data of people in the form of personal, financial, or other related information, sadly, most of it takes place without the knowledge of the person. Therefore, protecting data against any malicious activity is a tough task. Author in [14] states that for protection of system, proper authentication, authorization, and accessibility methods are mandatory; thereby, only permissioned applications and services are allowed to gain access over the information. It is also proposed to follow two-factor authentication for making devices invulnerable. In [15], author researches about RF Interference on RFID as the IoT sensors are more prone to this kind of attack. In this technique, the intruder makes use of the Radio Frequency Identification (RFID) tag to disrupt or create noise in RF signals which result in deterioration of the quality of communication. In [16], the author presents that devices are at a higher risk of being vulnerable to network layer, most commonly to Distributed Denial of Service (DDoS) attack. So security prevention measures and mechanisms need to be implemented for early detection, so that layers can be guarded in a timely manner. Moreover, [17] suggested the need for strong encryption algorithms along with a key agreement to protect data. In [18], it is identified that information between client and server should be guaranteed against any malicious activity; only verified users are permitted so that data is secured from any kind of theft or vulnerability. In [19], the need for well-grounded security measures for overcoming security and privacy issues in the Internet of Things (IoT) is mentioned. In [20], researchers put forward that while designing architecture for the IoT systems, authenticity and embedded security ought to be considered.

3 State-of-the-Art Network Security Techniques

Following is the Table 13.1 that shows various State-of-the-art network security techniques being used today:

4 Layers of IoT

Over the years, there have been many architecture models proposed for IoT based on layers. It is a common understanding that dividing the architecture into layers allows for better understanding and learning of the architecture. It is to be understood that all these architectures essentially point to the same network; the difference arises only in the approach. Many scholars and researchers for the same reason have gone on to provide variable number of layers to the same network based on its functioning, properties, security, etc. In [28], the Cisco company proposed a reference model for the IoT network that comprises of 7 layers. It can be argued that this

Table 13.1 State-of-the-art network security techniques

Author	Technique	Purpose	Limitation
[21]	PiOS method	It determines whether there is any information transgression in the IOS application.	Client is unclear about the privacy breach. It is not an effective method for consideration by the end user.
[22]	TaintDroid method	It spots if there is a risk to the client's android application and if sensitive data is breached.	If any infringement occurs it is not detected on time, as safeguarding information on time is impractical.
[23]	User-ware privacy control approach	For protecting privacy it utilizes user-driven access control method which only permits authorized users to access data.	Implicit control flow cannot be tackled.
[24]	Kirin method	Uncertain policy configuration in the android framework is detected by which install time decisions for users defer.	Not capable of identifying data leakage, also cannot provide reliability against automatic security.
[25]	R- IoT	For protecting privacy, the key is used for the exchange of information that permits the user and gateway by which data are made secure from several attacks.	Protocol might not be available at the time of intermittent connectivity.
[26]	ICAC	Data are preserved by using CAC and SHA – 1.	For constrained devices, performance has not been checked.
[27]	SM – EAPOL	Provides secure end-to-end transmission between constrained devices.	EAPOL is required with DTLS, which is not suitable for constrained devices.

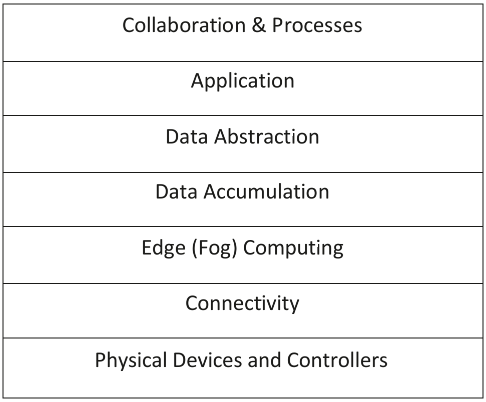


Fig. 13.1 Cisco IoT reference architecture

is a very finely designed architecture of IoT that delves deeply into functionality and purpose of each component that is a part of the IoT structure.

Figure 13.1 shows the reference model that was proposed by Cisco. This architecture divides the whole network into 7 layers, each specific in its functioning and responsibilities. The bottom most layer consists of physical devices and controllers. This is the layer that consists of devices such as sensors and actuators. These devices are the ones that are responsible for creating the data that is later put to use. Above this layer is the connectivity layer. This layer deals with the communication part of the network. The transmission of data through various means between the devices and base stations come under this layer. Edge computing is a relatively new concept that refers to a decentralized approach toward the processing of data. The edge computing layer aims at processing the data that is generated at the device level and transported by the connectivity layer, for further layers that are above it. Fog computing aims to process data as close to the network as possible so as to provide services in a quicker fashion.

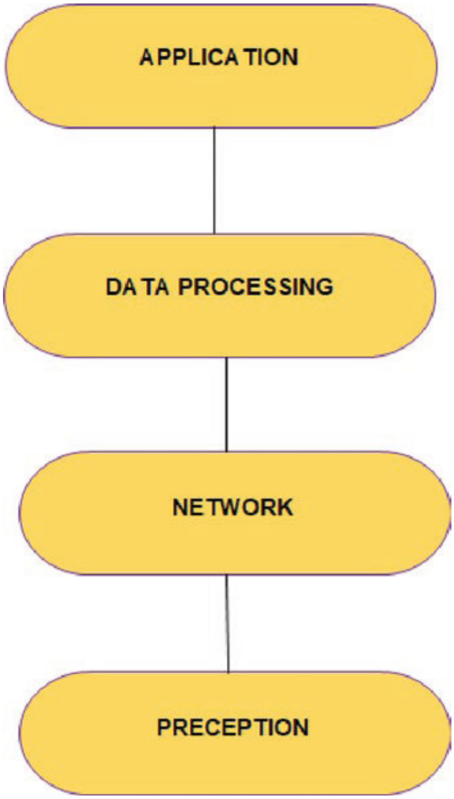
The data accumulation layer holds the responsibility that is well established in its name. The data are stored and made available for access through different applications at any time, preferably. The data here is at a resting state and can be used to gather information or any other purpose that befits the need. The data abstraction layer deals with arranging the data in a format such that it is easily accessed by applications. With a hoard of devices at device level, the amount of data that is created is very high. For any application, looking up any particular piece of data might be a problem. The range of data that is created might not be an ideal precedent for data accessing without having the data aligned in a manner that allows globalized data access. This data abstraction layer makes accessing the data for the applications simpler and more efficient. The application layer is the layer that allows us to access the data at the data accumulation and abstraction layer. The applications could differ based on the type of service we intend to incur from the devices or just plainly from the data that is gathered. The application layer gives us the ability to interpret data and interact with the devices at the first level, thus letting us to decide what the device should do based on its functions. The last layer is the collaboration and processes layer. It lies outside, in many senses, of the machine realm that encompasses the IoT. This layer brings the people and their ability to make something out of raw data.

In [29], the IoT Network is divided into 4 layers. This approach helps us simplify the understanding of the network while allowing us to deeply understand the functions, features, and security aspects respective to each layer. This architecture is a cloud-based approach to the IoT. Not only is it easy but also the overall distinction helps us grab the vitality of IoT working, so we can have a better look at its weaknesses. Figure 13.2 shows the layered architecture of IoT.

4.1 Perception Layer

This layer consists of devices such as sensors and actuators that are responsible for data creation. The devices create the data that is necessary for the working of IoT. In an IoT network, things are used in order to gain some benefit or service from their

Fig. 13.2 4-layered IoT architecture



functioning. The devices could be anything such as environmental sensors, cameras, and smart meters. These devices are used to collect data and that data is sent to the layers above it. This layer encompasses all the devices that are used in an IoT network, to create and send data. The devices need not necessarily be sensors that create data, they can also be actuators that are supposed to carry out any specific operation such as smart locking system. In a smart locking system, the smart lock can be used to remotely lock or unlock any piece of property, building, etc. based on the use case. Thus, this layer not only consists of the devices that create data, it also consists of the devices that help in interaction with the environment or any other entity, to carry out an action.

4.2 Network Layer

The network layer, in essence, connects all the entities of an IoT architecture. This layer deals with the transmission and transportation of the data packets between the devices, base stations, cloud storage, etc. The network layer is responsible for the communication paradigm of the IoT. The network layers consist of devices, gateways, base stations, etc. that are used for the seamless communication in the IoT architecture. The data that is created at the perception layer and needs to be processed is transported to the upper layer by the network layer. This layer makes use of different technologies for data transmission such as Sigfox, BLE, MQTT, Wi-Fi, Zigbee, LTE-M, NB-IoT, etc. These technologies can provide different benefits based on the use case they are applied to.

4.3 Data Processing Layer

This layer is the one that is responsible for data storage and processing. The cloud aspect of IoT, which stores data in a manner that they are easily accessible to the user for different purposes, falls in this category. The data that is created at the perception layer and transported by the network layer wound up in the data processing layer where it is stored on the cloud. The other purpose that this layer serves is data processing. The data that is assembled is properly accumulated in order to be accessed by the applications in a manner that is simple and effective. The data that is once collected is then processed to be filtered, analyzed, and prepared to be fit for application use. The raw data is essentially changed into meaningful data that suits the purpose.

4.4 Application Layer

This is the final layer in IoT architecture. The application layer in a simpler sense provides humans a medium to interact with the IoT network. The application layer allows the user to mingle with the cloud data to get information out of it. It helps in getting the services that we need from a user perspective. People also use this layer to control the perception layer devices, allowing them to procure desired services from them.

5 Layer-Based Threats

In Table 13.2, major threats pertaining to various layers are mentioned.

Table 13.2 Major threats in various layers of IoT

Layer	Type of threats
Perception layer	Physical damage to hardware
	Eavesdropping [30, 31]
	Impersonation [31]
	Jamming [30, 32]
Network layer	Phishing [33]
	Sybil [34]
	Sinkhole [34, 35]
	Black hole [30]
	Wormhole
Data processing layer	Malicious insider attack
	DoS attack
	Cloud malware injection [33]
	Cloud Flooding Attack [33]
Application layer	Malicious code injection [30, 31]
	SQL injection [31–33]
	Sniffing Attacks [33]
	Data aggregation & distortion [30, 34]

5.1 Attacks on Perception Layer

Physical damage to the hardware device: This attack deals with tampering of the perception layer devices. The attacker tends to physically harm the sensor, actuator, camera, etc. that is responsible for the data creation. This way the network suffers loss even before the data is generated. This can be very costly for maintenance, since hardware components can be very expensive. This type of attack can be countered by installing the devices at places that are not obvious for the attacker to pay attention to or simply putting up the devices at spots that can be very tricky to reach.

Eavesdropping: This is a very prominent issue and is actually a threat at other layers too. In this type of attack, the attacker intercepts the data that is exchanged by obstructing the network traffic. This way the unauthorized party is able to gain information illegally for them. This is a very prominent problem in Computer Networks. The isolation of the affected node could help control the damage due to this attack [30].

Impersonation: In this type of attack, the attacker uses a fake identity to pass off as a legitimate user. This allows the attacker to gain access to unauthorized data. Such attacks can be countered by putting up firewalls or adopting appropriate cryptographic algorithms.

Jamming: This attack refers to leading the devices into the state where they are unable to provide services to the legitimate user. Jamming can be countered by putting the jammed region of the IoT network to sleep by the help of detection [36].

5.2 Attacks on Network Layer

Phishing attacks: Phishing is an attack that is used for stealing user data. In IoT, this attack can be used to compromise many devices in a single attack [33]. This type of attack can be countered with the help of detection algorithms based on machine learning.

Sybil attack: It is an attack in which the network is compromised due to a number of pseudo-users that infiltrate the network. The number of such users leads to degradation of services in the network [37]. One way to subvert such an attack is to verify the users that are part of the network to ensure their identity as a legitimate user.

Sinkhole attack: In this attack, the attacker tends to offer a false destination node to other neighbouring nodes. This allows the attacker to send the data from other nodes to the node of its desire which causes the data packets to be compromised. This attack can be countered with the help of cryptographic algorithms [38].

Blackhole attack: In this attack, the attacker attracts the data from other nodes by pretending to be the shortest path to the receiver. This allows the attacker to control the data traffic in the network. Different mechanisms to thwart such attacks are discussed in [39].

Wormhole attacks: It is a routing attack that disrupts the data flow of the network by channelling data packets to different destinations in the network. This type of attack can be countered by employing different techniques of monitoring or by checking the network for the delays of different paths to the receiver. Wormhole detection mechanisms are discussed in [40].

5.3 Attacks on Data Processing Layer

Malicious insider attack: This attack deals with the legitimate user of the network who uses his authenticity or authorization to look up the data of other users. This allows the attacker to gain access to illegal data. To keep such attacks from happening it should be noted that the authorization may only be provided to the ethical people.

DoS attack: This is a very common attack. It involves sending huge amounts of data into the network, overloading the components, which keeps the users from getting resources. The use of Artificial Intelligence (AI) in detecting such attacks is promising [41].

Cloud malware injection: This attack involves the injection of the cloud with malicious code or malware that gives them access to the cloud data and services. Employing Hypervisor could help defend against such attacks [42].

Cloud flooding attack: This attack is focused on the cloud and disrupts its function by sending huge loads of data packets thus ‘flooding’ the cloud. It keeps the cloud from engaging with requests from valid users. Using firewalls could help in such situations [42].

5.4 Attacks on Application Layer

Malicious code injection: In this type of attack, the application layer is injected with a piece of code that comes from untrustworthy elements. This makes the system vulnerable and easy for attackers to gain an opening to enter the system. Cross-site scripting (XSS) is a common example. Regularly checking codes, avoiding vulnerable constructs in code, scanning the applications, etc. are a few ways in general to steer clear of such attacks [43].

SQL injection: This involves injecting suspicious SQL commands in the system that may lead to access of data the attacker is not authorized to gain. To thwart such attacks, few measures such as decompiling the source code of any weak apps could be taken to limit SQL injection success rate [44].

Sniffing attacks: Such attacks are used to intercept the data transfer over the network with help of applications called sniffer and gain illegal information. Using different techniques and protocols to monitor and identify threats are suggested amongst other things in [45].

Data distortion and a: The data that is being exchanged can be collected and/or distorted by an eavesdropper. This would compromise the entire data and system since the attacker could gain access to all the relevant information. Different mechanisms could be employed to ensure such things are kept from happening.

6 Transmission Technology-Based Threats

This section describes the common threats in two types of networking technologies, namely, short-range wireless communication and low power wide area network.

6.1 Short-Range Wireless Communications (Bluetooth, Wi-Fi, and Zigbee)

6.1.1 Bluetooth

Bluetooth is a very common technology in today's date. It would be fair to assume that every smartphone today has this facility in it. Bluetooth is a short-range wireless technology that allows for the exchange of data. Smartphones, smart watches, wireless head/earphones, etc. are major devices in this century that utilize this technology for connectivity and data transmission.

Bluetooth operates at 2.4GHz unlicensed ISM (Industrial, Scientific, and Medical) spectrum. The range of Bluetooth varies between 10–100 m. The frequency range of Bluetooth does become a problem since there are many appliances that operate at similar frequencies such as microwave and hence causes interference [46].

6.1.2 Main Security Threats

It is to be mentioned that security vulnerabilities in Bluetooth vary with the version of it. But there are still older versions that are present in relatively older devices that are being used today.

Man-in-the-middle attack: This is one of the prominent attacks that harm the Bluetooth technology. In this attack, if the attacker is aware of the passkey that has been used for pairing previously, then he can impersonate either of the parties in future when they try to pair again [47].

Blue-tracking: In this attack, the attacker assumes a name that would seem harmless to the user Bluetooth and hence drives any suspicion away from the attacker, while monitoring the user's address to track them [48].

Backdoor: In this attack, to gain access to the user's device, the attacker exploits the pairing mechanism. This allows the attacker to use the device posing as a legitimate user.

Worms and viruses: The Bluetooth is very much vulnerable to viruses and worms. Major example of worms that affect the Bluetooth are Lasco, which affects the Bluetooth devices that are set in discoverable mode [49].

DoS attacks: There are various DoS attacks that can be found in Bluetooth, such as BD_ADDR duplication attack, SCO/eSCO attack and L2CAP Guaranteed Service attack. They all affect the service capability of the Bluetooth.

Victim device cloning attack: In this type of attack, the attacker uses the address of the user and clones it and uses it to pose as the original user. This can be a very dangerous attack since it would allow the attacker to pose as a valid user. Different types of victim cloning attacks are MAC address spoofing, brute force attack, forced re-pairing attack, etc.

6.1.3 Wi-Fi

Wi-Fi, or 802.11, is a wireless protocol that is intended for data transmission and communication. This is a very popular technology since many households these days have a Wi-Fi router that provides for their internet connectivity. This technology is also a prime player in IoT to bring together multiple devices providing them a means to interact with each other.

Wi-Fi since its inception has added up many frequency ranges at which it operates, such as: 900 MHz, 2.4 GHz, 3.6 GHz, 4.9 GHz, 5 GHz, 5.9 GHz, 6 GHz, and 60 GHz bands [50]. Wi-Fi employs several security mechanisms such as WEP (Wired Equivalent Privacy), WPS (Wi-Fi Protection Systems), IEEE 802.11i [51], and several others. The initial WEP provided authentication, data integrity, and encryption but was found to have major weaknesses after which, IEEE 802.11i was introduced, it had two versions hence known as Wireless Protected Access (WPA) and the second one WPA2. This was followed by the arrival of WPS which enhanced security furthermore.

6.1.4 Main Threats in Wi-Fi

The authors in [52], define various stacks prominent in Wi-Fi that can cause damage, few of them are mentioned as follows:

Identity Spoofing: In this attack, the attacker copies the MAC address of any Wi-Fi device in order to be able to impersonate the device to gain access as a valid device and can pose a huge threat to the Wi-Fi network. Such attacks can generally be thwarted with the help of intrusion detection algorithms.

Packet Replay: In this type of attack, the attacker uses WEP packets from a previous exchange and replays it to gain access as a legitimate user. This may allow the attacker to be able to get vital information.

Physical attack: Many devices such as routers have vital security data printed on them which may leave the network in danger of getting hacked. This calls for installing such pieces of equipment in safer environments.

ICV tampering: Integrity Check Value (ICV) is a piece of information that is generated for the purpose of integrity of the network. According to [53], it is very much possible for an attacker to make any sort of modification in a message and tamper ICV too in order to make that message seem legit.

Device desynchronization: This is a type of attack that allows the attacker to meddle with the synchronization routine of the Wi-Fi Stations in a manner that causes the loss of power by disrupting the Power Saving mechanism.

Packet Trashing: This attack involves the attacker in a manner that they match the transmission time of a valid user in a network to send data packets at the same time, this causes collision of packets which is an utter waste of medium and resources.

Channel Jamming: This attack causes similar complications to packet trashing, in the sense that it wastes the resources of the channel. In this attack the attacker tends to send noise into the network in order to jam the communications.

Cryptanalysis Attacks: There are several attacks of this kind that involve cracking keys and passwords to break encryptions or other mechanisms. Few such attacks are: WPA cracking, online WPS cracking, and key reinstallation attack.

6.1.5 ZigBee

Zigbee is a short-range, low power communication technology that has garnered much attention over the years as an effective IoT transmission technology. It is majorly deployed in mesh topology, a network that is very well-suited for Home Automation. Its range is typically >100 m [54].

Zigbee is based on the IEEE 802.15.4 radio frequency standard and is deployed at an unlicensed 2.4GHz frequency. As mentioned, it finds its applications majorly in home automation and similar short-range networks that require low power consumption and low data rate.

6.1.6 Main Threats for Zigbee

Sink attack: This attack offers other nodes, the shortest path to their destination, thus gathering the network traffic towards it, which is not healthy for the network [55].

Neighbour attack: This attack is very good for discovering the neighbouring nodes that are present in a network. The infected node sends HELLO commands to other nodes which are not aware of the deceit and respond to it, which in turn causes waste of energy [56].

Physical layer attacks: Zigbee is also vulnerable to attacks such as collision attacks or attacks that focus on jamming the network [57].

Rogue acknowledgment: In this attack, the attacker spoofs any device and sends fake acknowledgment pretending to be a real valid device. This allows them to create establishments on the wrong grounds, yet successfully.

Same-Nonce attack: This type of attack occurs when a device sends two consecutive messages with the same NONCE and there is an eavesdropper listening in, then the eavesdropper would be able to decipher the message with XOR function, thus leaking the data [58].

Packet injection: This type of attack allows the attacker to inject any malicious content into a regular packet that's part of the network.

6.2 Low Power Wide Area Networks (LoRaWAN, Sigfox, and NB-IoT)

6.2.1 LoRaWAN

The LoRaWAN or long-range wide area network is a low power wide area network that has proven to be effective in communicating over great distances with limited power consumption. LoRa is a radio technology that is very intuitive to the IoT application. It is deployed in unlicensed spectrum and is a MAC layer protocol. The LoRa has a channel bandwidth of 125–500 kHz and employs CSS modulation. It has found its application in many fields such as Glasgow, where the IoT network is held together with the help of LoRa for communication [57], Smart parking [58], Smart irrigation [59], and many more.

There are a few vulnerabilities and threats that surround LoRa:

Replay attack for ABP-activated nodes: This attack allows the attacker to use the messages from last session with larger counter values and use them in the current session. This type of attack is possible for ABP-activated devices [60].

Eavesdropping: The LoRa also suffers from eavesdropping. This is possible by the method of Key stream reuse. It also employs crib dragging to fully decrypt the text.

Bit-flipping attack: This attack takes place between the server of the network provider and the application server. Since the security in this region is not so strong it is very easy to steal packets or reroute them [60].

ACK spoofing: Due to the implementation of spread spectrum technology, the packet transmission takes a longer time in LoRa which gives the attacker opportunity to monetize the situation and block the acknowledgment signal from reaching a device, and use the ACK to spoof [60].

6.2.2 Sigfox

Sigfox is a narrowband technology that is a part of the LPWAN family. Sigfox is deployed in unlicensed spectrum and makes use of BPSK modulation. It has a bandwidth of 100 Hz.

It is used for long-range communications and is well-suited for IoT applications that require low data rates. It is being employed all over the world for different IoT purposes and more.

6.2.3 Security Threats for Sigfox

Wide-band and Selective jamming: This is possible in Sigfox [61].

Eavesdropping: Given its unique structure it is hard but an attacker who is aware of sigfox technology can easily commit eavesdropping.

Traffic analysis: This is a very prominent threat in Sigfox when Phase-shift keying (PSK) is compromised.

Replay attacks: This might be the biggest vulnerability of Sigfox. The Sequence number that is used in Sigfox is checked only at the backhaul network, leaving room for replay attacks [61].

DoS attack: Due to the limit on sequence number, the system can get DoSed all by itself [62].

6.2.4 NB-IoT

This is another type of LPWAN technology that was developed by the Third Generation Partnership Project (3GPP). It is a radio technology that is heavily inspired from Long Term Evolution (LTE). It works at a bandwidth of 180 kHz and is a cellular technology. NB-IoT has a wide range and is known for its higher data rates, lower latency, and better QoS amongst other LPWANs. It has numerous applications such as smart parking [63], tracking [64], and Smart bikes [65].

6.2.5 Security Vulnerabilities

Port scanning: The attacker uses this attack to conduct a scan in order to find open ports and services that are provided in the network [62].

ARP spoofing: In this type of attack, the attacker sends fake Address Resource Protocol (ARP) packets into the network in order to gain a connection with the devices of the network [66].

DNS spoofing: In this attack, the attacker puts up false and fraudulent Domain Name System (DNS) records into the network which compels the server to revert corrupted DNS information.

Hacking into network using UE: A malicious Federal Intelligent Transportation Systems (UE) could be used to hack into the network by sending in malicious data into a port that was created by other UE to gain access to the Packet Data Gateway, which gives further access to the network services.

DoS attack: This attack can come as a consequence of the abovementioned hacking, by engaging nodes in a repeated packet transfer.

7 Smart Applications

Today, more than half of the globe's population lives in urban areas, and this will rise significantly over time. Therefore, cities are facing issues brought by this population growth, which are air pollution, scarcity of resources, and unmanaged traffic. So smart city applications are introduced for a smoother flow of resources in cities that will allow for real-time response to people in society. In order to do that, a smart city needs to prepare itself to respond to any kind of threat and attack while transitionally maintaining relationships with every person in the nation. Some of the smart city applications are discussed in this section:

1. Smart energy: Smart city's main objective is to efficiently control the resources and consumption of energy to promote the utilization of renewable energy for strategic management of available resources and for better planning. Smart cities are in charge of creating eco-friendly and liveable environments for every individual, for pioneering development so that citizens can live comfortably and adapt to novel changes in their surroundings. Smart energy depends on resilience, sustainable energy solutions, agile end users, and renewable resources. In recent times for consumption and energy generation, sensor nodes are established to check for any failure that occurs in supply of electricity it can leverage and check [67].
2. Smart infrastructure: The main motive is to make sure how resources can be utilized to their full extent and in what way performance can be improved so that it can be blended well with different intelligent infrastructures like smart people, smart energy, smart economy, and smart environment. As a considered fact while setting up smart infrastructure, environment-friendly infrastructure should be set up which guarantees privacy and security of every individual and that their data is safe from any unauthorized access [68].
3. Smart governance: For productively managing cities, smart governance is needed. In the present scenario, information and communication technology

plays a vital role in every individual's life. Without its presence, urbanization is not possible. Mobile application, website, and online accessibility are helping the public to directly access portals while permitting them to give any suggestions and feedback to the government. To enable this, some models have been proposed in smart governance: Government to Citizen Model(G2C), Government to Business Model(G2B) as well as sustainable development are all mandatory for smart governance. However, for the creation of digital platforms funds are needed and as a known fact developing nations face difficulty with respect to funds, they are at a crunch. Lack of interest of individuals as they do not want to participate as not everyone is literate. Therefore, these are some challenges faced by the government in setting up Smart governance [69].

4. Smart healthcare: Healthcare [70] is a significant sector in every person's life. The main elements are patients, medical practitioners, pharmacists, etc. along with monitoring their health, detection of disease, treating them with the help of advanced technological developments [71], and algorithms for remote patient monitoring and smart techniques. Researchers collected ample amount of data that can be used to investigate the behaviour of individuals and how much they are aware of healthcare.
5. Smart traffic: In smart cities, major issues are managing traffic fuel saving, accident detection, traffic violators [72], etc. Thus, Intelligent Traffic Management Systems (ITMS) [73] is the need of the hour. Smart traffic makes use of data gathered through sensing devices and vehicles' licence plates, along with the help of machine learning approaches to know the best route for traffic mobilization, to collect information on road accidents and to address time wastage and fuel scarcity issues. Also, green corridors are built for emergency services to avoid traffic blocking.

Some of the applications of smart cities with threats and solutions are reviewed in Table 13.3 [74–80]:

7.1 Threats to Smart City

With the emergence of Internet of Things (IoT), so many connected devices are positioned in cities all around the globe, but with this technological advancement IoT possesses some threats as cyber criminals, malicious attackers, and intruders may take advantage of it which results in harming the security and privacy of citizens when it is neglected [79]. Some of the common threats are:

1. Man-in-the-middle: It is an eavesdropping attack in which a conversation is interrupted by an attacker who positions themself in the middle of the conversation and acts like an authorized entrant. This data is hacked by the attacker which is then transferred by any party, and any malicious activity by an intruder is detected behind time [81].

Table 13.3 Smart Applications–based Threats

Serial no.	Application	Threats	Solutions
1	Smart energy	• Distributed denial of service (DDoS) • Spoofing • Unauthorized aAccess • Denial of Service (DoS) • Phishing • Security Breaches	• Cybercrime intelligence. • Encroachment detection and prevention mechanism implementation. • Analysis of risk. • Capitalisation on resilient resource system. • Regulatory policies and planning. • Rules and regulations guiding data. • Data policy confidentiality.
2	Smart infrastructure	• Man-in-the-middle • Denial of Service (DoS) • Security breaches • Untrusted service providers • Malware infection • Failure of system	• Interoperable security • Secure middleware • Implementation of application security control methods • Two factor authentication process • Risk assessment
3	Smart healthcare	• Eavesdropping • Jamming attack • Denial of service (DoS) • Fingerprint and Timing based Snooping (FATS) • Sensor attack • Select Forwarding Attack (SF)	• Intrusion detection and prevention mechanism • Routing monitoring • Detection of node failure at earliest to done replacement timely • Secure suthorization • Data confidentiality
4	Smart traffic	• Malicious actors • Security Gaps in Controller Area Network (CAN bus) • Technical and environmental challenges • Network attack • Data breaches • Malware attack • Distributed Denial of Service (DDoS) • Man-in-the-middle • Jamming • Eavesdropping • Sybil attack • Spoofing	• Policy enforcement technique • Secure middleware • Authorized authentication • Encipherment • Control mechanism monitoring • Proper monitoring of routes • Enabling sensing techniques • Implementation of data encryption techniques like and AES • Inappropriate behaviour detection mechanism • Smart traffic controllers implementation • Utilizing non-repudiation
5	Smart governance	• Digital security breach • Mirai botnet attack • Malicious actor • Distributed Denial of Service (DDoS) • Networking attack • Failure of system • Data leakage • Malware infection	• Authentication and confidently • Resilience • Lightweight intrusion detection prediction at earliest • Following privacy preserving data mining techniques • Homomorphic encryption • Two-level authentication • Regulatory policies and planning • Interoperable security

2. Data and identity theft: It is a type of phishing attack where data are stolen by cyber criminals and they ingress details by unauthorized data similar to access details about bank, digital life, and credit card information. Susceptible information is collected by them side by side from account access, information about social circles and family members. Identity theft is done by them to gain wealth and to cause damage which is not rectifiable, such as harming the reputation of individuals. Cyber criminals can easily gather data as online portals always require an email address that can be easily hacked by them; through it they enter another account detail as they are more vulnerable to data and identity theft [82].
3. Device hijacking: In this, intruders hijack systems and devices by attacking network security of network. Device Hijacking is also known as cyber hijacking and computer hijacking. Several kinds of hijacking attacks are browser hijacking, session hijacking, page hijacking, domain name system hijacking, and internet protocol hijacking [83]. This intruder takes control over the session which is between client and server and man-in-the-middle enters between session by kicking off the client out of session which disturbs all the networking and by the time this issue is addressed it is too late [84].
4. Distributed Denial of Service (DDoS): In this normal traffic is distributed by a malicious actor of a server network by forwarding suspicious amounts of traffic over the web. Botnet lays its foundation for establishment, intruder gets access to send instructions to each bot and it requests to gain access over IP address but this normal traffic is flooded with internet traffic which results in denial of Service. Distributed Denial of Service attack can be known if website connection slows down or page is not responding. Also with the help of traffic analytics tools, DDoS attacks can be spotted. Other forms of Distributed Denial of Service are SYN flood, HTTP flood, and DNS Amplification. If network experts are able to differentiate between normal traffic and traffic forwarded by an intruder, the Distributed Denial of Service attack can be stopped in time [85].
5. Permanent Denial of Service (PDoS): It is also referred to as phlashing, in which Permanent Denial of Service occurs, through disruption in hardware as more data are getting stored or moving to cloud computing in present time, this attack disturbs and damages the ongoing work. In PDoS intruder attacks, system and devices are destroyed by the intruder by entering corrupt firmware pictures, so the person who suffers is left with no other choice than to get it repaired or to buy a new system. It causes loss which is irreplaceable as attack is irreversible and as organizations are always ready to pay a hefty amount to prevent it. Attacker's purpose is to wreak havoc. Permanent Denial of Service attack saves malicious actor's time when compared to Distributed Denial of Service attack where in DDoS attack exists till the time permitted by intruder which is not the case in PDoS. In the Internet of Things (IoT) connected devices with greater financial gains are more vulnerable to Permanent Denial of Service attack. While keeping the system upgraded Permanent Denial of Service attack risk can be minimized and by keeping security check all round the day too [86].

8 Conclusion

Internet of things (IoT) devices are capable of connecting with almost all real-world things for information sharing. IoT uses sensors which collect data over the web, which provides knowledge that helps in improving the lives of citizens by creating novel applications. IoT architecture is required for the successful implementation of devices and for enhancing the quality of life but privacy, security, and threat issues are major challenges that are faced by IoT environment in present time. Moreover, safety should be considered as it plays a major role in safeguarding systems as well as providing a reliable method for protection of IoT components from any kind of threat and attack. As IoT devices work on rules and policies, proper ethics and protocols should be guided. In this chapter, review of privacy, security, and threats were presented in addition to layers of IoT, attack types, and major security threats which were also discussed. Respectively, it highlights smart city applications with their solutions. Besides, previously done work is reviewed and examined which focuses on security requirements. Lastly, many issues and threats are identified that open the way for eminent research in future.

Acknowledgement This work is partially funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the Project UIDB/50008/2020; and by Brazilian National Council for Scientific and Technological Development - CNPq, via Grant No. 313036/2020-9.

References

1. Jang-Jaccard, J., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences, 80*(5), 973–993.
2. https://www.alliedmarketresearch.com/internet-of-things-iot-in-agriculture-market. DOA: 06-06-2021.
3. https://www.statista.com/statistics/992118/worldwide-internet-of-things-top-patent-owners/. DOA: 06-06-2021.
4. https://www.forbes.com/sites/zakdoffman/2019/09/14/dangerous-cyberattacks-on-iot-devices-up-300-in-2019-now-rampant-report-claims/?sh=49add6c58926. DOA: 06-06-2021.
5. Alqassem (2014, May–June). Privacy and security requirements framework for the internet of things (IoT). *International Conference on Software Engineering (ICSE) Companion India* (pp. 739–741).
6. Rahman, F. A., Daud, M., Mohamad, M. Z. (2016, March). Securing Sensor to Cloud Ecosystem using Internet of Things (IoT) Security Framework. *ICC (International Conference on Internet of things and Cloud Computing)*, United Kingdom, Article No.: 79.
7. Lee, Y. J., & Kim, D. H. (2015). Threats analysis, requirements and considerations for secure internet of things. *International Journal of Smart Home, 9*, 191–198.
8. Abomhara, M., Køien, G. M. (2014, May). Security and privacy in the internet of things: current status and open issues. *Privacy and Security in Mobile Systems (PRISMS) Denmark*, (pp. 1–8).

9. Alqassem, I., Svetinovic, D. (2014, December). A taxonomy of security and privacy requirements for the internet of things (IoT). *Industrial Engineering and Engineering Management (IEEM) Malaysia* (pp. 1244–1248).
10. Kim, H.-J., Chang, H.-S., Suh, J.-J., Shon, T.-S. (2016, May). A study on device security in IoT convergence. *Industrial Engineering, Management Science and Application (ICIMSA) South Korea* (pp. 1–4).
11. S. Li, T. Tryfonas, H. Li, "The internet of things: A security point of view", Internet Research, vol. 26, pp. 337-359, April 2016.
12. Atlam, H.F., Alenezi, A., Alassafi, M.O., Walters, R.J., Wills, G.B. (2018). XACMLforbuildingaccess control policies in internet of things. In: *Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security (IoTBDS 2018)* (pp. 253–260).
13. Atlam, H.F., Walters, R.J., Wills, G.B. (2018). Internet of nano things: security issues and applications. In *2018 2nd International Conference on Cloud and Big Data Computing, no. October* (pp. 71–77).
14. Atlam, H. F., Walters, R. J., & Wills, G. B. (2018). Fog computing and the internet of things: A review. *Big Data Cognitive Comput., 2*(2), 1–18.
15. Deogirikar, J. (2017). Security attacks in IoT : a Survey. In *International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)* (pp. 32–37).
16. Abdur, M., Habib, S., Ali, M., & Ullah, S. Security issues in the internet of things (IoT): A comprehensive study. *International Journal of Advanced Computer Science and Applications, 8*(6).
17. Suo, H., Wan, J., Zou, C., & Liu, J. (2012). Security in the internet of things: A review. In *International conference on computer science and electronics engineering (CCSEE 2012)* (Vol. 3, pp. 648–651).
18. Maple, C. (2017). Security and privacy in the internet of things. *Journal of Cyber Policy, 2*(2), 155–184.
19. Martin, P., & Brohman, K. (2014). CLOUDQUAL: A quality model for cloud services. *IEEE Transactions on Industrial Informatics, 10*(2), 1527–1536.
20. Cerf, V., Ryan, P., Senges, M., & Whitt, R. (2016). IoT safety and security as shared responsibility. *Business Information, 1*, 7–19.
21. Egele, M., Kruegel, C., Kirda, E., & Vigna, G. (2011). PiOS: Detecting privacy leaks in iOS applications. In *Proceedings of 28th annual network and distributed system security symposium* (pp. 1–15).
22. Enck, W., Gilbert, P., Han, S., Tendulkar, V., Chun, B.-G., Cox, L. P., Jung, J., McDaniel, P., & Sheth, A. N. (2014). TaintDroid: An information flow tracking system for realtime privacy monitoring on smartphones. *ACM Transactions on Computer Systems, 32*(2), 1–29.
23. Xiao, X., Tillmann, N., Fahndrich, M., de Halleux, J., Moskal, M., & Xie, T. (2015). User-aware privacy control via extended static-informationf low analysis. *Automated Software Engineering, 22*(3), 333–366. https://doi.org/10.1007/s10515014-0166-y
24. Enck, W., Ongtang, M., & McDaniel, P. (2008). *Mitigating android software misuse before it happens*. Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, Tech. Rep.
25. Ndibanje, B., Lee, H.-J., & Lee, S.-G. (2014). Security analysis and improvements of authentication and access control in the internet of things. *Sensors (Basel, Switzerland), 14*(8), 14786–14805. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4179010/
26. Mahalle, P. N., Anggorojati, B., Prasad, N. R., & Prasad, R. (2012). Identity driven capability based access control (ICAC) scheme for the internet of things. In *Proceedings of international conference on advanced networks and telecommunciations systems (ANTS)* (pp. 49–54).
27. Hernandez-Ramos, J. L., Pawlowski, M. P., Jara, A. J., Skarmeta, A. F., & Ladid, L. (2015). Toward a lightweight authentication and authorization framework for smart objects. *IEEE Journal on Selected Areas in Comm., 33*(4), 690–702.
28. http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf. DOA: 06-06-2021.

29. https://www.hiotron.com/iot-architecture-layers/. DOA: 06-06-2021.
30. Jha, R. K., Puja, H. K., Kumar, M., & Jain, S. (2021). Layer based security in narrow band internet of things (NB-IoT). *Computer Networks, 185*, 107592. ISSN 1389-1286.
31. Dorsemaine, B., Gaulier, J., Wary, J., Kheir, N., & Urien, P. (2016). A new approach to investigate IoT threats based on a four layer model. In *2016 13th international conference on new technologies for distributed systems (NOTERE)* (pp. 1–6). https://doi.org/10.1109/NOTERE.2016.7745830
32. Wheelus, C., & Zhu, X. (2020). IoT network security: Threats, risks, and a data-driven defense framework. *IoT, 1*, 259–285. https://doi.org/10.3390/iot1020016
33. Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., & Sikdar, B. (2019). A survey on IoT security: Application areas, security threats, and solution architectures. *IEEE Access, 7*, 82721–82743. https://doi.org/10.1109/ACCESS.2019.2924045
34. Nawir, M., Amir, A., Yaakob, N., & Lynn, O. B. (2016). Internet of things (IoT): Taxonomy of security attacks. In *2016 3rd international conference on electronic design (ICED)* (pp. 321–326). https://doi.org/10.1109/ICED.2016.7804660
35. Atlam, H. F., & Wills, G. B. (2020). IoT security, privacy, safety and ethics. In M. Farsi, A. Daneshkhah, A. Hosseinian-Far, & H. Jahankhani (Eds.), *Digital twin technologies and smart cities. Internet of things (technology, communications and computing)*. Springer. https://doi.org/10.1007/978-3-030-18732-3_8
36. Namvar, N., Saad, W., Bahadori, N., & Kelley, B. (2016). Jamming in the internet of things: A game- theoretic perspective. In *IEEE global communications conference (GLOBECOM)* (pp. 1–6).
37. K. Zhang, X. Liang, R. Lu and X. Shen, "Sybil attacks and their defenses in the internet of things," IEEE Internet of Things Journal, vol. 1, no. 5, pp. 372-383, Oct. 2014.
38. Salehi, S., Razzaque, M. A., Naraei, P., & Farrokhtala, A. (2013). Detection of sinkhole attack in wireless sensor networks. In *IEEE international conference on space science and communication (IconSpace)* (pp. 361–365).
39. Ali, S., Khan, M. A., Ahmad, J., Malik, A. W., & ur Rehman, A. (2018). Detection and prevention of black hole attacks in IOT & WSN. In *2018 third international conference on fog and mobile edge computing (FMEC)* (pp. 217–226). https://doi.org/10.1109/FMEC.2018.8364068
40. Goyal, M., & Dutta, M. (2018). Intrusion detection of wormhole attack in IoT: A review. *International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET), 2018*, 1–5. https://doi.org/10.1109/ICCSDET.2018.8821160
41. https://portswigger.net/daily-swig/artificial-intelligence-can-stop-iot-based-ddos-attacks-in-their-tracks-research. DOA: 06-06-2021.
42. Chouhan, P., & Singh, R. (2016). Security attacks on cloud computing with possible solutions. *International Journal of Advanced Research in Computer Science and Software Engineering, 6*(1), 92–96.
43. https://www.netsparker.com/blog/web-security/code-injection. DOA: 06-06-2021.
44. https://www.smartdatacollective.com/assessing-severity-sql-injection-threats-iot-security/. DOA: 06-06-2021.
45. Swamy, S. N., Jadhav, D., & Kulkarni, N. (2017). Security threats in the application layer in IOT applications. In *2017 international conference on I-SMAC (IoT in social, Mobile, analytics and cloud) (I-SMAC)* (pp. 477–480). https://doi.org/10.1109/I-SMAC.2017.8058395
46. Fowler, P. (2002). 5 GHz goes the distance for home networking. *IEEE Microwave Magazine*, 49–55.
47. Hypponen, K., & Haataja, K. M. J. (2007). Nino: Man-in-the-middle attack on Bluetooth secure simple pairing. In *The third IEEE/IFIP international conference in Central Asia on internet* (pp. 1–5).
48. Jakobsson, M., & Wetzel, S. (2001). Security weaknesses in bluetooth. In *Topics in cryptology: The Cryptographer's track at RSA conference* (pp. 176–191).
49. https://www.f-secure.com/v-descs/bluetooth-worm_symbos_lasco_a.shtml. DOA: 9-6-2021.
50. https://en.wikipedia.org/wiki/List_of_WLAN_channels. DOA: 9-6-2021.

51. IEEE, "IEEE Std 802.11i," Amendment 6: Medium Access Control Security Enhancement, 2004.
52. Lounis, K., & Zulkernine, M. (2020). Attacks and defenses in short-range wireless technologies for IoT. *IEEE Access, 8*, 88892–88932. https://doi.org/10.1109/ACCESS.2020.2993553
53. N. Borisov, I. Goldberg, and D.Wagner, "Intercepting mobile communications: The insecurity of 802.11," in Proceedings of the 7th annual international conference on mobile computing and networking, pp. 180– 189, ACM, 2001.
54. https://behrtech.com/blog/6-leading-types-of-iot-wireless-tech-and-their-best-use-cases/. DOA: 9-6-2021.
55. Khanji, S., Iqbal, F., & Hung, P. (2019). ZigBee security vulnerabilities: Exploration and evaluating. In *2019 10th international conference on information and communication systems (ICICS)* (pp. 52–57). https://doi.org/10.1109/IACS.2019.8809115
56. Ďurech, J., & Franeková, M. (2014). Security attacks to ZigBee technology and their practical realization. In *2014 IEEE 12th international symposium on applied machine intelligence and informatics (SAMI)* (pp. 345–349). https://doi.org/10.1109/SAMI.2014.6822436
57. https://www.smartcitiesworld.net/news/news/iot-boost-for-glasgow-2265/ Date of access 9 June 2021.
58. https://www.iotevolutionworld.com/smart-transport/articles/432166-practical-parking-pni-sensor-corporation-senet-roll-out.htm/ Date of access 9 June 2021.
59. https://sfvbj.com/news/2018/feb/13/semtech-selected-smart-agriculture-system/Date of access 9 June 2021.
60. Yang, X., Karampatzakis, E., Doerr, C., & Kuipers, F. (2018). Security vulnerabilities in LoRaWAN. In *2018 IEEE/ACM third international conference on internet-of-things design and implementation (IoTDI)* (pp. 129–140). https://doi.org/10.1109/IoTDI.2018.00022
61. Kail, E., Banati, A., Lászlo, E., & Kozlovszky, M. (2018). Security survey of dedicated IoT networks in the unlicensed ISM bands. In *2018 IEEE 12th international symposium on applied computational intelligence and informatics (SACI)* (pp. 000449–000454). https://doi.org/10.1109/SACI.2018.8440945.5
62. Coman, F. L., Malarski, K. M., Petersen, M. N., & Ruepp, S. (2019). Security issues in internet of things: Vulnerability analysis of LoRaWAN, Sigfox and NB-IoT. *Global IoT Summit (GIoTS), 2019*, 1–6. https://doi.org/10.1109/GIOTS.2019.8766430
63. https://smartparkingsystems.com/en/nb-iot-sensors-applied-to-parking-system/Date of access 9 June 2021.
64. https://www.libelium.com/iot-solutions/smart-tracking/Date of access 9 June 2021.
65. https://e.huawei.com/topic/leading-new-ict-ua/nb-iot-ofo-smart-bike.html/ Date of access 9 June 2021.
66. https://www.veracode.com/security/arp-spoofing/ Date of access 9 June 2021.
67. Security and Privacy of Smart Cities: A Survey, Research Issues and Challenges Mehdi Sookhak, Helen Tang, Senior Member, IEEE, Ying He, Student Member, IEEE, and F. Richard Yu, Fellow, IEEE, Citation information: DOI 10.1109/COMST.2018.2867288, IEEE Communication Survey and Tutorial.
68. Ota, K., Kumrai, T., Dong, M., Kishigami, J., & Guo, M. (2017). Smart Infrastructure Design for Smart Cities. IT Professional, 19(5), 42–49. https://doi.org/10.1109/mitp.2017.3680957
69. https://www.google.com/amp/s/smartcity.press/smart-governance-for-smart-cities/amp/ Date of access :1 June 2021.
70. https://www.frontiersin.org/research-topics/21020/ai-powered-smart-healthcare-in-smart-cities/ Date of access : 1 June 2021.
71. Agarwal, P., Hassan, S. I., Mustafa, S. K., & Ahmad, J. (2020). An effective diagnostic model for personalized healthcare using deep learning techniques. In *Applications of deep learning and big IoT on personalized healthcare services* (pp. 70–88). IGI Global.
72. Agarwal, P., Chopra, K., Kashif, M., & Kumari, V. (2018). Implementing ALPR for detection of traffic violations: A step towards sustainability, 2018. In *Proceedia: Computer science* (pp. 738–743). Elsevier Journal Publication. ISSN: 1877-0509.

73. Agarwal, P., & Alam, A. (2018). Use of ICT in sustainable transportation. *Proceedings of International Conference on Future Environment and Energy, 150*(1), 1–7.
74. Smart Cities. (2019). Opportunities, challenges, and security threats January 2019. *Journal of Strategic Innovation and Sustainability, 14*(3).
75. https://www.google.com/amp/s/www.computerweekly.com/opinion/Smart-cities-face-challenges-and-opportunities%3famp=1 Date of access 8 June, 2021.
76. https://www.information-age.com/iot-governance-compliance-security-challenges-123490573/ Date of access 8 June2021.
77. IoT Security, Privacy, Safety and Ethics Hany F. Atlam and Gary B. Wills ©Springer Nature Switzerland AG 2020 M. Farsi et al. (eds.), Digital Twin Technologies and Smart Cities, Internet of Things, https://doi.org/10.1007/978-3-030-18732-3_8.
78. Zhang, K., Ni, J., Yang, K., Liang, X., Ren, J., & Shen, X. S. (2017). Security and privacy in smart city applications: Challenges and solutions. *IEEE Communications Magazine, 55*(1), 122–129. https://doi.org/10.1109/MCOM.2017.1600267CM
79. https://www.rambus.com/iot/smart-cities/Date of access 1 June 2021.
80. Oh, S., & Kim, Y. (2017). Security requirements analysis for the IoT. *International Conference on Platform Technology and Service (PlatCon), 2017*, 1–6. https://doi.org/10.1109/PlatCon.2017.7883727
81. https://www.google.com/amp/s/www.veracode.com/security/man-middle-attack%3famp/ Date of access 5 June 2021.
82. https://www.trendmicro.com/vinfo/it/security/news/online-privacy/identity-theft-and-the-value-ofourpersonaldata#:~:text=Identity%20theft%20happens%20when%20your,media%2C%20and%20credit%20card%20details/ Date of access 5 June 2021.
83. https://searchsecurity.techtarget.com/definition/hijacking/ Date of access 5 June 2021.
84. https://www.sciencedirect.com/topics/computer-science/hijacking/ Date of access 5 June 2021.
85. https://www.cloudflare.com/en-in/learning/ddos/what-is-a-ddos-attack/#:~:text=A%20distributed%20denialofservice%20(DDoS)%20attack%20is,a%20flood%20of%20Internet%20traffic/ Date of access 5 June 2021.
86. https://www.datafoundry.com/blog/what-is-a-permanent-dos-pdos-attack#:~:text=A%20Permanent%20Denial%20of%20Service,have%20moved%20to%20cloud%20computing/ Date of access 5 June 2021.

Chapter 14
Efficacy of Bio-absorbent Concept in Textile Effluent Treatment Technology Using Low-Cost Materials by Implementing Banana Bark and Orange Peel

A. Arivoli and X. Agnello J. Naveen

Abstract Textile Effluents include dyes that have color variation, though this is considered as a physical parameter of water. Due to their hazardous nature, dyes are considered to have devastating chemical effects on the bio-ecological system. The elimination of dye-color from the industrial wastewater assimilation process gives a classification in treatment technology, mainly if the adsorbent is cost-effective and naturally available. Banana bark and orange peel were used as adsorbent materials in this pilot research. The outcomes of the research shows a variation in potential of Hydrogen, contact of materials in time and specific concentration, assimilation of the adsorbent materials indicated the removal of pollutants present in the textile effluent. This investigation revealed that there is a significant difference between the use of orange peels which were more essential than when compared to the banana bark for the assimilation of pollutants from the textile wastewater. In this paper, attempts have been made to remove color, pH, TS, TDS, TSS, COD, and BOD by using two adsorbents. The highest color removal competence has been noticed as 65% and 75% for banana bark and orange peel, respectively. It result concludes a variation in the parameter like pH has decreased from 8.02 to 8.4. The TDS has observed the maximum percent removal of 50 and 85%. The investigational outcome shows the efficacy of the materials that have superior capability to eliminate color from wastewater and act as cost-effective adsorbent materials.

Keywords Bio-adsorbent material · Textile industry effluent · Banana bark · Orange peels · Efficacy of color change

---

A. Arivoli (✉)
Department of Environmental Science, Government Arts College, Ariyalur, Tamil Nadu, India

X. A. J. Naveen
Department of Environmental Science, Thanthai Roever Institute of Agriculture and Rural Development (TRIARD), Perambalur, Tamil Nadu, India

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9_14

1 Introduction

The environmental issues for textile effluent treatment were enforced by the Pollution committee board in 1989, mainly on releases of wastewater by nonpoint sources of the environment. Textile firms use kiloliters of water for operation and production; the waste has been classified into organic and inorganic chemicals by classifying goods manufactured from the outsourcing of vast quantities of wastewater discharged into the environment [33]. The water is used in synthetic dye preparation The amount, quality and assessment of wastewater changes from firm to firm by its production capacity, directly affecting the water input and output usage. The primary principal of textile wastewater evaluation is to evaluate physical parameters such as pH, color, temperature, electrical conductivity, and total dissolved solids and assess chemical parameters such as organic and inorganic chemicals, chemical oxygen demand, chloride, sulfate, salinity, and heavy metals [20]. The effluent produced from the classification of the manufacturing process in a textile firm is desizing, scouring, bleaching, mercerizing, dyeing, printing, and packing discharging of inorganic material [34]. Many synthetic dyes used are toxic/hazardous; have genotoxic, mutagenic, and carcinogenic effects; and cause biomagnification in aquatic life and human health [5, 9]. Wastewater from textile industries contain contaminants/toxic elements above the permissible levels or standards given by the pollution committee for discharge into point sources. Correct effluent treatment should be done by mechanical, chemical, phytoremediation, or bioremediation treatment process before being discharged into the ecosystem, to match a permissible limit (standards) of wastewater norms. Several technologies have to reduce water consumption by recycling the wastewater that comes from the textile industries. Bio-adsorption is one of the cost-effective techniques [17]. Different adsorbents are functional for removing dyes from aqueous solutions, such as alumina, crushed bricks, peat, sand, charcoal bentonite silica, and apricot [19, 21, 32]. Several nonconventional low-cost adsorbents used for dye removal (natural ingredients) contain fruit waste of *Prosopis juliflora*, wood, orange peel, banana pith, maize cobs, barley husk, bagasse pith, etc. The utilization of agricultural waste as a cost-cutting adsorbent has enormous significance in India, with more than 200,000 tons of dyes deposited into agriculture irrigation by kiloliters per annum [35]. This study is to advocate the efficacy, feasibility, viability, assimilation and concentration of using orange peel and banana bark as cost-cutting adsorbents.

Natural adsorbent is used to estimate the assessment of different physicochemical parameters such as color adsorbent capacity and concentration of material with varying intervals of time (dosage initial and final). This study is conducted on a laboratory-scale, and if the research has good potential for the removal of color from effluent, it can be used on a kiloliters per day.

2 Materials and Methods

2.1 Sample Parameter Collection

The effluent is collected from the end-operation processing unit of the dye industry in Tirupur for physicochemical analysis in a storage bottle. Figure 14.1 describes methods used for the preparation of absorbent methods used for the treatment of textile effluent.

2.2 Methods of Analysis (Table 14.1)

2.3 Principal of Bio-adsorption Experiment

The experiment principal involves mixing chemical reagents in five beakers. A volume of 150 ml textile effluent in operational process with two banana bark and orange peel adsorbent dosages (100, 200, 300, 400, and 500 mg/l), with varying

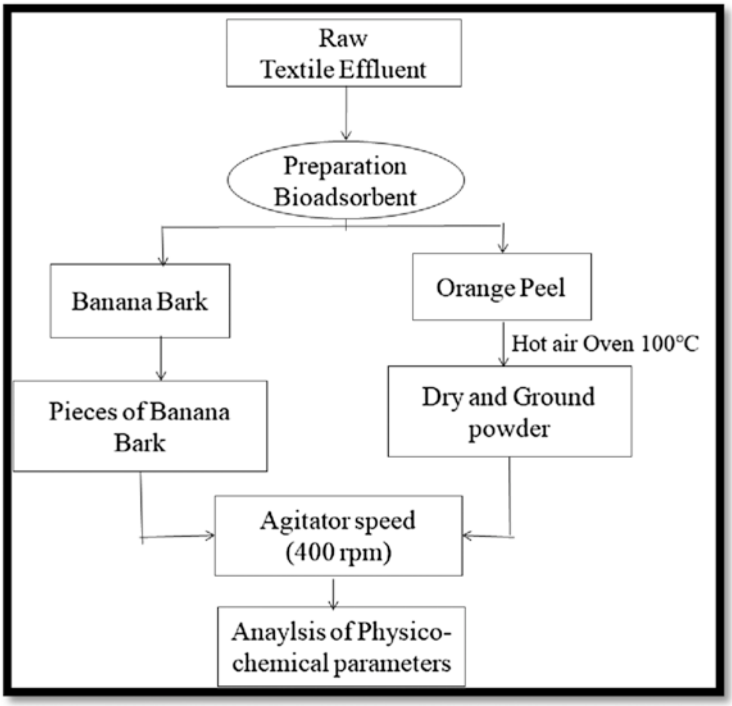


Fig. 14.1 Methods of Preparation adsorbent methods to treat textile effluent

Table 14.1 Describe the method of analysis the textile effluent samples

S. no	Parameters	Methods	References
1	Temperature	Thermometer	[18]
2	pH	Redox potential method	[14]
3	EC	Electrochemical method	[18]
4	TS	Gravimetric method	[27]
5	TDS	Gravimetric method	[4]
6	TSS	Gravimetric method	[2]
7	Alkalinity	Acid–base titration	[3]
8	Total hardness	Complexometric titration	[11]
9	Chloride	Argentometric titration	[2]
11	Dissolved oxygen	Winkler method	[8]
12	COD	Closed reflux method	[38]
13	BOD_5	Winkler method – 5 days incubation at 20°C	[2]

time intervals of 2, 4, 6, 10,12, and 24 hours in room temperature and centrifuged or agitator speed of 4000 rotations per minutes. Fix the conical flask tightly. These processes requires 24 hours for stable settlement, the homogeneous mixture has to settle for 24 hours. The adsorption methods used is isothermal reaction based on mathematical models inference to the complex binding substance between moisture, liquid, and solid relationship to a complexity between homogeneous and heterogeneous solid molecules. The adsorption process efficacy has been calculated by using the mathematical formula given below. [22, 30] C_0 and C_t, are essential factors required in all the physical and chemical parameters determine the oxidation and reduction process depending on reagent usage and dosage. Two factors are introduced in the process kinetic operation (2) for pseudo-first-order (PFO), (3) for pseudo-second-order (PSO) [15, 24]. The Langmuir isotherm explains an envisage approach on the adsorptions phenomenon at specific homogeneous (holistic mixture combination) at an adsorbent surface tension changes, which gives a (4) result of the equation shows variable give the expression on K_L, Langmuir constant (L·mg^{-1}) [25]. The Freundlich variable expression in nonuniform and multilayer adsorption on heterogeneous surfaces is shown in Eq. (14.5).

$$adsorption\left(\%\right) = \frac{Co - Ct}{Co} x100, \tag{14.1}$$

$$\ln\left(qe - qt\right) = \ln qe - K_1 t, \tag{14.2}$$

$$\frac{t}{qt} = \frac{1}{K_2 qe2} + \frac{t}{qe} \tag{14.3}$$

$$\frac{Ce}{qe} = \frac{Ce}{qm} + \frac{1}{KL.qm}, \tag{14.4}$$

$$\log = logKf + \frac{1}{n} loglogCe, \tag{14.5}$$

3 Results and Discussion

3.1 Physicochemical Elements of Textile Waste Water:

The elements of textile wastewater are collected from the Tirupur district, as presented in Table 14.2. The present study explores the feasibility of orange peel and banana bark as a cost-cutting natural adsorbent material concerning various parameters, such as the color adsorbent capability of elements in a lab conditon dosage system and tested for physico- chemical parameter [10].

4 Influence of Temperature

The textile effluents output is higher temperatures after passing through which bio-absorption process. The ambient weather system is taken into a consideration on the bio-absorption to understand there is any change of or variation in dyes temperature which varies from initial 26.2°C respectively (Fig. 14.2). The textile wastewater from the absorbent banana bark and orange peel had a mean temperature of 26.1°C and 25.8°C, respectively, and control mean temperature is 26.4°C during samples testing in lab condition for 24 hours at room temperature adsorption of pollutants on plants. A similar result showed examples. This concept of bio-absorption explains the feeble interaction forces into Van der Waals Forces, and hydrogen bonding reaction has involved due to increase in temperature results in which result

Table 14.2 Quality assessment of physico-chemical parameters of textile industrial wastewater

S. no	Quality of physico-chemical variables	Unit	Textile effluent	BIS standard (17)
1	pH*	–	8.39	7 to 8.5
2	EC*	μS/cm	6.79	NA*
3	Temperature	°C	26.2	NA
4	TS*	Mgl^{-1}	7999	NA
5	TDS*	Mgl^{-1}	7000	500
6	TSS*	Mgl^{-1}	1000	NA
7	TA*	Mgl^{-1}	200	NA
8	Chloride	Mgl^{-1}	662.02	200
9	DO*	Mgl^{-1}	2.43	6
10	TH*	Mgl^{-1}	106	300
11	COD*	Mgl^{-1}	680	250
12	BOD*	Mgl^{-1}	120	NA

Note: NA – not available, mgl^{-1} – milligram per liter, μS/cm

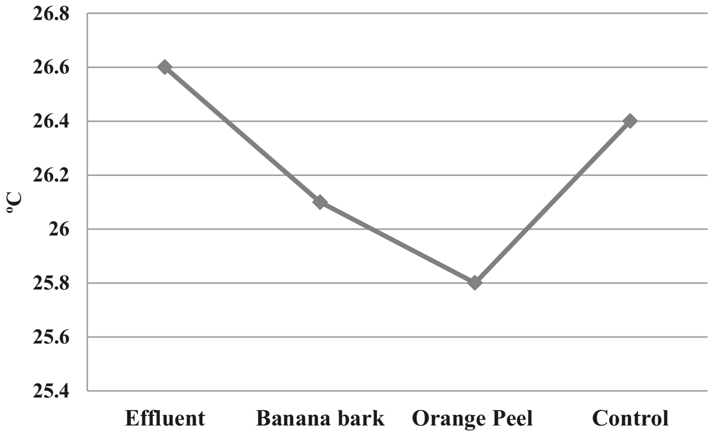


Fig. 14.2 Comparison Temperature of low-cost absorbent banana bark and orange peel

in decrease the temperature of the addition in dye due to the reduction process at higher temperature concentration due to organic matter [16]. Asgher and Bhatti [13] also set-up to reduce the dyes color change used in bio-adsorption respectively with temperature variations.

4.1 Influence of Potential of Hydrogen (pH)*

Hydrogen is an essential parameter to estimate the water status. The difference from the water having potential of hydrogen below 6.5 and above 8.5 is below the permissible limits. The potential of hydrogen of textile effluent is 8.39; the alkaline nature is due to colors, dyes, chemicals, etc. The textile wastewater from Fig. 14.3 shows the bio-absorbent banana bark and orange peel had a mean potential of hydrogen 8.18 and 7.39, and control had a mean pH of 8.02 during a contact time of 24 hours, respectively. pH factor is very critical in adsorption studies, especially for dyes in bio-adsorption. The pH of a medium controls the magnitude of electrostatic charges, which are stated both by the ionized dye molecules and the directions on the surface of the adsorbent. As a result, the rate of adsorption will vary with the pH of an aqueous medium [12, 29].

*4.2 Impact on Electric Conductivity (EC *)*

Electrical conductivity (EC) is an essential water status parameter which reveals the difference between soluble salts by bio-absorption reaction, which explains that bio-absorption is efficient in treating textile effluents. Electrical conductivity from

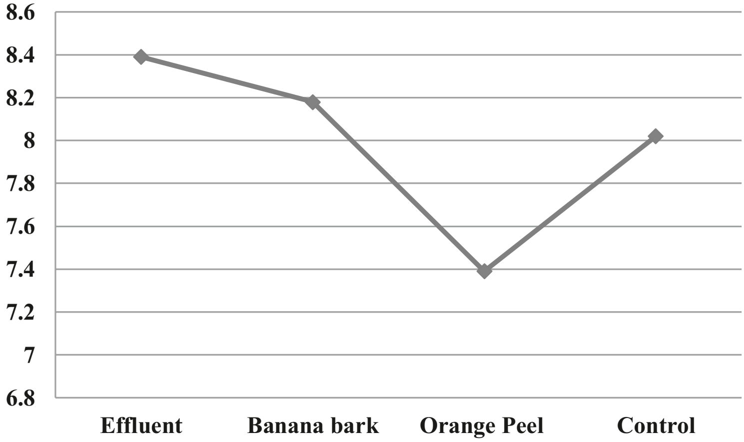


Fig. 14.3 Comparison pH of low-cost absorbent banana bark and orange peel

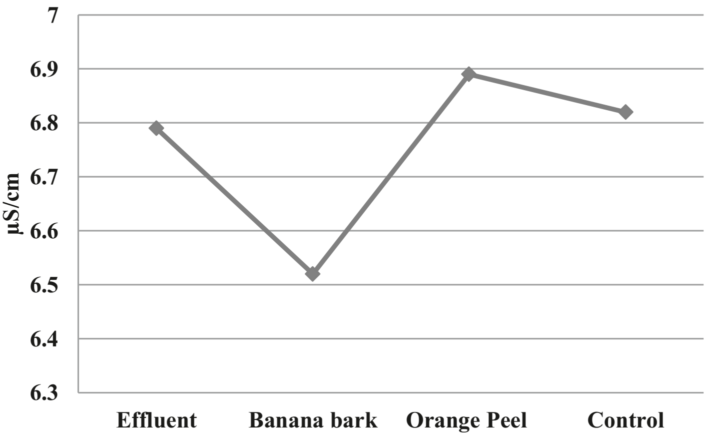


Fig. 14.4 Comparison of EC value at low cost absorbent between banana bark and orange peel

the experimental setup has been noted at 6.79 µS/cm. The effluent from the absorbent banana bark and orange peel absorbents had a mean EC of 6.52 µS/cm and 6.89 µS/cm, respectively, and control had a mean EC of 6.82 µS/cm during 24-hour contact time (Fig. 14.4).

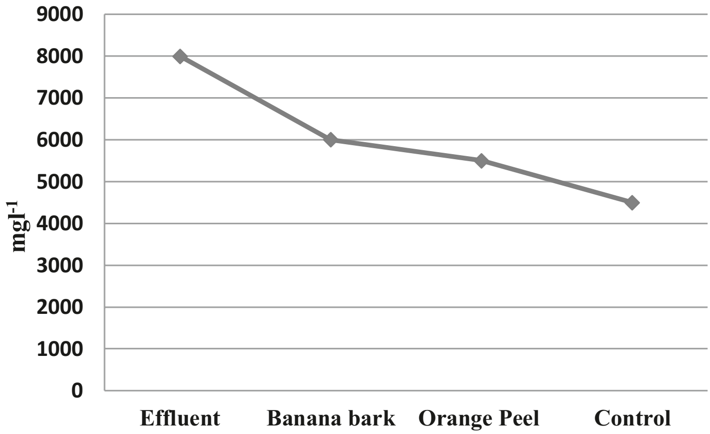


Fig. 14.5 Comparison TS of low-cost absorbent banana bark and orange peel

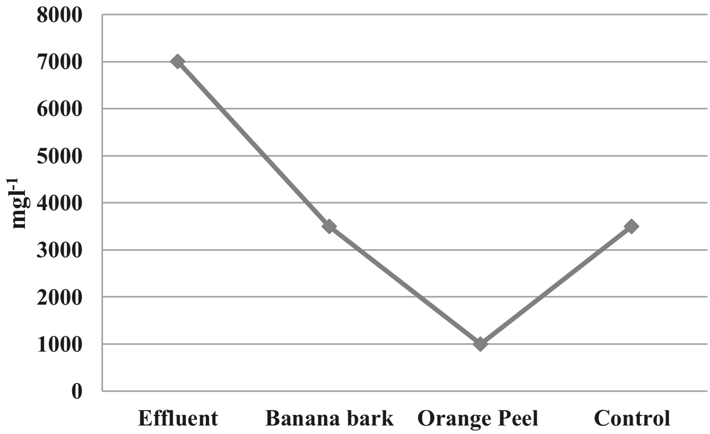


Fig. 14.6 Comparison TDS of low-cost absorbent banana bark and orange peel

*4.3 Impact on (TS *)*

TS of textile wastewater has been observed to be 8000 mgl^{-1}. The banana bark and orange peel absorbents had mean TS of 4500 and 1500 mgl^{-1}, respectively, and control had a TS of 7000 mgl^{-1} during 24-hour contact time (Fig. 14.5).

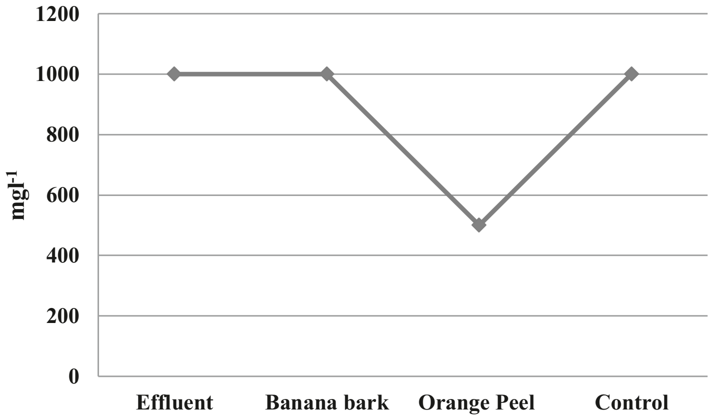


Fig. 14.7 Comparison TSS of low cost absorbent banana bark vs orange peel

4.4 Effect of TDS

The total dissolved solids (TDS) of textile effluents have been observed to be 7000 mgl^{-1}. The effluent from the banana bark and orange peel absorbents had a mean TDS of 3500 and 1000 mgl^{-1}, respectively, control had an indicated TDS of 6000 mgl^{-1} during 24–hour contact time (Fig. 14.6). TDS has insoluble minerals like salts and metals cations and anions undissolved in water.

4.5 Impact on (TSS)*

The observed result from TSS is 1000 mgl^{-1}. This flow into banana bark and orange peel works has a bio-absorbent were, industrial wastewater had a mean TSS of 1000 and 500 mgl^{-1}. Control TSS had a value of 1000 mgl^{-1} during contact of time interval 24 hours (Fig. 14.7).

4.6 Impact on Chloride

The value of chloride parameter in effluent has been observed at 662.02 mgl^{-1}. The effluent using the bio-absorbent banana bark and orange peel reduction had a mean Chloride 481.23 and 18.61 mgl^{-1}, control had a mean value of Chloride 550.5 mgl^{-1} during passing time 24 hour respectively (Fig. 14.8).

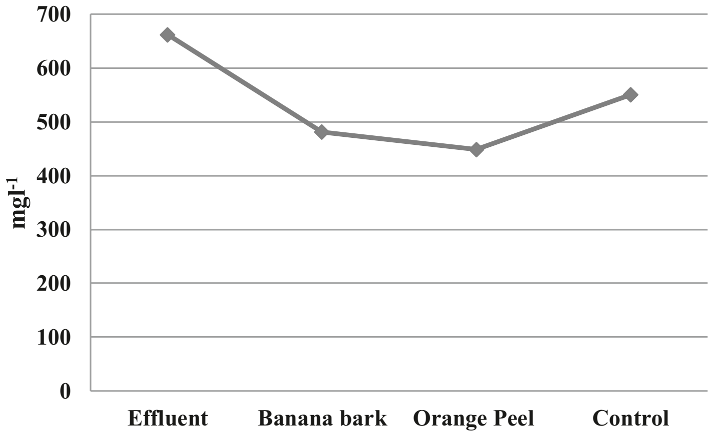


Fig. 14.8 Comparison chlorides of low-cost absorbent banana bark and orange peel

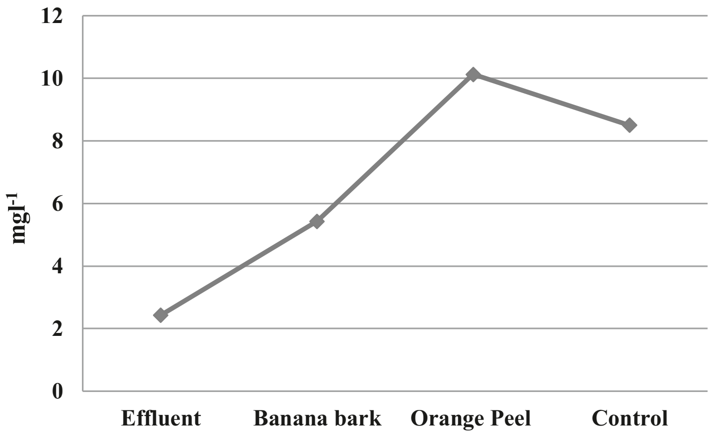


Fig. 14.9 Comparison DO of low-cost absorbent banana bark and orange peel

*4.7 Effect of Dissolved Oxygen (DO *)*

The amount of dissolved oxygen (DO) has been observed at 2.43 mgl^{-1}. The effluent passed through the bio-absorbent banana bark and orange peel had a mean DO of 5.43 and 10.13 mgl^{-1}, respectively. The control treatment for a average mean value is calculated for DO value is 8.5 mgl^{-1} during treatment time 24 hour respectively (Fig. 14.9).

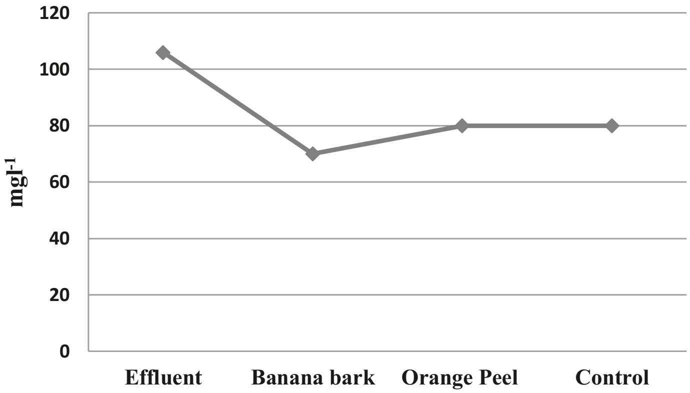


Fig. 14.10 Comparison TH of low-cost absorbent banana bark and orange peel

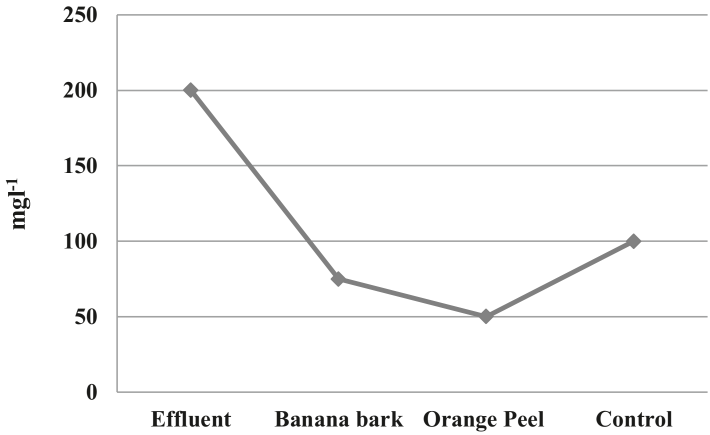


Fig. 14.11 Comparison TA of low-cost absorbent banana bark and orange peel

*4.8 Effect of Total Hardness (TH *)*

The total hardness (TH) has been observed to be 106 mgl^{-1}. The wastewater passed through the absorbent banana bark and orange peel had a total mean hardness of 70 and 80 mgl^{-1}, respectively, and control treatment mean value of total hardness of 80 mgl^{-1} treatment time 24 hour (Fig. 14.10).

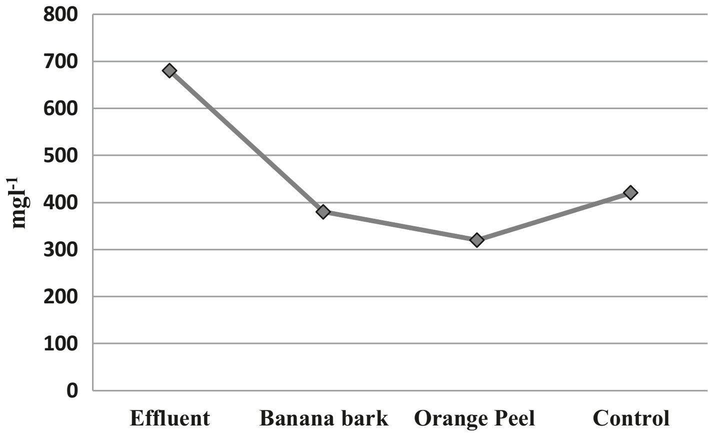


Fig. 14.12 Comparison COD of low-cost absorbent banana bark and orange peel

4.9 Effect of Total Alkalinity

The total alkalinity (TA) of the effluent has been observed to be 200 mgl^{-1}. Wastewater was passed through the bio-absorbent banana bark and orange peel for a mean total alkalinity of 75 and 50 mgl^{-1}, respectively, and control after the reduction treatment, the mean value is 100 mgl^{-1} during contact time 24 hour is notified (Fig. 14.11).

*4.10 Effect of COD**

The Chemical Oxygen Demand (COD) of textile wastewater has been observed to be 680 mgl^{-1}. The effluent from the absorbent banana bark and orange peel had a mean COD of 380 mgl^{-1} and 320 mgl^{-1}, respectively, and control had a mean COD of 420 mgl^{-1} during contact time of 24 hours. Figure 14.12 shows that decrease in adsorbent particle size results in an increase of percent removal of COD, which may be due to the rise in the available surface area for the adsorption process. Patel and Vashi (2010) also worked on the treatment technology of textile wastewater through adsorption. While outcomes are the effect of bio-adsorbent dose, they found similar trends in increasing dye adsorption by amplifying the adsorbent dose.

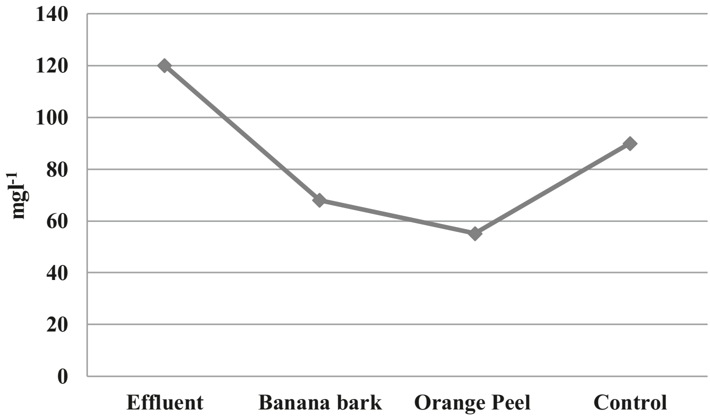


Fig. 14.13 Comparison BOD_5 of low-cost absorbent banana bark and orange peel

Table 14.3 Performance of banana bark and orange peel adsorption using textile effluent

Physicochemical parameters	Percentage removal efficiency		
	Banana bark (absorption)	Orange peel (adsorption)	Control (without)
TS	25	31	12
TDS	50	85	14
TSS	NIL	NIL	20
TA	62	75	NIL
Chloride	27	32	16
DO	NIL	NIL	NIL
Hardness	33	26	12
COD	44	52	39
BOD_5	43	54	25

*4.11 Effect of BOD_5**

The Biological Oxygen Demand (BOD_5) from the textile effluent has been observed to be 120 mgl^{-1}. The effluent from the absorbent banana bark and orange peel had a mean BOD_5 of 68 mgl^{-1} and 55.2 mgl^{-1}, respectively, and control had a mean BOD_5 of 90 mgl^{-1} during contact time of 24 hours (Fig. 14.13). Table 14.3 gives the performance of banana bark and orange peel adsorption using textile effluent.

Table 14.4 Shows the Classification of different concentration for natural assimilation between treatment technologies for textile effluent treatment using bio-adsorbent materials

Bio-adsorbent material	Coagulant used	Textile effluent	Removal of parameters	References
Bentonite adsorbent	Adsorbents	Yes	pH COD – 76%	[37]
Sawdust	Adsorbents	Yes	Colour	[1]
Alum	Using activated carbon	Yes	COD – 90%	[6]
	Using graphene oxide	Yes	Turbidity – 85% Colour – 60%	[28]
	Using activated carbon	Yes	Nil	[7]
Sawdust	Adsorbent	Yes	pH – 45.50% TDS, TSS – 27% Color	[23]
Moringa Oleifera	Coagulant	Yes	pH – Color – 99.9%	[36]
Water lily bark	Coagulant and aquatic	Yes	pH – TDS – 43.38% Colour –	[2]
Orange and banana peel	Synthetic compounds based on iron/aluminum	Yes	pH – 68%	[26]
		Yes	COD – 60.0%	[39]
Adsorbent	Activated charcoal	Yes	COD – 100%	[31]

4.12 *Comparative Study on the Performance by Literature Using Past Study*

The removal efficiency using banana bark and orange peel treatment has a bio-adsorbent material used on textile wastewater shows according to literature sources from Table 14.2. The outcome of the study explains that the condition between the two natural technology treatments should be established in view of comparisons. Table 14.4 classifies the natural assimilation of treatment of textile using bio-adsorbents.

5 Future Scope

Wastewater treatment and management in India has a higher demand to reuse and recycle by a high superior technology (green and clean) and cost-benefit effects towards technologies for development and future projections of wastewater treatment. The use of bio-adsorbents in wastewater treatment reduced expenditure on the processing of costly chemicals. This treatment process is very economical for developing countries. These treated water can be used in garden areas for irrigation purposes, particularly in cities, will have to make use of better and new methods. Industrial wastewater treatment processes are designed to minimize wastewater inflow into the environment for long-term ecological sustainability and recycling of natural and economic resources. The study evaluated energy consumption, operating costs, environmental emissions, and land usage, leading to process optimization and economic development. There are different types of state of the art innovations behind science and engineering technology [15, 40–43].

6 Conclusion

The pilot research has given a lot of insight on the removal of color dyes in the effluent using bio-adsorption material using orange skin (peel) and banana bark on reduction of physical parameters like color bio-adsorption and reduction for removal of higher capacity the retention time should be improved towards increase in the surface tension density mechanically increase the volume with time optimization from 24, 48, 72 hours on concordant basis were the increase in the rate flow can be calculated treatment technology. This can be used as cost-benefit technology (benefit means environmental sustainability), but the demand for raw material can be higher; still, the efficacy has to be tested up to 1000 kiloliters per day so we can find the average actual cost-benefit that can be understood, and also specific design, optimization, operation, maintenance, management, any other technology that has to be used on an add-on process has to be research-checked for sustainability on the dye removal process.

References

1. Abdulsalam, K. A., Giwa, A. R. A., & Adelowo, J. M. (2020). Optimization studies for decolourization of textile wastewater using a sawdust-based adsorbent. *Chemical Data Collections, 27*, 100400.
2. Adnan, A., Taufeeq, A., Malik, E., Irfanullah, M., Masror, K., Muhammad, A., & Khan. (2010). Evaluation of industrial and city effluent quality using physicochemical and biological parameters. *Electronic Journal of Environmental, Agricultural and Food Chemistry, 9*(5), 931–939.
3. Aftab, B. S. Y., Noorjahan, C. M., & Dawood, S. S. (2005). Physico-chemical and fungal analysis of a fertilizer factory effluent. *Nature, Environment and Pollution Technology, 4*(4), 529–531.
4. Agarwal, A., & Manish, S. (2011). Assessment of pollution by physicochemical water parameters using regression analysis: A case study of Gagan River at Moradabad – India. *Advances in Applied Science Research, 2*(2), 185–189.
5. Ai, L., Zhang, C., Liao, F., et al. (2011). Removal of methylene blue from aqueous solution with magnetite loaded multi-wall carbon nanotube: Kinetic, isotherm and mechanism analysis. *Journal of Hazardous Materials, 198*, 282–290.
6. Aleem, M., Cao, J., Li, C., et al. (2020). Coagulation and adsorption based environmental impact assessment and textile effluent treatment. *Water, Air, and Soil Pollution, 45*, 231.

7. Amalraj, A., & Pius, A. (2017). Removal of fluoride from drinking water using aluminum hydroxide coated activated carbon prepared from bark of *Morinda tinctoria*. *Applied Water Science, 7*, 2653–2665.
8. American Public Health Association (APHA). (2005). *Standard methods for the examination of water and wastewater* (21st ed.). APHA.
9. Anastopoulos, I., & Kyzas, G. Z. (2014). Agricultural peels for dye adsorption: a review of recent literature. *Journal of Molecular Liquids, 200*, 381–389.
10. APHA. (1998). *Standard methods for the examination of water and wastewater*. American Public Health Association; American Water Works Association; Water Environment Federation.
11. Arivoli, A., & Mohanraj. (2013). Efficacy of *Typha angustifolia* based vertical flow constructed wetland system in pollutant reduction of domestic wastewater. *International Journal of Environmental Sciences, 3*(5), 1497–1508.
12. Islam, A., & Arun, K. G. (2013). Removal of pH, TDS and color from textile effluent by using coagulants and aquatic/non aquatic plants as adsorbents. *Resources and Environmental, 3*(5), 101–114.
13. Asgher, M., & Bhatti, H. N. (2012). Evaluation of thermodynamics and effect of chemical treatments on sorption potential of citrus waste biomass for removal of anionic dyes fromaqueous solutions. *Ecological Engineering, 38*, 79–85.
14. AWWA. (2006). *Standard methods for the examination of water and waste water*. American Water Works Association.
15. Bezerra de Araujo, C. M., Filipe Oliveira do Nascimento, G., Rodrigues Bezerra Da Costa, G., et al. (2018). Adsorptive removal of dye from real textile wastewater using graphene oxide produced via modifications of Hummers method. *Chemical Engineering Communications, 206*, 1–13.
16. Chatterjee, S., Lee, D. S., Lee, M. W., & Woo, S. H. (2009). Congo red adsorption from aqueous solutions by using chitosan hydrogel beads impregnated with nonionic or anionic surfactants. *Bio Resource Technology, 100*, 3862–3868.
17. Cotoruelo, M. L., Marques, M. D., & Rodriguez, J. J. (2007). Adsorption of aromatic compounds on activated carbons from lignin: Equilibrium and thermodynamic study. *Industrial and engineering Chemical Research, 46*, 4982–4990.
18. CPCB. (2000). *Water quality status of Yamuna river: Assessment and development of river basin*. Central Pollution Control Board.
19. HuQH, Q. S. Z., Haghseresht, F., Wilson, M. A., & Lu, G. Q. (2006). Adsorption study for removal of basic red dye using bentonite. *Industrial and Engineering Chemical Research, 45*, 733–738.
20. Irina, I. S., & Romen, B. (2008). Wastewater characteristics in textile finishing Mills. *Environmental Engineering and Management Journal, 7*(6), 859–864.
21. Jain, R., Sharma, N., & Radhapyar, K. (2009). Electrochemical treatment of pharmaceutical azo dye amaranth from waste water. *Journal of Applied Electrochemical, 39*, 577–582.
22. Kachabi, M., Mrabet, I. E., Benchekroun, Z., Nawdali, M., & Hicham, Z. (2019). Synthesis and adsorption properties of activated carbon from KOH-activation of Moroccan Jujube shells for the removal of COD and color from wastewater. *Mediterranean Journal of Chemistry, 8*(3), 68–178.
23. Khatmode, N. P., & Thakare, S. B. (2015). Removal of pH, TDS, TSS and color from textile effluent by using sawdust as adsorbent. *International Journal of Science Basic Applied Research, 24*(2), 158–163.
24. Larrechi, M. S., Callao, M. P., & Gome, V. (2007). Kinetic and adsorption study of acid dye removal using activated carbon. *Chemosphere, 69*, 1151–1158.
25. Loubna, N., Miyah, Y., Assila, O., El Badraoui, A., El Khazzan, B., & Zerrouq, F. (2019). Kinetic and thermodynamic study of the adsorption of twodyes: brilliant green and eriochrome black T using a natural adsorbent sugarcane bagasse. *Moroccan Journal of Chemistry, 7*, 715–726.

26. Mane, R. S., & Bhusari, V. N. (2012). Removal of colour (dyes) from textile effluent by adsorption using Orange and Banana peel. *International Journal of Engineering Research and Applications, 2*(3), 1997–2004.
27. Manivaskam, N. (2011). *Physicochemical examination of water sewage and industrial effluent* (6th ed.). Pragati Prakashan.
28. Mejbar, F., Miyah, Y., Badraoui, A. E., et al. (2019). Studies of the adsorption kinetics process for removal of methylene blue dye by residue of grenadine bark extraction. *Moroccan Journal of Chemistry, 6*, 436–444.
29. Larrechi, M. S., Callao, M. P., & Gomez, V. (2007). Kinetic and adsorption study of acid dye removal using activated carbon. *Chemosphere, 69*, 1151–1158.
30. Ncibi, M. C., Mahjoub, B., & Seffen, M. (2007). Adsorptive removal of textile reactive dye using *Posidonia oceanica* (L.) fibrous biomass. *International journal of Environmental Science and Technology, 4*(4), 433–440.
31. Patel, H., & Vashi, R. T. (2010). Treatment of textile wastewater by adsorption and coagulation. *E-Journal of Chemistry, 7*(4), 1468–1476.
32. Senthilkumar, S., Kalaamani, P., Porkodi, K., Varadarajan, P. R., & Subburaam, C. (2006). *Journal of Bioresearch and Technology, 97*, 1618–1625.
33. Sivakumar, D., & Shankar, D. (2012). Effect of aeration on colour removal from textile industry wastewater. *International Journal of Environmental Sciences, 2*(3), 1386–1397.
34. Smita, V., Pandey, N. D., & Quoff, A. R. (2014). Decolorization of synthetic dye solution containing Congo red by advanced oxidation process (AOP). *International Journal of Advanced Research in Civil, Structural, Environmental and Infrastructure Engineering and Developing, 2*(1), 49–55.
35. Somasekhara Reddy, M. C. (2006). Removal of direct dye from aqueous solutions with an adsorbent made from tamarind fruit shell, an agricultural solid waste. *Journal of Scientific and Industrial Research, 65*, 443–446.
36. Alen, S., & Vinodha, S. (2014). Studies on colour removal efficiency of textile dyeing waste water using *Moringo Olifer*. *International Journal of Civil Engineering, 1*(5), 6–10.
37. Tebeje, A., Worku, Z., Nkambule, T. T. I., et al. (2021). Adsorption of chemical oxygen demand from textile industrial wastewater through locally prepared bentonite adsorbent. *International journal of Environmental Science and Technology*.
38. World Health Organization (WHO). (2011). *Guidelines for drinking-water quality* (4th ed.). WHO.
39. Zidane, F., Rhazzar, A., Blais, J. F., et al. (2011). Contribution a la pollution des caux uses composes a base de fer et d aluminium. *International of Biological and Chemical Sciences, 5*, 1727–1745.
40. Hasan, I., & Agarwal, P. (2021). Analytical approach to sustainable smart city using IoT and machine learning. In R. Agrawal, M. Paprzycki, & N. Gupta (Eds.), *Big data, IoT, and machine learning*. CRC Press. https://doi.org/10.1201/9780429322990
41. Agarwal, P., Chopra, K., Kashif, M., & Kumari, V. (2018). Implementing ALPR for detection of traffic violations: A step towards sustainability, 2018. In *Proceedia: Computer science* (pp. 738–743). Elsevier (Scopus indexed).
42. Abdur Rahman, F. B., Akter, M., & Abedin, M. Z. (2013). Dyes removal from textile wastewater using orange peels. *International Journal of Scientific and Technology Research, 2*(9), 47–50.
43. Nguyet, P. N., Watari, T., Hirakata, Y., Hatamoto, M., & Yamaguchi, T. (2019). Adsorption and biodegradation removal of methylene blue in a down-flow hanging filter reactor incorporating natural adsorbent. *Environmental Technology*.

Index

J. Rodrigues et al. (eds.), *IoT for Sustainable Smart Cities and Society*,
Internet of Things, https://doi.org/10.1007/978-3-030-89554-9