IoT-Based Smart Water Monitoring using Machine Learning and Deep Learning Techniques

A thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Electronics and Communications Engineering

by

Ayush Kumar Lall
2020122001
ayush.lall@research.iiit.ac.in

Advisor: Dr. Sachin Chaudhari

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

International Institute of Information Technology Hyderabad
500 032, India

December 2023

International Institute of Information Technology Hyderabad
Hyderabad, India

CERTIFICATE

This is to certify that work presented in this thesis proposal titled ***IoT-Based Smart Water Monitoring: Using Machine Learning and Deep Learning Techniques*** by *Ayush Kumar Lall* has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Advisor: Dr. Sachin Chaudhari

To

My Family and Friends

Acknowledgments

I would like to express my heartfelt gratitude to Dr Sachin Chaudhari for his invaluable guidance and unwavering support throughout my thesis journey. His mentorship and insightful feedback have been instrumental in shaping the direction of my research. There is not even a single time when he is not approachable. He always showed a friendly nature and was always there to help me out if I got stuck somewhere. I still remember joining this prestige institute was a big cultural and academic change compared to where I was earlier. But Sachin sir was always there to ensure that I kept up with the pace of the institute's expectations and guided me from research to course-related doubts. I am really blessed that Prof. Sachin always had faith in me and believed I could take projects based on internet-of-things (IoT), which require applications from diverse domains. I will always be thankful to him for shaping the researcher in me from my undergraduate level and always showing his support at every stage of the journey.

I am profoundly appreciative of my co-authors, namely Prof. K.S. Rajan, Nitin Nilesh, Ayush Dwivedi, Nilesh Bawankar, Ansh Khandelwal, Rishikesh Bose, Archit Goyal, Aakash Terala, and my other lab mates, who provided unwavering assistance and guidance throughout every phase of my research expedition.

I extend my appreciation to the Smart City Research Centre especially to Ms Anuradha Vattem for offering immense support that greatly contributed to the successful completion of this project. Additionally, I am indebted to TIH-IoT (IIT-Bombay) for awarding me the CHANAKYA Fellowship, which played a pivotal role in advancing the progress of this research endeavour.

My heartfelt gratitude extends to my friends and lab companions, including Ayu, Rajashekar, Spanddhana, Usha, Viswanadh, Sasanka, Ihita, Ruchi, David, and the entire supportive community. The knowledge I have acquired during these years in the lab will always hold a special place in my heart. As new students like Om, Ritik, Ankit, and others embark on their remarkable journey within the lab, I wish them all the best in their forthcoming endeavours.

Finally, I would like to show my sincere gratitude towards my Mother, Father and cherished family members and friends. Their unwavering presence through the highs and lows of my journey has been a steadfast source of support, akin to pillars of strength. Their unwavering belief has endowed me with inner resilience and propelled me forward during moments of adversity.

Abstract

The conventional method of manually reading analog meters to track consumption trends is both laborious and costly. Moreover, it falls short in effectively managing sustainable water supplies, necessitating accurate monitoring techniques to provide real-time insights into water usage for consumers. While digital water meters have been introduced, their high cost makes them impractical for widespread adoption, and they lack analytical capabilities for interpreting consumption patterns. In contrast, traditional analog water meters boast simplicity, low power consumption, durability, and reliability, but they still rely on manual readings, which is inconvenient. To address this issue, an automated data-capturing system is required to transmit real-time meter readings to a cloud server. Smart water meters, powered by robust machine learning (ML) and deep learning (DL) algorithms for meter reading detection, can analyze the data collected to gain valuable insights into water consumption patterns and detect leaks, facilitating more efficient water management. Ultimately, smart water monitoring devices can empower users to reduce their water usage and contribute to water conservation efforts. Keeping all these aspects in mind, the thesis can be divided into three parts:

Firstly, this thesis introduces an IoT based economic retrofitting setup for digitising the analog water meters to make them smart. The setup contains a Raspberry-Pi microcontroller and a Pi-camera mounted on top of the analog water meter to take its images. The captured images are then preprocessed to estimate readings using a ML model. The employed ML algorithm is trained on a rich dataset that includes digits from the images of water meters captured by the hardware setup for ten days. The readings are posted on a cloud server in real-time using Raspberry-Pi. High temporal resolution plots of flow rate and volume are generated to derive inferences. The collected data can be used for deriving water consumption patterns and fault detection for efficient water management.

After that, the thesis proposes a DL-based algorithm which is used for improving the performance of digit detection from IoT-based analog water meters. The DL algorithm is trained on a rich dataset of over 160,000 images collected from six water nodes deployed at locations with different environmental conditions. A detailed comparison between the proposed DL and ML algorithm is made based on detection accuracy, feature analysis, error analysis, and computational complexity analysis. It is observed that compared to the ML model, the proposed DL model maintained a higher detection accuracy and is more generalized in terms of feature extraction, which makes the algorithm robust.

Finally, the thesis presents a comprehensive analysis of water supply behaviour on an educational campus, focusing on two distinct regions: student hostels and faculty/staff quarters. The investigation

delves into the impact of water supply patterns on a monthly and weekly basis. Notably, it highlights how each month, with its unique characteristics such as holidays, exams, and class schedules, influences the water supply in both regions. One key difference between the two regions is that students reside in one, leading to significant variations in water usage based on the number of holidays. Conversely, the other region accommodates families, resulting in a consistent water requirement regardless of college holidays. The findings from this analysis are crucial for understanding water distribution patterns, particularly within intermittent water supply (IWS) systems, with the ultimate goal of enhancing the efficiency and robustness of water distribution. By thoroughly examining the water supply behaviour in an educational campus and considering various factors that influence it, this work contributes to a better understanding of water management on campus.

Contents

List of Figures

List of Tables

Abbreviations

AI	Artificial Intelligence
AWS	Amazon Web Services
BCM	billion cubic meters
BIS	Bureau of Indian Standards
CNN	convolution neural network
CoAP	Constrained Application Protocol
CPCB	Central Pollution Control Board
CWS	continuous water supply
DL	deep learning
ECG	Electro Cardio Gram
HOG	Histogram of Oriented Gradients
HTTP	Hypertext Transfer Protocol
IoT	internet-of-things
IWS	intermittent water supply
LoRaWAN	Long Range Wide Area Network
ML	machine learning
MQTT	Message Queuing Telemetry Transport
NAMP	National Air Quality Monitoring Program
NLP	Natural Language Processing
NRW	non-revenue water
NWMP	National Water Quality Monitoring Programme
OHT	Overhead Tank
RF	Random Forest
RoI	region of interest
SWaMM	Smart Water Metering Middleware
YOLO	You Only Look Once

List of Related Publications

Conference papers published:

[P1] A.K.Lall, A. Khandelwal, R. Bose, N. Bawankar, N. Nilesh, A. Dwivedi, S. Chaudhari, **"Making Analog Water Meter Smart using ML and IoT-based Low-Cost Retrofitting"**, in proceedings of *8th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2021.

[P2] A.K.Lall, A. Khandelwal, N. Nilesh, S. Chaudhari, **"Improving IoT-based Smart Retrofit Model for Analog Water Meters using DL based Algorithm"**, in proceedings of *9th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2022.

Patent filed:

[P1] A.K. Lall, A. Khandelwal, R. Bose, N. Bawankar, N. Nilesh, A. Dwivedi, S. Chaudhari, **"System and Method for Digitizing a Reading in an Analog Water Meter Using Machine Learning"**, India Patent Appl. Num. 202 141 021 341, September, 2022.

Journal submitted:

[P1] A.K. Lall, A. Terala, A. Goyal, S. Chaudhari, K. Rajan **"Behavioural Analysis of Water Consumption Data Using IoT-based Smart Retrofit Meter"**, Elsevier IoT journal, 2023. (View [PDF])

Chapter 1

Introduction

1.1 Motivation

The Internet of Things (IoT) is driving a transformative revolution across the globe, interconnecting billions of devices and objects to the Internet, enabling seamless communication, data exchange, and automation. This interconnected ecosystem is transforming various industries and aspects of daily life without much human interference [1]. From small wearables to large assembly machines, IoT finds applications in diverse domains, vastly improving work efficiency in our day-to-day activities. By effortlessly collecting vast amounts of data, IoT empowers us to analyze and understand the world around us, opening avenues for improvement [2]. Leveraging the power of Artificial Intelligence (AI) algorithms, IoT becomes more intelligent, autonomous, and adaptive [3]. This potent integration of IoT with computer vision, Natural Language Processing (NLP), machine learning (ML), and deep learning (DL)-based algorithms allows real-time data collection with exceptional precision. The amalgamation of IoT systems, sensor interconnections, and AI-based algorithms enables the identification of patterns, trends, and anomalies that might otherwise remain unnoticed, providing invaluable insights for progress.

The world is grappling with a significant challenge: an inefficient water management system [4]. In India, the current water monitoring network, part of the National Water Quality Monitoring Programme (NWMP), consists of 4484 stations for surface and groundwater across 28 States and 8 Union Territories, with monitoring conducted on a monthly, quarterly, half-yearly, and yearly basis [5]. Despite Government efforts to improve the situation, there remains a pressing need for responsible water usage from the end-users to conserve this precious resource. Freshwater scarcity is a severe issue, with many households lacking access to this basic necessity. Although India receives about 4000 billion cubic meters (BCM) of annual precipitation, only a mere 8% of the rainfall is captured [6]. Apart from concerns regarding drinking water scarcity, the shortage of water for agricultural purposes is also on the rise [7,8]. Numerous water management systems have been proposed in the past, employing various technologies to address the issue, but these solutions often come with high costs and energy consumption. However, with the emergence of the IoT, the development of smart water management systems is gaining momentum [9]. Real-time water monitoring systems offer promising solutions for leak prevention in pipelines, efficient water distribution among households, reduction of non-revenue water (NRW), water quality assessment, and understanding water usage patterns, leading to responsible household water consumption [10]. Several IoT-based water management systems are under development, some of which can measure water levels in real-time [11], monitor water quality [12], and assess soil moisture [13]. These advancements hold the potential to revolutionize water management, helping us conserve and utilize this vital resource more sustainably.

Deployment of smart IoT-based water meters can make efficient management of water supply and keep a check on water usage. These devices can help the user understand their water usage based on which they can optimize their usage pattern and monitor consumption regularly [14], [15], [16], [17]. In particular, the thesis provides a method to make preinstalled traditional analog meters smart and use them to monitor water usage. The work combines the ML/DL-based algorithms and computer-vision-based techniques to make the IoT-based smart water monitoring system robust and easy to deploy. The smart meters are designed such that they do not alter the preexisting water pipelines and easily get retrofit on the preinstalled traditional analog meters, making them a low-cost alternative compared to high-cost digital meters. The thesis provides a detailed understanding of the working and designing of smart meters, which are deployed across the campus of IIIT-Hyderabad, collecting data in real time. A huge amount of dataset was collected, which is one of its kind based on Indian analog water meters and is rich in features. The data collected from them provides a useful understanding of water usage patterns concerning different campus buildings.

1.2 Summary of contributions

The main contributions from this thesis are presented in the chapters mentioned as follows:

- **Chapter 3**
  - An IoT-based sensor node is designed and deployed in field to take water meter images, convert them into digits and send it to cloud.
  - The existing meter is not physically altered as a retrofitting method is employed. It is assumed that the meter fully depends on the conversion of pictures to digits and lacks any pulse output.
  - The node's design allows it to deliver data in real time with great temporal precision (readings every few seconds) and is outfitted with lighting so that readings can be taken at night. In this manner, the data from the metres can also provide precise estimations of derived factors like flow rate.
  - The meter image's digits are recognised using a simple ML method that can be implemented at the node itself and requires little processing.
  - The performance of the ML algorithm is further improved by using specific constraints related to the water meters.

  - The proposed approach is evaluated based on the data of over 10,000 images collected from the field deployment of 10 days. The collected data will be made public for further research in future.

    Note: In this work, I was involved in refining the dataset, training the ML algorithm, making the algorithm work on Raspberry Pi, and generating time series graph. For designing the retrofit hardware setup, credits belong to Nilesh Bawankar.

- **Chapter 4**

  - A DL-based algorithm is proposed, which performs more accurately than the ML algorithm described in Chapter 3. The proposed DL algorithm is based on CNN with transfer learning.
  - A rich dataset is collected using images captured from six IoT-based smart water meters deployed in locations having different environmental conditions. For training, approximately 160,000 images have been used, which have been collected over 20 days. The testing data contains approximately 80,000 images collected over 10 days in real time.
  - A performance comparison against the proposed DL-based algorithm is carried out with the ML-based algorithm proposed in Chapter 3. It is shown that the DL-based algorithm performs better than the ML-based algorithm, even without the need for post-processing constraints to correct the false detections. Moreover, the proposed DL algorithm is computationally light enough to be deployed on our retrofit model (on edge).
  - To understand the performance improvement of the proposed DL model over the ML-based model, the extracted features from both the models are analyzed to infer what models are observing during detection.

- **Chapter 5**

  - Behavioural analysis of water consumption of educational institute campus buildings.
  - Huge data collection from water meters via IoT-based retrofit over an academic semester.
  - Monthly and weekly analysis of the water supply patterns for student's hostel block and faculty/staff block having two different demand requirements.
  - Analysis of variation of water supply pattern with holidays in semesters between hostel block and faculty/staff.

1.3 Structure of thesis

The rest of this thesis is organized as follows-

- **Chapter 2** gives an overview of IoT and provides information on the literature survey on smart analog water meters.

- **Chapter 3** describes an ML-based IoT retrofit method to make analog water meters smart.
- **Chapter 4** illustrates the necessity for employing a DL-based approach to accurately recognize digits in analog water meters.
- **Chapter 5** presents behavioural analysis on the water consumption data for two regions on the IIIT-H campus.
- **Chapter 6** conclude this thesis.

Chapter 2

An Overview on IoT

This chapter provides a comprehensive overview of IoT systems, encompassing their architecture, applications in diverse domains, and the challenges they entail. To facilitate a deeper understanding of IoT, the chapter delves into a four-layer architecture, outlining the fundamental steps involved in building an IoT system.

2.1 Introduction to IoT

The Internet of Things (IoT) refers to the network of interconnected physical devices, vehicles, appliances, and other objects that are embedded with sensors, software, and network connectivity, allowing them to collect and exchange data over the internet. In simple terms, it is the concept of connecting everyday objects to the internet, enabling them to communicate and interact with each other and with humans. The IoT is built on the idea of enabling devices to gather and share data, which can then be analyzed and used to make informed decisions, automate processes, and improve efficiency. These devices can range from small, simple sensors to complex machinery, and they can be found in various environments such as homes, offices, factories, cities, and even in wearable devices.

2.2 IoT architecture

IoT architecture refers to the structure and components that make up an IoT system. It provides a framework for designing and implementing IoT solutions, allowing devices and applications to connect, communicate, and share data seamlessly. The architecture typically consists of several layers and they can vary between three to seven [18], [19], [20], [21], [22]. However, this chapter will cover an in-depth understanding of four-layer architecture 2.1, which is comprised of 1. perception/sensor layer, 2. processing layer, 3. network layer, and 4. cloud/platform Layer.

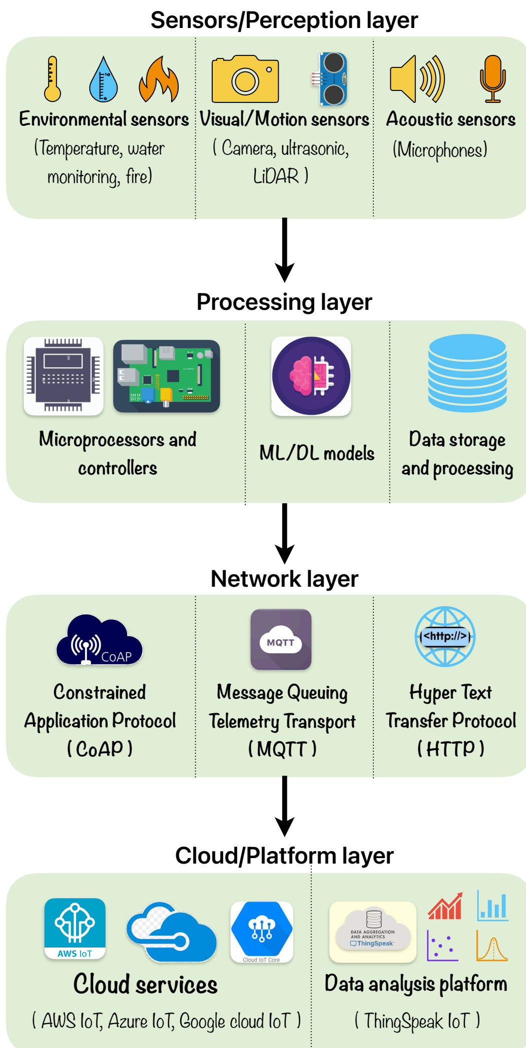


Figure 2.1 A four-layer IoT architecture consisting of perception/sensor, network, processing and cloud/platform layers.

2.2.1 Perception/sensor layer

This is the lowest layer of the architecture, consisting of physical devices such as sensors, actuators, and controllers. These devices collect data from the environment or take actions based on commands received from higher layers. They often have embedded processors, wireless communication capabilities, and various sensors to capture data. The device layer interacts with the physical world and converts analog signals into digital data. This layer performs the edge computation and analytics at the node level. They play a critical role in interacting with the surrounding environment and gathering essential information and data. This collected data serves as a valuable resource for further analysis and insights.

A wide range of sensor types are available, each with its distinct characteristics and functionalities; for example:

- Visual sensor: The data collected from them can be of various types like face information, digit and text information, and various physical object information. This data can be used for smart city surveillance [23], water meter monitoring [24] and other smart applications.
- Motion sensor: It detects movement or changes in the surrounding environment. It is commonly used in various applications, including security systems, home automation, lighting control, and occupancy detection. Some of the examples are IR sensors, ultrasonic sensors, accelerometers, gyroscopes etc.
- Environmental sensors: These sensors are engineered to collect data about the surrounding environment's physical and chemical characteristics. For instance, DHT11 sensors excel at measuring both temperature and humidity levels in their vicinity, while MQ135 sensors are proficient in gauging diverse gas compositions and monitoring air quality.

2.2.2 Processing layer

They serve as the central processing unit and the core intelligence behind IoT devices. They play a vital role in controlling and facilitating communication between the device's sensors and actuators. With their robust computing power, processors efficiently handle data processing tasks and facilitate seamless transmission to the user end. The IoT landscape offers a diverse range of processors and controllers, each boasting distinct capabilities tailored to specific requirements. For instance:

- Raspberry-pi: Raspberry Pi boards are designed to be a complete computer system on a single board. They consist of a system-on-a-chip (SoC) that integrates a processor, memory, graphics processing unit (GPU), and various input/output (I/O) ports. They are used in IoT for data collection, local processing, connectivity, gateway functionality, customization, and development. They provide a versatile and affordable platform for building IoT solutions.
- Node-MCU: They are open-source firmware and development boards based on the ESP8266 Wi-Fi module. It combines a microcontroller unit (MCU) and Wi-Fi capabilities, making it a popular choice for Internet of Things (IoT) projects.

2.2.3 Network layer

The network layer in an IoT system plays a vital role in facilitating seamless communication between devices. It encompasses the network architecture, gateways, and protocols that enable efficient data transfer. This layer incorporates various technologies such as local area networks (LANs), wide area networks (WANs), Wi-Fi, Bluetooth, and cellular networks. Its primary function is to transmit and process data collected by the sensor devices. The network layer establishes a foundation for higher layers to further process and analyze the data by ensuring reliable and secure data transmission. Some of the communication protocols commonly used in the network layer of IoT systems:

- **Message Queuing Telemetry Transport (MQTT)**: MQTT is a lightweight publish-subscribe messaging protocol that is designed for efficient communication in constrained environments. It is widely used in IoT applications for its low overhead and support for reliable messaging. They can be combined with controllers like NodeMcu [25] in order to send to user.

- **Constrained Application Protocol (CoAP)**: CoAP is a specialized protocol designed for resource-constrained devices and networks. It enables efficient communication between IoT devices and is often used in applications that require low power consumption and scalability. There are some work in literature like [26], [27] which which enables the users to select the CoAP that aligns best with the unique needs and requirements of each application

- **Hypertext Transfer Protocol (HTTP)**: Although primarily a protocol for web communication, HTTP is also utilized in the IoT network layer. It allows devices to communicate using the request-response model and is commonly used for device management, data retrieval, and control in IoT applications.

- **Long Range Wide Area Network (LoRaWAN)**: LoRaWAN is a low-power, wide-area network protocol designed for long-range communication [28]. It enables devices to transmit data over long distances with low power consumption, making it suitable for applications such as smart cities and agriculture.

- **Zigbee**: Zigbee is a wireless communication protocol designed for low-power, low-data-rate applications. It is commonly used in home automation, industrial control systems, and sensor networks where devices need to communicate wirelessly with each other. Reader can also go through its performance review in [29], which provides comprehensive performance analysis of both encrypted and unencrypted Zigbee, using specified metrics in a real-world testbed.

There are other protocols in IoT network and with new of them getting added. Each have their unique characteristics and vary with the user requirements.

2.2.4 Cloud/Platform layer

The cloud or platform layer serves as the central processing and storage hub for IoT data. It involves cloud-based services, edge computing, or a combination of both. Data collected from devices is sent to the cloud/platform for storage, analysis, and processing. This layer may include data management systems, data analytics tools, and machine learning algorithms to derive insights from the collected data. It also provides APIs and interfaces for application development and integration. The data collected from the IoT devices are very diverse and in large amount, thus different ways and methods are required to analyse them. For different uses there are several cloud platforms which perform tasks like visualisation of data, monitoring management, device management, system management etc. Cloud platforms such as Amazon Web Services (AWS), IBM IoT, Oracle IoT cloud, ThingSpeak etc. can handle one or more of the stated tasks. These cloud platforms have several advantages which can increase the efficiency of the IoT system. For example:

- Scalability: Cloud platforms provide the ability to easily scale IoT systems to handle changing demands and large data volumes. Resources can be dynamically adjusted to ensure optimal resource utilization and accommodate system growth.
- Flexibility: With cloud platforms, users have the freedom to choose from a range of infrastructure and services tailored to their specific requirements. This includes options for data storage, computing power, analytics tools, and machine learning capabilities, allowing for customization and adaptability.
- Data Storage and Analytics: The storage of data produced by IoT devices can be handled securely and effectively by the extensive storage capabilities offered by cloud systems. Large datasets can be processed and analysed using sophisticated analytics tools and machine learning algorithms, which can yield insightful data that can be used to make well-informed decisions.
- Security: Cloud platforms prioritize the security of IoT data and systems. They employ multiple layers of protection, including encryption, access control, authentication mechanisms, and data privacy measures. Compliance with industry security standards and regular audits further ensure a secure environment for IoT deployments.

2.3 Applications of IoT

IoT has brought the world even more closer than ever. With its elegant design and methods, it has brought a multitude of conveniences, efficiencies, and advancements across various aspects of life. It allows various devices to interact with each other, enabling smart homes for task automation, enhanced security, and energy optimization. It can facilitate remote patient monitoring and personalized care through wearables and telemedicine in hospitals. IoT-driven agriculture optimizes resources and boosts crop yield sustainably. Transportation systems benefit from smart traffic management and efficient fleet tracking, improving safety and reducing congestion. IoT enables personalized marketing, seamless inventory management, and efficient supply chain operations in retail. Some of the applications are briefly discussed below:

- **Smart cities:** IoT plays a pivotal role in fostering sustainable and efficient cities, and the Indian government is actively spearheading various initiatives to leverage its potential. These initiatives encompass smart applications such as intelligent lighting, waste and traffic management, environmental monitoring, public safety systems, and optimized energy distribution. Smart cities harness IoT technology to elevate the overall quality of life and streamline resource management. For instance, the implementation of smart water meters transforms traditional analog meters into digital counterparts, providing valuable data on water consumption patterns [24, 30]. Additionally, air pollution monitoring systems [31] [32], like the National Air Quality Monitoring Program (NAMP) and the Central Pollution Control Board (CPCB), are of utmost importance for India, and the data they provide is accessible to the public, even at the state level [5, 33]. Leveraging IoT-based solutions has the potential to significantly mitigate air pollution, empowering the Government and relevant agencies to effectively analyze the data and address the issue. Furthermore, ongoing efforts focus on understanding particulate matter density and its impact on pollution levels [34], [35]. In the realm of smart traffic management, IoT-derived traffic data can be analyzed and utilized for intelligent signalling systems, optimizing traffic flow and ultimately reducing accidents and congestion [36]. Employing sensors and real-time data analysis for smart energy management can greatly enhance sustainable energy practices in tourist destinations across the country [37]. By incorporating technologies like smart street lights and energy-efficient buildings, India can make substantial strides in efficient energy management, catering to its diverse and vibrant tourist spots.

- **Healthcare:** The healthcare sector has undergone a revolutionary transformation with the advent of IoT, leveraging advanced sensors and data analytics to enhance the quality of care and accessibility of medical services [38], [39]. IoT has ushered in a new era of possibilities, empowering healthcare professionals with real-time data from wearable devices and smart sensors to facilitate remote patient monitoring, chronic disease management, and medication adherence, resulting in personalized care and timely interventions that reduce hospital visits and improve patient outcomes. This is particularly crucial in countries like India, where there is a shortage of healthcare professionals [40]. Moreover, the development of smart hospitals driven by IoT [41] has revolutionized healthcare facilities, optimizing operations, facilitating telemedicine, and providing easier access to medical expertise. IoT's positive impact extends to supply chain management [42], improving waste management, inventory accuracy, and timely delivery of medical supplies. Through the integration of Electro Cardio Gram (ECG) with IoT, healthcare providers and patients gain real-time insights into heart health, enabling better management of cardiovascular conditions and early detection of potential issues [43], [44]. Nevertheless, it is crucial to address data security, privacy, and interoperability challenges to ensure the successful and secure integration of IoT in healthcare [45].

- **Agriculture:** IoT, often referred to as ”Smart Agriculture” or ”Agriculture 4.0,” has revolutionized the modern agricultural landscape [46]. By harnessing IoT technology, farmers and agricultural professionals are empowered to make data-driven decisions, optimize resource utilization, boost productivity, and tackle challenges associated with sustainable farming practices. A prime example of this is precision farming, where IoT-enabled sensors and devices are strategically deployed across agricultural fields to collect real-time data on crucial parameters like soil moisture, temperature, humidity, and nutrient levels. Utilizing this information, farmers can create precise field maps, enabling tailored irrigation, fertilization, and pest control practices for specific areas, thereby optimizing resource usage and minimizing waste. Additionally, IoT plays a pivotal role in apple orchards, as evident in [47], where predictive models are employed for apple growth and development. Moreover, IoT solutions extend to improving livestock conditions using devices such as smart collars and tags [48]. These innovative technologies monitor the health and behavior of livestock, providing valuable data on activity, location, and vital signs, which in turn aids in early disease detection, optimization of feeding schedules, and overall improvement of animal welfare.

- **Smart homes:** IoT is used in smart homes to connect and control various devices and appliances through the internet [49]. Smart home devices, such as smart speakers, thermostats, lighting systems, security cameras, and home appliances, are equipped with IoT technology, allowing users to remotely monitor and manage them using their smartphones or voice commands. This connectivity and automation improve convenience, energy efficiency, and security in modern households, allowing several devices to communicate and interact with each other and with homeowners. Many household appliances, such as refrigerators, ovens, washing machines, and robotic vacuum cleaners, can be IoT-enabled, offering advanced features, remote monitoring, and improved efficiency. IoT-based security cameras, doorbell cameras, and smart locks provide enhanced security for smart homes. Homeowners can monitor their property in real-time, receive alerts for suspicious activities, and grant access remotely to authorized individuals [50].

- **Autonomous vehicles:** In autonomous vehicles, IoT plays a crucial role in enabling advanced connectivity and intelligence for safe and efficient self-driving operations. IoT sensors and devices integrated into autonomous vehicles collect and process vast amounts of real-time data from various sources, such as cameras, LiDAR, radar, GPS, and vehicle-to-vehicle (V2V) communication. This data is continuously analyzed to perceive the vehicle's surroundings, identify obstacles, detect pedestrians, and make split-second decisions to navigate through traffic and changing road conditions. Additionally, IoT facilitates communication between autonomous vehicles and infrastructure (V2I), such as traffic lights and road signs, optimizing traffic flow and enhancing safety. The seamless integration of IoT in autonomous vehicles ensures a comprehensive and responsive understanding of the driving environment, allowing self-driving cars to operate efficiently, avoid accidents, and provide a safe and convenient transportation experience [51].

- **Remote triggered labs:** IoT has made the science labs accessible to people from where ever they are in the world [52–55]. Through interactive dashboard interfaces, students can now engage in various science experiments virtually, while still adhering to real-world safety precautions that are crucial in a physical lab setting. This virtual experience helps maintain a high level of scientific curiosity and interest among students who face limitations in accessing traditional science labs due to infrastructure constraints.

2.4 Challenges

IoT brings numerous benefits, but it also faces several challenges that need to be addressed for its widespread adoption and successful implementation [56], [57]. Some of the key challenges in IoT include:

- **Security and privacy:** IoT presents a major security challenge due to the vast number of connected devices, significantly expanding the potential attack surface for cybercriminals. Numerous IoT devices lack robust security measures, rendering them susceptible to hacking and unauthorized entry. As a result, breaches in IoT security can lead to severe repercussions, including data theft, privacy infringements, and even physical harm if critical systems are compromised. One of the works in this domain can be referred to in [58], in which it provides an approach for end-to-end encryption, protocol and dashboard security, and a proof of concept de-authentication detector.

- **Scalability:** As the number of connected devices grows, IoT systems must be scalable to accommodate the increased data flow and processing requirements. This challenge includes handling the massive influx of data and ensuring that IoT infrastructure can support the growing number of devices without significant performance degradation.

- **Data management and analytics:** IoT generates enormous amounts of data from various sources. Managing, processing, and analyzing this data in real-time can be complex and resource-intensive. Extracting meaningful insights from such vast datasets is essential for making informed decisions and optimizing IoT applications.

- **Power consumption and battery life:** Many IoT devices operate on battery power, making power consumption a critical concern. Balancing functionality with power efficiency is crucial to extend the battery life of IoT devices and reduce the need for frequent replacements or recharging.

- **Connectivity and network reliability:** Reliable and robust connectivity is essential for IoT devices to communicate effectively. In areas with poor network coverage or unreliable connections, IoT devices may encounter difficulties in transmitting data and receiving updates.

- **Ethical and social Implications:** As IoT collects and processes vast amounts of data, there are ethical considerations regarding data ownership, consent, and potential misuse of personal information. Ensuring transparent and responsible data practices is essential to build trust in IoT technologies. IoT is subject to various regulations and legal frameworks concerning data privacy, security, and compliance. Navigating these legal complexities can be challenging, especially as IoT evolves rapidly.

- **Interoperability:** IoT devices and platforms often come from different manufacturers, leading to compatibility issues and lack of standardized communication protocols. Ensuring seamless interoperability and integration between diverse devices is crucial for IoT's efficient and effective operation.

Addressing these challenges requires collaboration among stakeholders, including governments, businesses, academia, and consumers. Finding effective solutions will pave the way for a more secure, efficient, and interconnected IoT ecosystem.

2.5 Smart water monitoring solutions

Several innovative projects have taken place in the field of smart water monitoring technologies. These projects encompass various approaches, such as developing retrofit devices, utilizing digital meters with wireless data transmission, and implementing ultrasonic-based sensors for pipeline water flow detection.

2.5.1 Digital meters:

Digital meters utilize a variety of sensor technologies, including ultrasonic, Hall effect, electromagnetic, and turbine-based sensors. For instance, in the study referenced as [59], magnetic hole sensors are employed to calculate water consumption. Another example is presented in [60], detailing an ultrasonic water meter that employs a low-power ultrasonic sensor to measure flow within pipelines. In the context of smart home systems, [61] incorporates turbine flow sensors and ZigBee technology to transmit data, enabling real-time consumption monitoring across different points within a building.

While the market does offer industry-standard digital meters based on these principles, they often come with significant costs when implemented on an industrial scale. Additionally, the replacement of existing analog water meters with these digital counterparts could result in resource wastage. For example, electromagnetic-based digital meters start at a price point of approximately Rs. 24,000 for a 2-inch pipe [62]. Ultrasonic-based smart water meters can cost around Rs. 50,000 [63] for industrial usage, and turbine flow meters can range from Rs. 33,000 to Rs. 80,000 [64]. Consequently, the deployment of such digital meters on a larger scale, especially within extensive industrial settings, would entail substantial financial investments.

Alternatively, the adoption of a smart retrofit model proves to be more cost-effective, with an estimated cost of around Rs. 5,000. Furthermore, this approach offers water consumption analysis, and its deployment does not disrupt existing pipeline infrastructure.

2.5.2 Retrofit and IoT-based meters:

In one notable work [65], a wireless middleware solution for smart water metering is introduced. This solution employs Smart Water Metering Middleware (SWaMM), an interoperable wireless IoT middleware built on the Edge computing paradigm. Another remarkable development is the smart water meter solution with energy harvesting, which relies on an induction emitter to detect the position and movement of a metal target on the mechanical water meter's wheel [66]. Furthermore, several endeavours are focused on transforming existing analog utility meters into smart meters [24,67,68]. For instance, in [67], a wireless sensor network-based water management system is proposed. The system uses an analog water meter with a Reed switch, which provides a pulse output that is processed and transmitted to a server using the IEEE 802.15.4 protocol. The data is visualized and monitored through a dedicated web-based system.

In the realm of digit recognition, [68] explores a system to convert utility meter images into numeric data using convolutional neural networks (CNNs) based on two architectures: You Only Look Once (YOLO) [69] and LeNet [70]. A similar approach is taken in [71], where the authors address the gas meter reading problem using a Support Vector Machine for digit recognition, resulting in complex calculations. Meanwhile, [72] proposes a meter image capturing system (MICAPS) that utilizes a K-Nearest Neighbor (KNN) based machine learning algorithm to predict digits from meter images.

One IoT-based smart water-meter system is discussed in [73], combined with a custom smartphone app that offers robust water distribution management in urban areas. The system features real-time updates, logging of complaints, dynamic leak checks at the consumer's end, and hourly consumption monitoring. Lastly, in the pursuit of making analog meters smart, [74] explores digit recognition methods for gas meter readings. They achieve promising results using a CNN based on the Visual Geometry Group (VGG) architecture, achieving an end-to-end performance of 85.71%.

The thesis provides a novel method to utilize the preexisting deployed analog water meters and make them smart by using an IoT-based retrofit model. The data collected from the retrofit model can be used for understanding the usage pattern of water consumption. Note the novelty of the work done in this thesis as compared to the existing literature on smart meter devices. The work in [67] assumes pulse output and does not rely on the images of the meter as we do. The comparison in [68] is done on 100 images, while we have evaluated over a large dataset. The work in [68] and [72] focus only on the conversion of images to digits and do not take into account the use-case specific constraints. Also, in both of them, there is no clarity on the sensing interval or facility to provide light in low light times. In [74], the method implemented was not made to run at the IoT node, which becomes crucial as computationally complex models will not be able to run at the edge.

2.6 Intermittent and continuous water supply system:

There are two major types of water distribution systems: continuous water supply (CWS) and intermittent water supply (IWS) [75]. CWS system is mostly adopted by the developed countries and it involves distributing water from a single point of source to required buildings. The pipelines in this systems are always pressurised and so this system is capable of delivering uninterrupted and consistent access to water resources for residential, commercial, or industrial purposes. This system aims to overcome the limitations of supplying water at specific intervals or during certain hours of the day. In a continuous water supply system, water is available 24/7 without any interruptions, ensuring a steady and reliable flow of water to consumers. Although implementing these systems are financially exhaustive and complex.

On the other hand in case of IWS system, as the name suggest, the water is supplied from one source and get stored at a particular storage unit from where it gets distributed to rest of the desired locations. The water is provided to consumers in cycles or intervals rather than being available continuously. In this system, water is made available for specific durations, often during certain hours of the day, and is then shut off until the next scheduled cycle. Intermittent water supply is commonly found in areas with limited water resources, infrastructure constraints, or inadequate distribution systems and is adopted by many developing countries [76] [77] [78].

There are literature work on high-frequency data collected using smart water meters in CWS [79], [80] and [81]. In [80], the authors try to optimise the different stages of the urban water cycle, from supply to distribution to customer engagement. Efforts are done to disaggregate the various end-user events from the event stream data. The work presented in [79] tries to elucidate the over-engineered design of Australia's existing water distribution system by calculating the water demand rates. Although, efforts to harness high-frequency data for IWS systems remain largely unexplored. Unlike in CWS systems, obtaining precise information regarding water consumption at the user end is a challenging task in IWS. Moreover, in commercial areas like colleges, schools, and institutions, where water usage varies greatly, understanding the water distribution patterns becomes crucial to ensure sufficient water supply when needed and prevent unnecessary wastage. Thus, there is a significant need to delve into utilizing high-frequency data in IWS systems to enhance water management and conservation, which can be targeted by utilising the methods discussed in this thesis.

Chapter 3

ML-Based Smart Retrofit Solution for Digitising Analog Water Meters

This chapter introduces an IoT based economic retrofitting setup for digitising the analog water meters to make them smart. The setup contains a Raspberry-Pi microcontroller and a Pi-camera mounted on top of the analog water meter to take its images. The captured images are then preprocessed to estimate readings using a ML model. The employed ML algorithm is trained on a rich dataset that includes digits from the images of water meters captured by the hardware setup for ten days. The readings are posted on a cloud server in real-time using Raspberry-Pi. High temporal resolution plots of flow rate and volume are generated to derive inferences. The collected data can be used for deriving water consumption patterns and fault detection for efficient water management.

3.1 Introduction

The conventional method of manually reading analog meters to calculate a consumption trend is cumbersome and expensive. This approach is also incapable of effectively managing sustainable water supplies, as it needs accurate monitoring techniques that enable the consumers to know the level of water usage in real-time. The traditional analog water meters have a long life, and removing them for digitization is a waste of resources. Although digital water meters are introduced in recent times and are used in workplaces like government institutions, hospitals, they are very expensive. Also, the digital water meter themselves do not give any inference or do not do any analytical analysis on the water consumption patterns. A smart device for water monitoring can make users reduce their use of water to conserve it.

In this chapter, an ML and IoT-based low-cost retrofitting of existing analog water meters is proposed for making them smart. Specific contribution of this chapter are

1. An IoT-based sensor node is designed and deployed in field to take water meter images, convert them into digits and send it to cloud.

2. A retrofitting approach is used which does not temper the existing meter in any physical form. It is assumed that the meter does not have any pulse output and relies completely on the conversion of images to digits.

3. The designed node can provide real-time data with high temporal resolution (readings every few secs) and is equipped with lighting to enable readings even in nights. This way the meter readings can also give accurate estimates of derived parameters such as flow rate.

4. Simple ML algorithm is used to recognize the digits from the meter image, which requires low processing and is implemented at the node itself.

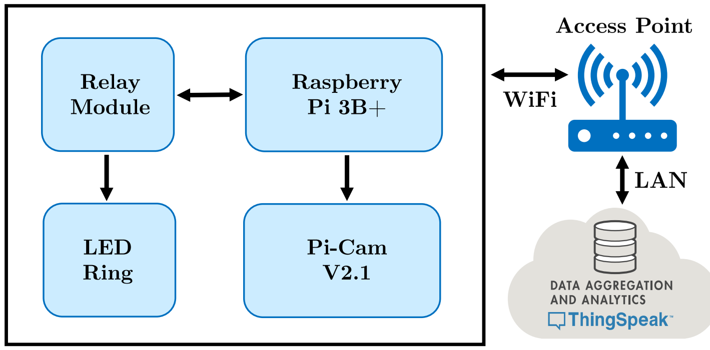
5. The performance of the ML algorithm is further improved by using specific constraints related to the water meters.

6. The proposed approach is evaluated based on the data of over 10,000 images collected from the field deployment of 10 days. The collected data will be made public for further research in future.

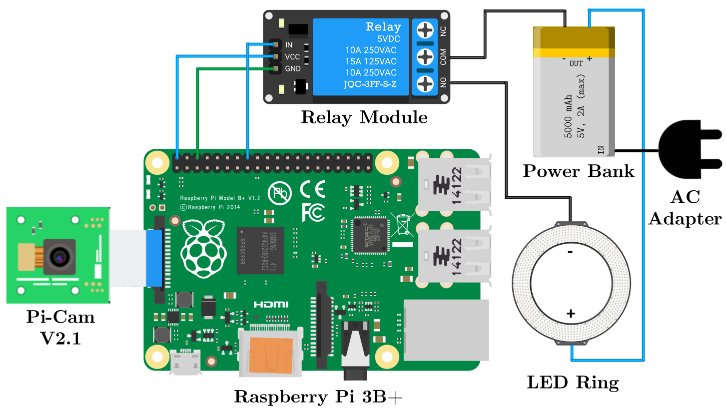
3.2 Hardware description

3.2.1 Circuit description

Figs. 3.1(a) and 3.1(b) shows the block architecture and circuit design of the proposed model. The model comprises a Raspberry-Pi 3B+ microcontroller [82], a Raspberry-Pi V2.1 camera module, an LED ring for illumination and an active-high relay module to control the LED. The hardware is powered by a Li-ion power bank which is connected to AC mains for charging. The power bank enables the device to function without interruption, even if the primary AC power supply is unavailable temporarily. The model is also equipped with a lighting feature, such that the readings can be obtained even at night when there is no ambient light available. The lights are controlled using an active high relay module to operate only when the camera captures an image. This feature helps in extending the battery life of the model. The numeric values are extracted from the images of the water meter dial captured by the camera. The micro-controller executes the ML-based image processing algorithm to detect the reading on the meter. This reading is transmitted in real-time to *ThingSpeak* [83] using an LTE based portable WiFi hotspot [84]. ThingSpeak is a cloud-based IoT platform for aggregating and processing data. The POST method of the HTTP protocol is used to write data on the ThingSpeak server. This setup can sense the reading at a high frequency (as quick as every 5 seconds), even when the flow is at its peak. Such high-frequency information can be used to derive valuable insights about the community's consumption patterns and timely detection of leaks or faults.



(a) Block architecture of the model



(b) Circuit diagram of the model

Figure 3.1 Hardware description of the proposed retrofit model

3.2.2 Retrofit structural view

Fig. 3.2 shows the structure of the developed enclosure. It is a 3D-printed multi-layer structure made up of Poly-lactic Acid (PLA) material which offers protection against various weather conditions in outdoor deployment. It consists of a four-layered stack which separates the various hardware component for comfortable placement. The first layer (bottom most) consists of an LED ring for providing adequate illumination to capture good quality images. In the second layer, the camera module is placed facing the dial of the meter. The camera is configured at a focus of 4 cm with the maximum resolution of 3280×2464 and pixel size of 1.12×1.12 μm. The Raspberry-Pi microcontroller is placed at the third layer. It is interfaced with an active high relay module using the Raspberry-Pi GPIO pins to control LED switching. At the fourth layer (topmost), the power bank is placed. The whole setup is mounted on top of the analog water meter without altering or tempering the analog meter in any sense. It is important to note that the discussed water meters are ISI marked [85], a standards-compliance mark for India's industrial products provided by the Bureau of Indian Standards (BIS). Hence any technological intervention made by physically altering the meters would lead to loss of standardization.

Figure 3.2 3D structure and deployed model

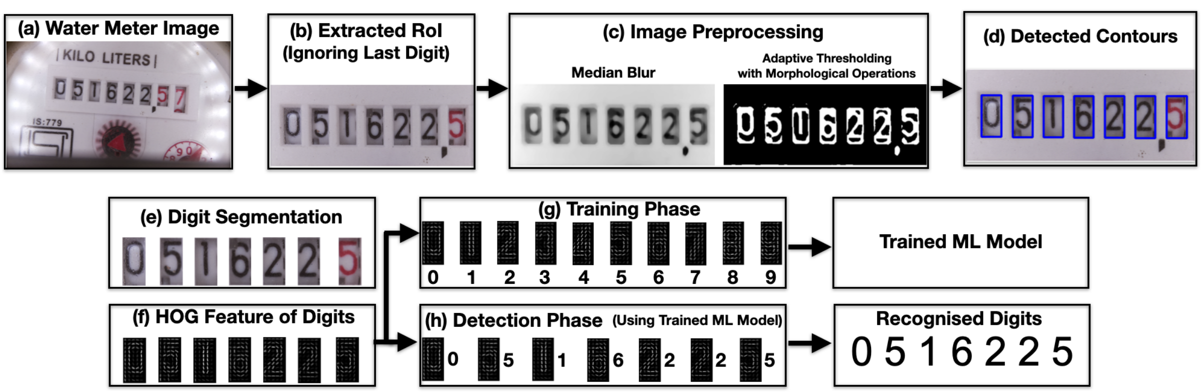


Figure 3.3 Algorithmic pipeline for the proposed method. Notice that last digit of the meter is discarded due to digit image ambiguity. Both the training and detection phases has shown. In training phase, the ML model was trained on digit features and their corresponding labels. In the detection phase, each segmented digit image feature is predicted using the trained ML model. *(Best viewed on screen)*

3.3 Dataset

With the help of the above-described hardware set up on the water meter, two separate datasets were created—one for training the model and the other for analyzing the volume flow and its flow rate. As explained in the hardware section, the camera position is fixed, and the orientation of the images remains the same for all captured images. The coordinates where the meter reading was present were specified and extracted the region of interest (RoI). The resultant RoI image is shown in Fig. 3.3(b). As shown in Fig. 3.3(a), the meter reading is tilted towards the right side. In order to straighten the RoI with respect to specified width and length, perspective transformation was used. The transformation matrix is calculated from the manually selected source and destination coordinates. Using these transform

Table 3.1 Frequency of the individual digit in the training dataset

Digit	0	1	2	3	4	5	6	7	8	9
Frequency	148	110	77	97	45	98	86	85	63	50

coefficients, warp perspective [86] was applied to get the transformed image. This step helps to form nice rectangle-shaped contours.

As the next step, all the digits in the complete reading are segmented into individual digit images and are stored separately as per their numerical value. This makes the dataset suitable for supervised learning. These digit images were further used to train the ML model. The dataset consisted of digits from 0 to 9. These digits were collected from the meter images captured using the Pi camera for 10 days, out of which 7 days data was used to analyze the water flow and the rest for training.

One of the significant advantages of the dataset is that the digits on the water meter have the same font style. Hence there are significantly fewer variations in the orientation while capturing the digit images. As shown in table 3.1, less numbers of individual digit images (ranging from 0 to 9) were needed to cover all possible digit variations. So training the ML model once on this dataset could be used for other meters also that were present on the campus. Below is the estimate of the number of digit images extracted for the ML model training.

Most of the possible variants of the digit images that occurred inside the water meter (RoI) were tried to be covered, i.e., since it is an analog meter, the digits are present on rolling wheels. As the water is flowing, these wheels will rotate for the next digit to come. It can be observed from table 3.1 that the distribution of the frequency of digits is not uniform. The primary reason for that is the dataset was made with the help of a real-time water meter, and thus the rate at which the digits of the meter changed was also not uniform. We also observed ambiguity in digit images (half of the previous digit and half of the incoming digit) cause of rotating-disc type meters. Simultaneously, the images were being captured, and such ambiguous digits were discarded for the dataset collection part.

After creating the digit dataset, water meter images were collected to study the volume flow and its corresponding rate with which the water is flowing. For this purpose, Raspberry Pi captured meter images for around seven days at every minute, which resulted in a total of 10,508 images. These images were then analyzed with the algorithm's help to get the volume flow and its corresponding flow rate.

3.4 Methodology

3.4.1 Training

The digit dataset, explained in the dataset section, was used to train a ML model to classify the given digit image from 0 to 9. As this is a supervised learning problem in ML, a classification-based method was used to train the model. There are several classification algorithms in this particular area. However, a tree-based method was chosen to solve the problem at hand.

The, Random Forest (RF) [87] classifier was used to train on the digit dataset. RF is a supervised learning algorithm that combines multiple decision trees and is trained together with bagging. The bagging method uses the idea that the combination of several weak learning models increases the overall result. To explain the RF model more straightforwardly, it can be said that it combines multiple decision trees and merges the results obtained by them to get a more accurate and stable prediction. The idea behind choosing this particular classifier is to reduce the number of hyperparameters while training the model. It is also one of the most used classifiers for these problems because of its simplicity and diversity.

The training procedure follows these steps:

1. For each digit in the dataset, the digit's Histogram of Oriented Gradients (HOG)-based features [88] were computed, which is a two-dimensional matrix as shown in Fig. 3.3(f). This matrix was flattened and converted to a one-dimensional feature vector with size $n \times 1$. As the image height and width remains the same for all the digit images, we always get a one-dimensional $n \times 1$ feature vector for each digit image. At the end of this step, we have $m \times n$ sized data matrix M where m is the number of samples in the digit dataset.

2. In the second step, the corresponding labels were collected for each digit image in the dataset. The label ranges from 0 to 9. After this step, we have $m \times 1$ label vector.

3. The data matrix M was split into two parts: training and validation with the ratio 80:20, respectively. This is a standard paradigm used in ML methods to test the generalization of the trained model. The RF model is trained on training data only and tested on validation data to find the error rate. The training data was used to train the RF model as a classification supervised learning problem. The RF model's input is the training data matrix M' (training part of M) and the corresponding label vector. The training phase is shown in Fig. 3.3(g). Finally, this trained model was saved for further detection tasks.

3.4.2 Detection

To detect the digits of the water meter a 4-step structure is presented: i) RoI Extraction, ii) Image Preprocessing, iii) Digit Image Segmentation, iv) Digit Recognition and Correction.

3.4.2.1 Region of interest (RoI) extraction

The specific region in the image, which consists of the digits, i.e., the object area, is manually extracted by inputting the RoI's coordinates in the algorithm. We have leveraged the fact that the camera is permanently fixed in one position, and hence setting the coordinates once is sufficient to get the fixed RoI. Notice that the last two digits of the meter reading are decimal places. The last digit was not included in RoI, as this part is the most ambiguous and sometimes even hard for humans to detect the reading. Again, as mentioned in the dataset section, the perspective transformation was used to straighten the RoI.

3.4.2.2 Image pre-processing

Recognizing the location of digits from the RoI is a challenging task. The RoI can be noisy and blurred because of the dusty environment and presence of dew on the water meter. Hence we need to pre-process the image beforehand to find the location of the digits. The following methods were used to pre-process the image: i) grayscale, ii) median blur iii) adaptive thresholding.

Initially, the RGB colored image is converted to a grayscale image to reduce the complexity and computation overhead: from a 3-dimensional pixel value (R, G, B) to a 1-dimensional value. Later on, the grayscaled image is median blurred to smoothen out the edges. Hence, all the high-frequency components (noise) of the image will be removed. Finally, adaptive thresholding followed by morphological operation (dilation) was performed on the blurred images to separate desirable foreground image objects (digits) from the background based on the difference in each region's pixel intensities. After this step, from Fig. 3.3(c), it can be noticed that a closed curve is successfully formed around the digits, which will further help in determining the location of the digits.

3.4.2.3 Digit image segmentation

The preprocessed image was used to find the location of the digits in the RoI. We have used the closed curves formed outside the digits on the water meter to determine the location. In any image, *Contours* are curves or the continuous lines that join all the continuous points, having the same color or intensity, to bound an object's complete boundary in the image. As in our case, the curves are formed around digits, finding them will provide the digit's locations. The contour retrieval mode was set to retrieve only the outer contours, so only the outermost is given, in case we have one contour enclosing another (like concentric circles). The contour approximation method is set to remove all redundant points and compress the contour to save memory. However, there are other contours as well in the preprocessed image that are not formed around the digits and they are discarded based on the contour area. The contours formed on the RoI image is shown in Fig. 3.3(d).

After image-processing and detecting the selected region, each digit present in that contour is segregated and extracted. This was made possible cause the contour stores the coordinates information as well. This process is called image segmentation, in which the image is partitioned into different regions based on a common feature, in our case, the digits in the selected region. As each contour is formed around digits only, the contour coordinates are sorted from left to right based on their position. Fig. 3.3(e) shows the segmented digit images. In the next step, each extracted digit image is detected using the trained ML model.

3.4.2.4 Digit recognition and correction

The last step of the proposed system is the recognition process for the meter-reading digit images obtained in the previous step. The digit images that were segmented with the contour method are now passed to the trained RF model that we created while training. The digit image features were computed using the HOG feature extractor whose output is $n \times 1$. As shown in Fig. 3.3(h), the RF model's input is one digit image feature at a time, and the output is the corresponding digit.

As the digits may be detected wrongly, two digit correction mechanisms as postprocessing are applied to the number collected after combining the predicted digits. These mechanisms are defined as follows:

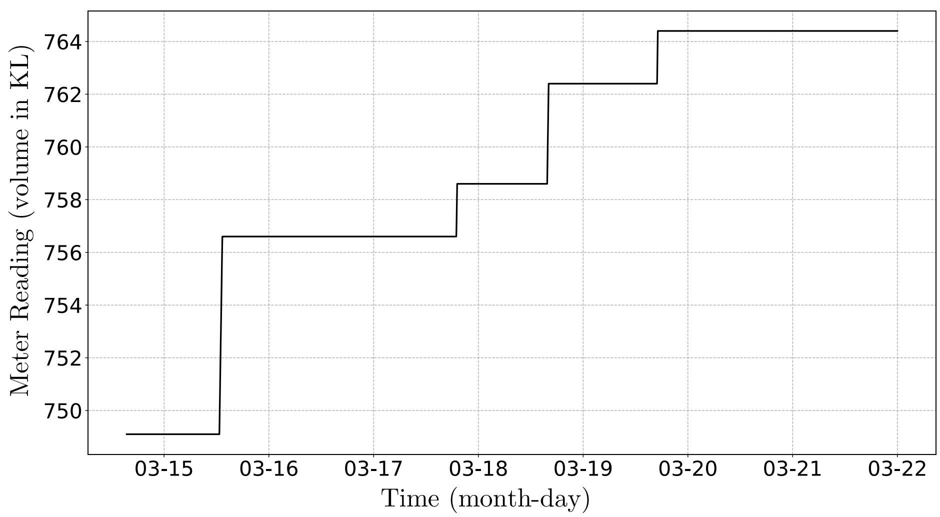
1. As this is continuous real-time chronological data, it is assumed that the current value must be greater than or equal to the previous value. This helps to mitigate the common detection error that occurred while detecting the digits.

2. Another assumption is made that the flow of the water can not increase suddenly by a huge number. Leveraging this fact, the digits detected are adjusted based on the previous flow.

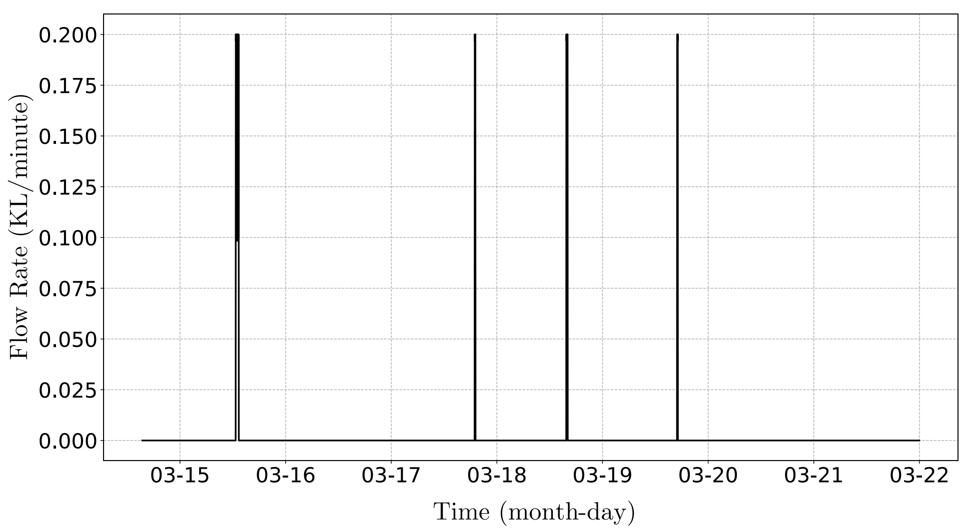
3.5 Results and observations

The developed model was deployed for ten days (Mar 11 - Mar 21, 2021) on a water meter at the pump-house of IIIT-H campus. Water from this pumping station is delivered to a residential area of forty families where it is stored in the overhead tanks. During the initial three days, the captured images were used for training the ML model. The trained model was later used for the next seven days to collect the flow data in real-time. The HOG-based image feature extraction was done using *skimage* [89]. The orientation value was set to 9 with 8 pixels per cell and two cells per block. The RF classifier was implemented using *Scikit Learn* [90], a popular python-based ML library. The *criterion* hyperparameter of the RF classifier, which measures the quality of split while training was set to *entropy*. For all kinds of image pre-processing, e.g., conversion to grayscale, median blurring (with window size 15), and adaptive thresholding, the popular computer vision library *OpenCV* [91] was used.

Figs. 3.4(a) and 3.4(b) show the sensed meter readings and flow rate w.r.t time for the retrofitted water meter, respectively. The time axis is plotted such that every tick marks the beginning of the day with 0000 hrs. Therefore the time elapsed between two consecutive ticks is 24 hrs. Every pumping instance is marked by a rise in the meter reading as well as the flow rate. The meter reading remained constant, and the flow rate remained zero for the duration when the pump was not running and, consequently, no water was flowing. It is observed that the water was pumped four times in the period of observation. The duration for which the water is usually pumped is significantly less when compared to the total duration of observation. Hence, although the pumping duration is few tens of minutes for every instance, it appears as a sudden step in the meter reading or a sharp peak in the flow rate. This is a valid observation because the campus has an automated pumping system. The water is pumped to the residential area's overhead tanks only when the tank's levels dip below a certain threshold. The various instances of water flow along with the duration and volume are shown in table 3.2. It is worth observing from Fig. 3.4(b) (Flow rate plot) that the flow rate does not remain constant for the entire duration of water flow; instead, it usually keeps varying between 0.1 KL/minute and 0.2 KL/minute. It means that there is no deterministic linear relationship between the duration and the volume of water flow. Instead, it is governed by the entirely random demand for water. The observations of table 3.2 validate the same.



(a) Meter reading (volume of water flown) w.r.t time, assuming 0000 hrs as the start of the day on the time axis



(b) Flow rate (KL/minute) w.r.t time, assuming 0000 hrs as the start of the day on time axis

Figure 3.4 Plots of meter reading (volume in KL) and flow rate (KL/minute) for the observation period from 14 Mar 2021, 1525 hrs to 21 Mar 2021, 2359 hrs.

Table 3.2 Water flow time, duration and volume during the period of observation

S.No.	Start Time (dd-mm, hrs)	Stop Time (dd-mm, hrs)	Duration (mins)	Volume (KL)
1	15-03, 1239	15-03, 1321	42	7.5
2	17-03, 1856	17-03, 1908	12	2
3	18-03, 1544	18-03, 1606	22	3.7
4	19-03, 1654	19-03, 1708	14	2.1

Table 3.3 Digit error rate

Digit	0	1	2	3	4	5	6	7	8	9
Error Rate (%)	0.0	0.0	0.0	1.3	0.6	0.4	5.0	5.2	3.2	0.0

The proposed image processing algorithm is tested on the manually annotated test dataset as explained in Section 3.3. The trained RF model achieved a validation accuracy of 97.69%. Even though the RF model is trained at very high accuracy, some digit recognition errors were noticed. Most recognition errors occur for the digits appearing at the tenths (first digit after the decimal) place or units place as they change most frequently. It is reasonable in the rotating-disc type meters. Images captured during the digit change process will lead to an ambiguous situation, where half of the previous digit and half of the incoming digit will be visible in the image. The errors due to rotating-disc issues are efficiently resolved by the digit correction techniques discussed in Section 3.4.2.4.

Next, the proposed solution's performance is evaluated by calculating the following three parameters.

1. **Digit Error Rate (DER):** It is defined as the number of times a particular digit is unrecognised by the total number of times a particular digit appears for recognition. This parameter helps evaluate the ability of the ML model in recognising the digit's image correctly. The obtained DER for all the digits is shown in table 3.3. It can be inferred that the DER is comparatively higher for digits whose shape resembles any other digit's shape.

2. **Value Error Rate (VER):** It is defined as the number of times the complete meter reading or value is incorrectly recognised by the total number of readings recognised. A meter reading is considered incorrectly recognised if it has one or more unrecognised individual digits. VER helps in evaluating the performance of the applied postprocessing techniques. Since the obtained meter values can only increase or remain unchanged, digit errors can be corrected to improve VER. The proposed postprocessing ideas achieve a VER of 4.49%.

3. **Root Mean Squared Error (RMSE):** It is defined as the square-root of the average squared difference between the actual meter reading and the reading obtained from the ML algorithm. This parameter helps in evaluating the severity of the unrecognised meter reading. The RMSE is calculated before and after the postprocessing to check the effectiveness of the technique. The proposed ML algorithm achieved an RMSE of 0.0748 KL before postprocessing and 0.0361 KL

after postprocessing. It can be inferred that the postprocessing technique has improved the RMSE. Moreover, it is observed that the RMSE is significantly low when seen in the light of the discussed application.

The above results and observations show that DER does not directly translate into high VER. Moreover, the achieved VER does not lead to high errors in the analysis as most of the errors occur in at least significant digits.

Chapter 4

Improving Smart Water Meters using DL-based Algorithm

This chapter discusses a DL-based algorithm which is used for improving the performance of digit detection from IoT-based smart retrofit model for analog water meters. The DL algorithm is trained on a rich dataset of over 160,000 images collected from six water nodes deployed at locations with different environmental conditions. A detailed comparison between the proposed DL and ML algorithm is made based on detection accuracy, feature analysis, error analysis, and computational complexity analysis. It is observed that compared to the ML model, the proposed DL model maintained a higher detection accuracy and is more generalized in terms of feature extraction, which makes the algorithm robust.

4.1 Introduction

The traditional analog water meter has a simple structure, minimal power dissipation, high durability, and high dependability. However, it still requires human effort to record meter readings manually, which is inconvenient. In order to counter this problem, there is a need for an automated system of data capturing, which can send the real-time detected meter readings to the cloud server. By analyzing the data collected, smart water meters can aid in understanding water consumption patterns and leakage detection for efficient water management. Thus it becomes essential to have a robust algorithm for meter reading detection.

In this chapter, specific contributions are

1. A DL-based algorithm is proposed, which performs more accurately than the ML algorithm described in [24]. The proposed DL algorithm is based on CNN with transfer learning.

2. A rich dataset is collected using images captured from six IoT-based smart water meters deployed in locations having different environmental conditions. For training, approximately 160,000 images have been used, which have been collected over 20 days. The testing data contains approximately 80,000 images collected over 10 days in real-time.

3. A performance comparison against the proposed DL-based algorithm is carried out with the ML-based algorithm proposed in [24]. It is shown that the DL-based algorithm performs better than

the ML-based algorithm, even without the need for post-processing constraints to correct the false detections. Moreover, the proposed DL algorithm is computationally light enough to be deployed on our retrofit model (on edge).

4. To understand the performance improvement of the proposed DL model over the ML-based model as given in [24], the extracted features from both the models are analyzed to infer what models are observing during detection.

Hardware Description: The hardware setup is the same as described in Chapter 3. The components and overall design used in retrofit have been kept the same, and improvements are made on the software end.

4.2 Dataset

This section is divided into three parts. The first part demonstrates the variability in the meter images based on the environmental factors and the need of collecting data from nodes deployed in different locations. The second part presents the dataset collected to train both (ML [24] and the proposed DL) models. Finally, the last subsection will discuss the dataset on which the proposed DL and ML model was tested in real time.

4.2.1 Pre-dataset collection

In a dataset was created using only one deployed water meter and the ML model showed the best digit detection accuracy on that deployed node. However, it failed to maintain similar detection accuracy on the other water meters of the same model deployed at different locations. This is due to environmental factors like sunlight, moisture, etc. For instance, one of our water meters was present inside a small warehouse, where it was exposed to minimal sunlight and external environmental factors. In contrast, one water meter was present in the rooftop of a building, where it was exposed to direct sunlight and other external factors. These factors will influence the illumination in the retrofit model while capturing the image. To demonstrate this, histogram of captured images' pixels is plotted for all the six nodes in Fig. 2. For this, 50 images were randomly chosen from each node, and their RoI was extracted on which the average histogram of pixels was plotted. The observations were intuitive as the average histogram of pixels differed for all the nodes, as shown in Fig. 2, which helped us understand how the environmental factor can influence the data collected from the IoT devices. Based on the above observations, data was collected from all the six water meters for a month at the rate of 1 image per minute.

4.2.2 Training dataset

The data collected in first twenty days from all six nodes (approximately 160,000 images) was used to create the dataset. The captured images were transformed by manually inputting the selected coordi-

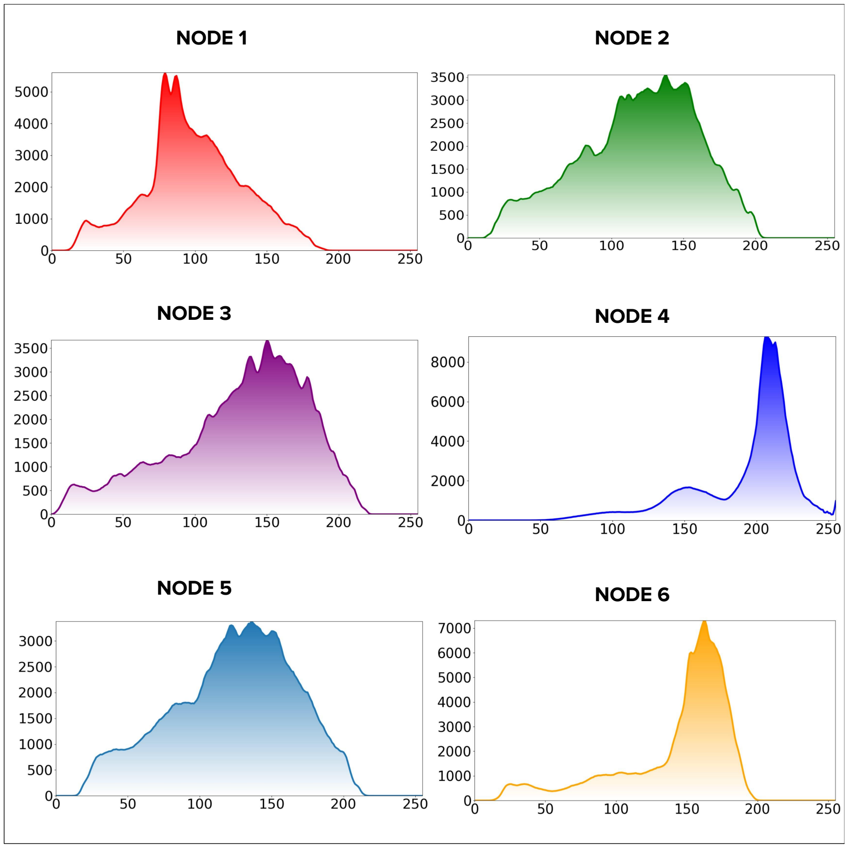


Figure 4.1 Histograms of pixels for the images of respective nodes are displayed here. About 50 images were chosen randomly from each node, and their RoI was extracted on which the average histogram of pixels was plotted. The x-axis represents the pixel intensity values in the plots, and the y-axis represents the number of pixels having that particular intensity value. (Best viewed in color)

nates, width, and height to get the transformation matrix. This matrix was used to apply warp perspective to get the RoI. After RoI detection, the digits were extracted by segmentation. These digits range from 0 to 9 and are saved in different folders to make a labeled dataset for supervised learning.

An ideal dataset should have all data points which are independent and identically distributed to decrease bias in the dataset. It was noticed that only the right-most digit (see water meter image in Fig. 4.2) was used to change for the time-series data, and the other digits moved seldomly due to the low water flow rate. This resulted in duplication of digits in the dataset. To remove duplicate images, images were first translated into tensors made by combining the respective RGB pixels and were resized to a pixel size of 50 to keep the comparison process computationally less expensive. These resized tensors of the two images were compared using mean square error (MSE). The lower the MSE, closer (similar) the images are to each other. Ideally, two images with MSE equal to 0 should be duplicates; however, due to resizing the images to 50 pixels, there is a slight variation in the original image. Therefore, we

Table 4.1 Frequency of the individual digit in the training dataset after preprocessing

Digit	0	1	2	3	4	5	6	7	8	9	Total
Node 1	274	75	27	42	142	195	339	57	33	103	1287
Node 2	254	505	208	329	386	658	451	382	436	481	4090
Node 3	497	551	588	226	434	743	472	403	524	726	5164
Node 4	1903	1336	1491	1871	1325	570	1002	1845	1035	703	13081
Node 5	103	134	125	76	150	130	103	144	134	131	1230
Node 6	151	31	116	154	125	156	131	103	248	215	1430
Total	3182	2632	2555	2698	2562	2452	2498	2934	2410	2359	26282

set a maximum MSE threshold of 20 (chosen based on trial and error) to find and remove the duplicate images.

Table 4.1 gives a statistical review of the amount of individual digit data collected from all the nodes. It can be observed that the frequency of the digits collected is not the same for all the nodes. The reason for this is that water flow through different meters differs depending on the water usage in corresponding regions. Because of this, more water flows through some meters than others. In some meters, water may rarely flow resulting in large number of duplicate images, which are getting removed. Therefore, we got different numbers of digit images from the six nodes.

4.2.3 Testing data

Both the models (ML model and the proposed DL model) were tested in real-time using the data collected in the last ten days (approximately 80,000 images in total) from all the six nodes. Using this data, both the models were compared based on various factors which will be discussed in the results section.

4.3 Methodology of proposed DL algorithm

4.3.1 Training

The problem at hand is defined as follows: Given a water meter image, extract the digit images and recognize the digits present. To solve the problem, the training dataset defined in the section 4.2 is used. As this is a classification-based supervised learning problem, DL based CNN algorithm is proposed to train and classify the digit images. In recent years, CNN-based algorithms have been extensively used in object detection and outperform image-processing-based feature extracted ML models. In our case of water meter digit recognition, usage of CNN based model is even more favorable due to the following conditions:

1. The trained CNN model extracts a rich feature set: scale, illumination, rotation, and translation invariant. As the water meter digit images contain all these variations in real-time, the CNN

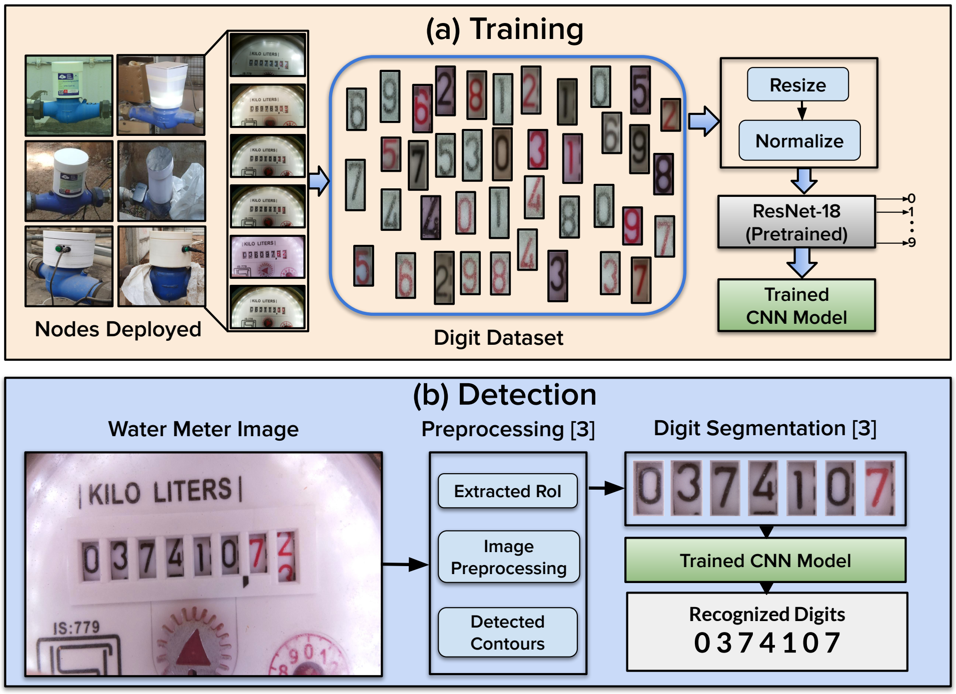


Figure 4.2 Algorithmic pipeline for the DL methodology proposed. The dataset collected from different nodes mentioned in section 4.2 is used to train the ResNet-18 model (last layer having 10 nodes) using transfer learning mechanism. At the detection phase, preprocessing used in was used to segment the digits from water meter image. Finally the digits are recognized using the trained CNN model. (Best viewed in color)

features-based trained model is generalized and more robust than image-processing based extracted features.

2. DL models like CNN can leverage the capability of transfer learning which allows the knowledge transferability of pretrained models on a huge dataset for similar tasks. Using this mechanism in the case of water meter images helps faster and more accurate training.

The training procedure of the CNN model is as follows:

CNN model description: Although there are many CNN models available for the object detection task, our use case required a CNN model with less computational complexity and yet is accurate enough to perform the digit recognition task. The main reason for being computational complexity aware model is to perform the inference on the hardware node itself. Following [92], a popular CNN model ResNet-18 [93] was a suitable choice in terms of computational complexity and accuracy.

As evident from the name, ResNet-18 is 18 layers deep network, and the model overall contains almost 11 million trainable parameters. Training the Resnet-18 model, where the weights are randomly initialized, directly on the digit dataset is an extremely challenging task. To solve this problem, the transfer learning mechanism is used. [93] has trained ResNet-18 model on a huge dataset named ImageNet [94] which contains around 14 million images ranging 20000 categories. We used this pretrained model to further train on the digit dataset as mentioned in section 4.2.2. The final fully connected layer of the pretrained ResNet-18 model was replaced by ten neurons to match the number of outputs of digit recognition.

Preprocessing: As the ResNet-18 model input is of fixed size, all the digit images are first resized to a fixed resolution of 224×224 to match the compatibility of the pretrained ResNet-18 model. Secondly, the standard normalization (zero mean and unit standard deviation) is applied to the input image. This step ensures that all the inputs to the CNN model follow a similar data distribution and helps converge faster while training the network.

Training: The dataset was split into training and validation sets with 90% and 10% randomly selected data points, respectively. This step helps generalize the model as it is only trained on the training dataset and tested on the validation dataset. While training the model, instead of only training the last fully connected layer, all the 18 layers of the ResNet-18 model were finetuned, which helped achieve better accuracy. The model was trained for 5 epochs where it hit the convergence. The choice of *optimizer* was *Adam* with a learning rate of 0.001 scheduled using a linear scheduler having *gamma* set to 0.1. For all the DL implementation, popular python-based framework *Pytorch* [95] was used.

4.3.2 Detection

To detect given water meter image, detection methods till segmentation (RoI extraction, image preprocessing, and digit image segmentation) defined in the detection section of is used. After obtaining the segmented digit images, the preprocessing step similar to training (image normalization and resize) was applied to the digit images. Finally, each of the digit image is passed through the trained CNN model to recognize the digit present between 0 to 9.

4.4 Results

This section is divided into four subsections. First subsection compares the detection accuracy of both the ML model in and the proposed DL model (CNN with transfer learning) on the testing data as mentioned in dataset section 4.2. Note that ML model in is the RF model with postprocessing techniques. Second subsection comprises of analysis of features extracted by the models. The third subsection does a detailed error analysis of both the models. Finally, the last subsection presents the comparison of the models based on computation complexity.

4.4.1 Detection accuracy

Table 4.2 presents the detection accuracy of the trained models on the testing data. The proposed DL model has more than 99% detection accuracy for all the nodes, whereas the detection accuracy of the ML model fluctuates from 91% to 97% depending on the node. It is essential to notice that the ML model gets this accuracy after some post-processing constraints as described in. However, the proposed

DL model does not require any constraints to maintain its high accuracy of more than 99%. This, in turn, shows that the proposed DL model is more generalized in terms of feature extraction and detection. Thus, it can be used with multiple nodes deployed nodes in different environmental conditions. Figs. 4(a) and 4(b) show the detected meter readings by both the RF model (with constraints) and proposed DL model of nodes 1 and 2, respectively. The time axis is set up so that every tick corresponds to the start of the day at 0000 hrs. As a result, the interval between two consecutive ticks is 24 hours. There is an increase in the meter reading records during every pumping event. For the period that the pump was not on the meter reading stayed steady. It can be observed from the plots that the ML model has more false detections than the proposed DL model. From both plots, the zoomed parts highlight the wrong detections by the ML model. The red spikes in the plots denote the wrong detections. The error analysis is done later in this section to show where the ML model fails to detect digits.

The accuracy of proposed DL model is above 99%; at very rare instance like in Fig. 4.3(a) at 0300 hrs on 9th January an error was observed. Here the CNN algorithm detected a tens digit 9 as 0. However, it is worth observing that the proposed DL model performs better than the ML model (which requires post-processing) with consistently higher accuracy on different nodes. The other four nodes also had a similar types of graphs but not shown here for brevity.

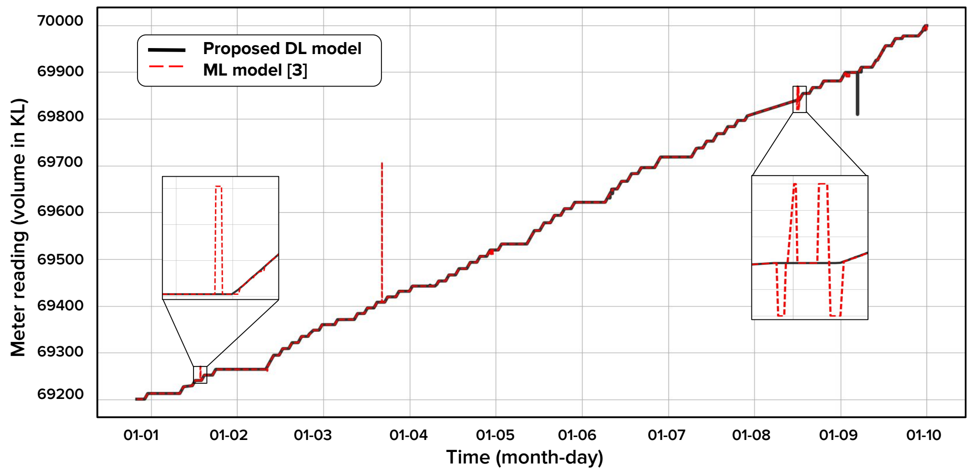
4.4.2 Feature analysis

A comparison has been made between the features extracted by the ML model (RF-based model) and the proposed DL model (CNN with transfer learning). The RF model trains on HOG-based features. To analyze the HOG extracted features, the method proposed by [96] named HOGgle is used. The HOG extracted features are inverted into the corresponding natural image to understand at a human level using this method. This also helps to understand what exact features a HOG-based learning algorithm (RF model in this case) has been trained on. The incorrectly detected digits by the ML model, trained on HOG features, have been visualized. On the other hand, to analyze the CNN extracted features, the method proposed by [97] is used. This method uses features extracted by a trained CNN model to reconstruct their corresponding original image. It has been claimed that features extracted by a highly accurate CNN model preserve photographically accurate information about the image and can be used to reconstruct their original image. In our case, the CNN model's first, third, and fifth convolution layer outputs are used to reconstruct the digit image from a randomly generated noisy image.

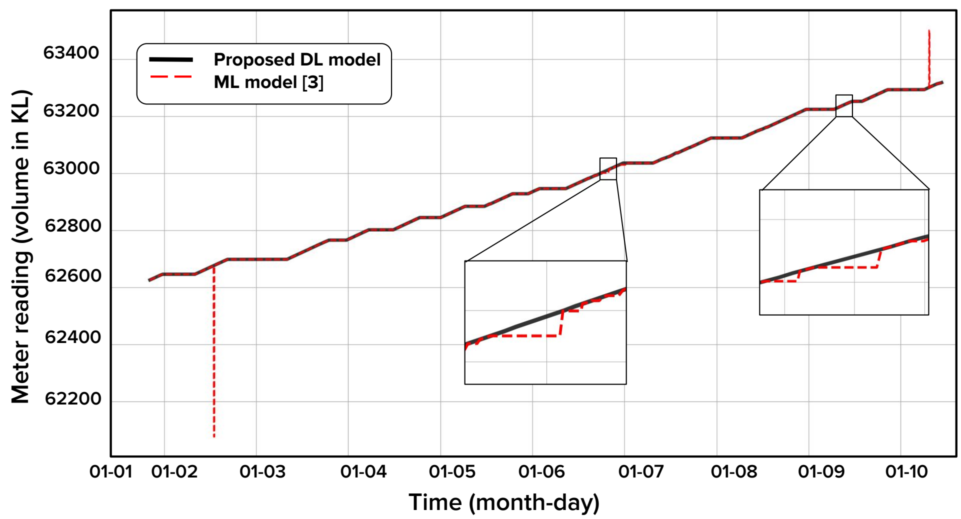
In Fig. 4.4, it can be observed what exactly leads the ML model trained on HOG features to make mistakes while detecting certain digits. For example, in the case of Fig. 4.4(a), it can be seen that the

Table 4.2 Accuracy in percentage for both models

	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6
ML model	96.2	93.4	94.5	91.2	95.4	97.2
Proposed DL Model	99.2	99.5	99.4	99.3	99.6	99.2



(a) Node 1



(b) Node 2

Figure 4.3 Meter reading (volume of water flown) w.r.t time, assuming 0000 hrs as the start of the day on time axis

ML model is misclassifying the digit 0 as 9. It was also observed that most errors occurred for the digits where some part of other digits had a shape influence. In contrast, the CNN model was able to detect it correctly. In order to understand the reason behind it, we have analyzed the digit HOG features by inverting them following the HOGgle method. It can be seen that the inverted HOG image resembles the shape of digit 9. While there were certain cases when the CNN model misclassified the digit images, the analysis of CNN extracted features using [97] was performed to understand the reason behind it. For example, in the case of Fig. 4.4(c), the digit encountered was a particular case where half of the previous digit and half of the upcoming digit ('half digit') were present due to the rotating-disc type meters. In this case, both the ML and DL model misclassified the digit image. After analysis of both features (HOG-based and CNN extracted) for this particular image, it can be observed that both resemble the

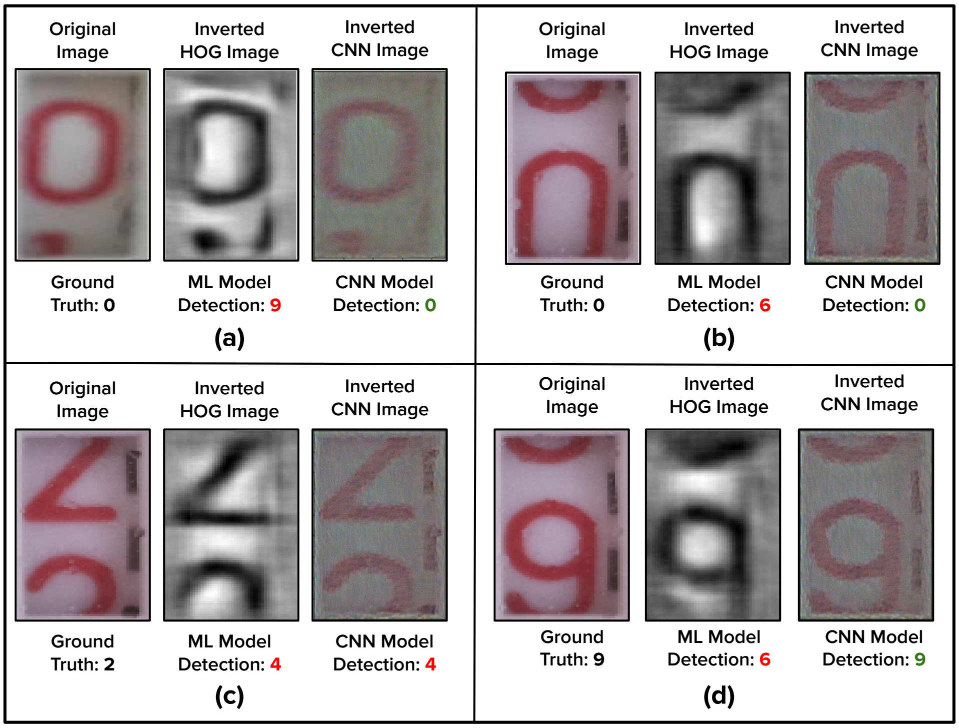


Figure 4.4 Analysis of features extracted between HOG and CNN model. (Best viewed in color)

shape of digit 4. This was one of the primary motivations to discard the last digit of the water meter image (i.e., the least significant digit) at the detection stage. The frequency of encountering such kind of 'half digits' at the last position increases significantly compared to the rest of the digit's position and may lead to error-prone detection.

For all other digit locations, the CNN model's overall error rate was significantly less even when the water meter node's location differed. On the other hand, in the case of the ML model, the same cannot be said. The primary reason behind this can be inferred by the difference in richness of features extracted between both the methods. It can also be claimed that features extracted by the CNN model are more generalized than the HoG-based features.

4.4.3 Error rate analysis

In the error analysis, we use the same performance parameters as in , i.e., digit error rate (DER), value error rate (VER), and root mean square error (RMSE). Table 4.3 shows the DER for all nodes on the digits that were detected using both the models. It can be observed that the proposed DL model has a very low DER values, whereas, ML model has a higher DER values for most of the digits. This again proves that the proposed DL model has more generalized features than the HOG features-based RF model. Along with DER, we also found the VER and RMSE for both the models given in table 4.4. The maximum VER for RF model came out as 8.61% and for proposed DL model 0.62%. While the

Table 4.3 DER (in percentage) of both the algorithms on the nodes

Digit		0	1	2	3	4	5	6	7	8	9
Node 1	ML model	6.43	3.05	0.0	5.0	0.2	3.07	0.2	0.5	6.1	1.05
	Proposed DL model	0.0	0.0	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.1
Node 2	ML model	0.0	0.78	0.0	1.41	0.19	0.65	0.15	1.47	0.5	0.0
	Proposed DL model	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0
Node 3	ML model	13.1	4.85	0.5	2.43	1.03	6.93	0.24	1.87	17.28	1.65
	Proposed DL model	0.4	0.0	0.0	1.221	0.0	0.0	0.12	0.0	0.0	0.0
Node 4	ML model	7.69	6.66	1.73	8.22	3.46	3.33	1.73	12.21	8.82	3.09
	Proposed DL model	0.0	0.0	0.86	0.0	0.0	0.0	0.0	0.0	0.0	1.13
Node 5	ML model	11.90	0.2	0.0	10.25	6.55	7.79	0.18	2.16	9.19	0.82
	Proposed DL model	0.0	0.0	0.0	2.5	0.0	0.0	0.09	0.0	0.0	0.0
Node 6	ML model	0.8	0.0	1.85	0.15	2.65	0.38	0.31	0.0	0.0	0.0
	Proposed DL model	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0

Table 4.4 VER and RMSE of both the algorithms on the nodes

Node		ML model	Proposed DL model
Node 1	VER (%)	5.53	0.44
	RMSE (KL)	0.157	0.06
Node 2	VER (%)	1.81	0.2
	RMSE (KL)	0.038	0.004
Node 3	VER (%)	7.83	0.6
	RMSE (KL)	0.174	0.027
Node 4	VER (%)	8.61	0.62
	RMSE (KL)	0.319	0.024
Node 5	VER (%)	5.61	0.4
	RMSE (KL)	0.351	0.016
Node 6	VER (%)	1.81	0.4
	RMSE (KL)	0.161	0.019

maximum RMSE for the ML model came out as 0.351 KL, and for the proposed DL model was 0.06 KL.

4.4.4 Complexity analysis

Table 4.5 shows the computational complexity analysis for both the models in terms of three main parameters of computational complexity: execution memory (RAM), storage memory (ROM), and execution time recorded during the run-time. It is observed that memory and space consumed by both models do not have much of a difference. However, the execution time is approximately 3.5 seconds more in the case of the proposed DL model. As the nodes take water meter images with a frequency of one image per minute, the slight increase in the execution time does not hinder the algorithm's functioning in any way.

Table 4.5 Complexity analysis

	Memory (MB)	Space (MB)	Execution time (s)
ML model	250	33.2	~ 1
Proposed DL model	280	44.8	~ 4.5

Chapter 5

Behavioral Analysis: Use Case of Two Buildings in IIIT-Hyderabad

This chapter presents an analysis of water supply behaviour on the IIIT-Hyderabad campus, focusing on two distinct regions: student hostels and faculty/staff quarters. The analysis was done for data collected in the monsoon semester of 2022, from August to November, at every minute interval. The investigation delves into the impact of water supply patterns on a monthly and weekly basis. Notably, it highlights how each month, with its unique characteristics such as holidays, exams, and class schedules, influences the water supply in both regions. One key difference between the two regions is that students reside in one, leading to significant variations in water usage based on the number of holidays. Conversely, the other region accommodates families, resulting in a consistent water requirement regardless of college holidays. The findings from this analysis are crucial for understanding water distribution patterns, particularly within IWS systems, with the ultimate goal of enhancing the efficiency and robustness of water distribution. By thoroughly examining the water supply behaviour in an educational campus and considering various factors that influence it, this work contributes to a better understanding of water management on campus.

5.1 Deployment setup

Fig. 5.1 illustrates the meter deployment setup across the campus, which comprises two distinct regions for analysis: the student hostel block and the faculty/staff block. Each region possesses its own unique water storage and consumption patterns. The map provides an overview of the buildings in each block and their corresponding water supply arrangements. Both blocks have their own borewell and sump to store water, which is subsequently distributed to an Overhead Tank (OHT) for further consumption. For drinking purposes, the Government supplies Manjeera water. The map also highlights different types of water meters installed on the supply lines. In the student hostel block, only smart retrofit meters are employed. On the other hand, the faculty/staff block uses two types of meters: digital meters (referred to as Shenitech meters) and smart retrofit meters. Within this region, three meters are present: two digital meters and one smart retrofit meter. One digital and one analog meter are installed in series on the same pipeline to measure the drinking water supply from the sump to the OHT. This arrangement allows us to verify the functionality of IoT-based smart retrofit meters in comparison to industry-based digital meters, and the results of this comparison are discussed in the following sections. At the same time, an individual digital meter is utilized to measure the tap water supply. Overall, this meter deployment setup provides valuable insights into water consumption patterns in different campus regions and facilitates the evaluation of various metering technologies for efficient water management. The naming convention of nodes and their respective locations are as follows:

- Node-HB1 and Node-HB2 are present in the hostel block of building A. The water measured by these meters is used for both tap and drinking (after refinement) purposes.

- Node-HB3 is present in building B of the hostel block. The water measured by this meter is used for both tap and drinking (after refinement) purposes.

- Node-FD1 and Node-FD2 both measures drinking water of building E, but the former is the smart retrofit meter and the latter is the digital meter, present in the faculty and staff block.

- Node-FT3 measures tap water of building E in the faculty and staff block.

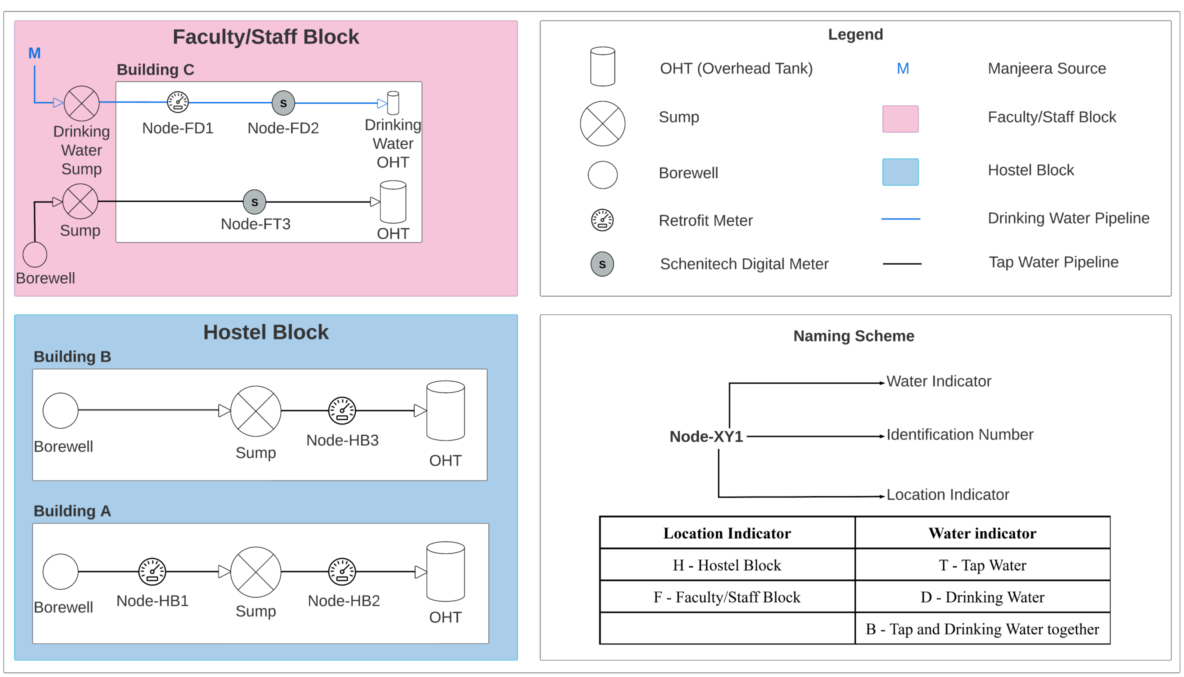


Figure 5.1 The region in blue colour denotes Hostel Blocks, containing two student hostels, and the region shown in pink colour is for the residential block which shows water distribution for the building in the region, namely Anand Nivas

5.2 Data processing

The section is divided into three subsections. The first subsection will be an overview of data collection, following which in the second subsection, challenges and errors related to collected data will be discussed and finally, in the third subsection, corrective measures taken for the errors are discussed.

5.2.1 Data collection

There were some challenges related to the data that were collected during the monsoon semester from August 2022 to November 2022 (at every one minute), such as some missing days of data and some delays in receiving the data. There were instances when the deployed node had to be brought back to the lab for maintenance purposes, due to which there were instances when no data were collected for a few days. Some days there were instances when due to heavy rain, the circuit board powering the device got short-circuited, so the repair had to be done, leading to data loss for that period. There were instances when some delays occurred due to poor network connectivity and power outage in sending the data. Refer to the table 5.1 to get an overview of how many such data losses occurred. The table shows the number of data points missing in days for each node in the respective regions for every month. Note that the days mentioned in the table are not consecutive days but rather discrete, and an accumulation of them is given. In the table, it can be seen that most of the missing days are in the month of October; the main reason behind this is that many holidays occur and so students go home, thus making it difficult to maintain the system. However, the low occupancy of students resulted in low consumption of water. During the time of holidays, classes don't take place, and water consumed goes even down; thus, not collecting data at high frequency won't affect the analysis much. Some errors that occurred during the data collection are explained in the next sections.

5.2.2 Errors analysis

Fig. 5.2 shows the issues that occurred while capturing the images of meters. The shown four images are some major issues that took place related to environmental factors, such as the deposition of dew inside the glass of the meter, which usually happened when heavy rain occurred, smudge/scratches on the meter, instances when some janitors mistakenly struck the device while cleaning the area and thus

Region	Node	August	September	October	November	Total
Hostel Region	Node-HB1	1	3	7	8	19
	Node-HB2	2	2	5	6	15
	Node-HB3	11	2	12	N/A	25
Faculty/Staff Region	Node-FD1	8	0	4	0	12

Table 5.1 Table showing missing data points of each node in respective regions (in days).

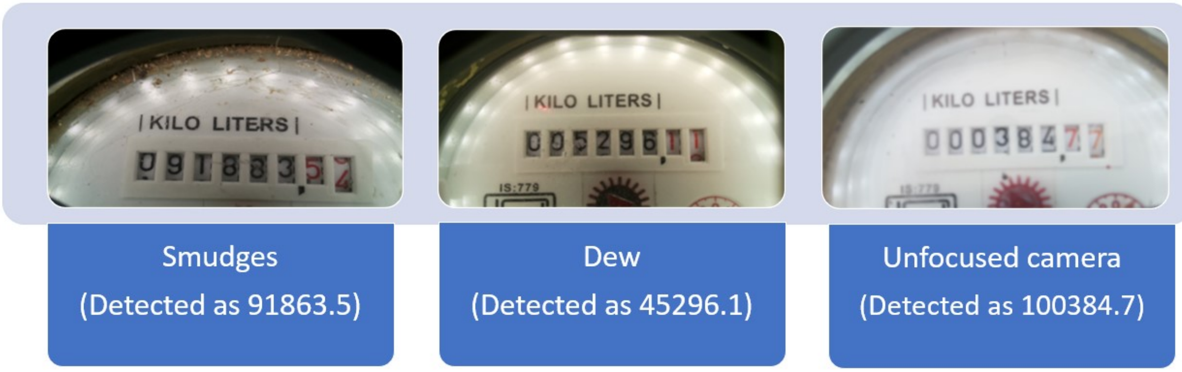


Figure 5.2 Issues Related to collected images.

causing issue with the orientation or resolution of the camera and small insects entering the device causing errors in the digit detection. These issues lead to detection errors that produce abnormally high or low output values. The wrongly detected value persists as the output until the meter is cleaned, and the repetitive nature of the error causes a decrease in detection accuracy. It's worth noting that these errors result in sudden, highly significant changes in a very short period, which is not possible based on the pipe's capacity. Taking these logical conclusions into account, a refinement algorithm is proposed in the following section to enhance the overall data quality.

5.2.3 Data correction

To tackle the issues with the detection, a refinement algorithm was proposed, which is based on the concept of Hamming distance. Where Hamming distance is a metric used to measure the difference between two strings of equal length. It specifically calculates the number of positions at which the corresponding symbols (characters or bits) in the two strings are different, in our case it was a string of digits. In this algorithm, a range was defined beyond which the values of the meter cannot go in the given time frame and based on that, the detected values will be checked if it is within that range or not. The range of the values can be dynamically changed based on the time difference between the two detected values. So the range will be $R = 10T_D$. In the equation, the number 10 represents the largest range for the changing values of the meter's digit in one minute (decided on the basis of the maximum possible rate of water supported by the motor in the specified time interval). T_D stands for the time difference between the capturing interval of images. If there's a delay in getting the image, the range will be expanded based on the time difference between the initial value's timestamp and the current received value. So, the current detected value should be between the previous detected value and the previous detected value plus R (the range calculated in the equation). We'll compare the current detected value with the values within this range by checking their individual digits, similar to how Hamming distance checks. We'll count how many digits are different between the current detected value and each value

T1 (Prev)		2	3	3	2	5	5
T2 (Detected)		2	3	3	1	5	9
Corrected		2	3	3	2	5	9

Table 5.2 Illustration of refinement algorithm.

within the range. The value with the fewest different digits will be considered the correct value. Ideally, this difference count will be 0, which means the current detected value is within the range. Refer to table 5.2 to understand from an illustration.

In the illustration, the previous detection occurred at time T_1. At the time T_2, the current detection took place, where one digit was wrongly detected as 1 instead of 2, while the other digits were correctly detected. The Hamming distance between the values of these two time periods is 2, as indicated by the two different digits highlighted in orange. The first highlighted digit is the wrongly detected value of 2 as 1, and the second highlighted digit is the difference between the detected digit 9 and the corresponding digit in the first value of the range i.e. 5. Therefore, in this illustration, the Hamming distance is 2. As described above, the range is set to as $(P_D,\ P_D + 10T_D)$ where P_D is the previously detected value, so the algorithm will iterate from 233255 to 233265. Thus when it reaches 233259, it will have the least Hamming of 1, where only digits 1 and 2 will mismatch so we will consider this least Hamming digit value. Hence our final value of the meter will be 23325.9 kL. Note that the least value at which the least Hamming distance has come will be considered to avoid any discrepancies with other digits having similar Hamming distance. This method provided us with more robust constraints and refinement methods while limiting the error range to 10 in the best-case scenario at a one-minute data interval.

Using this algorithm rest of the analysis was done. To get a statistical understanding of how the algorithm is working and how much improvement it has provided, refer to the table 5.3, which shows the RMSE (Root Mean Squared Error) value and MAE (Maximum Absolute Error) value of the detected DL-based algorithm and after refinement. To generate this table, approximately 4K data points were manually annotated. The majority of the detection error occurred due to reasons stated in the previous subsection. It's important to note that the original DL-based model had an accuracy of approximately 99%, as mentioned in [30]. However, in this specific case, the node was well maintained for a period of 30 days and during this period little or no rain occurred. The current scenario involved capturing images for the entire semester, which lasted around four months. These images included defects mentioned earlier and were influenced by varying climatic conditions, such as rain. The presence of dust particles and smudges significantly affected the image quality, especially after heavy rain, as insects would often enter the device during that time. The detection errors would persist until the dust was cleared, but they were repetitive in nature. For instance, when dust particles settled on the Most Significant Digit (MSD)

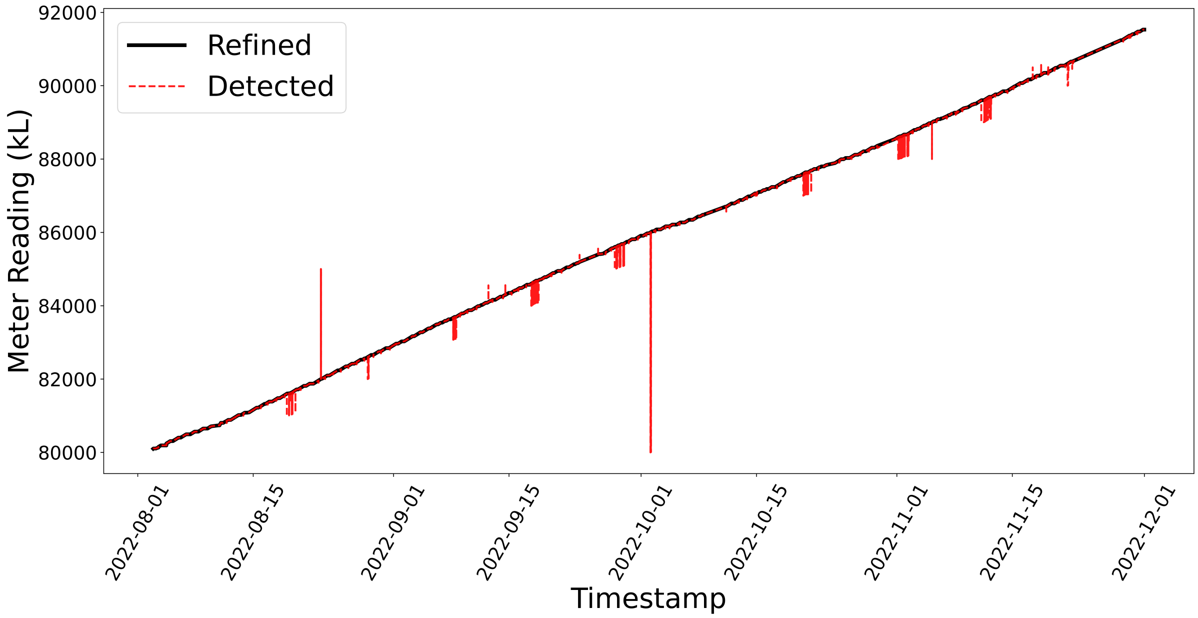


Figure 5.3 Graph of detected values and refined values for node Node-HB2, plotted over four months of data. On the X-axis timestamp is given, and on the Y-axis, meter readings in kL (Kilo Litres) is mentioned. (Best viewed in coloured)

that rotated slowly, the same detection error would persist for a prolonged period. However, these errors were effectively addressed by the refinement algorithm.

	Detected	Refined
Root Mean Square Error	8.60	0.24
Mean Absolute Error	0.67	0.02
Accuracy	97.01	97.37

Table 5.3 Table showing the improvement done by the refinement algorithm.

Fig. 5.3 shows a graph for the node Node-HB2 (refer Fig. 5.1), depicting detected and refined values over the four months of data where 3,362 data points had to be corrected by the refinement algorithm.

5.3 Result and analysis

This section is divided into four subsections. The first subsection compares the digital water meter (Shenitech meter) with smart analog water meters. The second subsection focuses on time series data

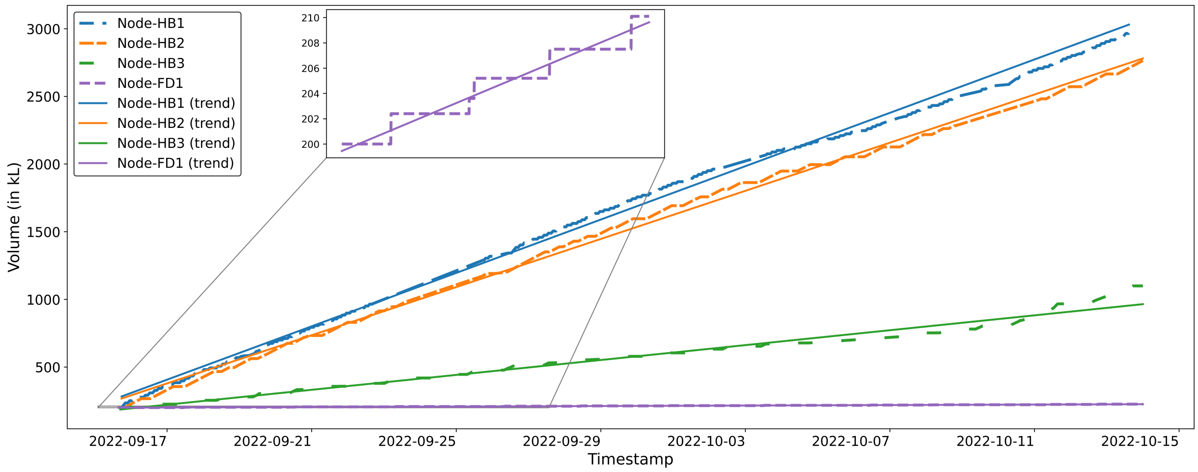


Figure 5.4 Time series plot for net water volume in kL. The dotted thick line shows the raw data for each node from September 17, 2022 to October 15,2022. The solid thin lines of the same color shows the best fit linear trend line corresponding to each node. (Best view in coloured)

analysis of all smart water meter nodes. The third subsection delves into monthly water usage analysis for all the meters, while the fourth subsection explores weekly water usage analysis.

5.3.1 Comparison between the digital and smart retrofit meters

Table 5.4, presents a comparison between two types of meters installed on the same pipeline: the smart retrofit meter (Node-FD1) and the off-the-shelf digital meter (Shenitech Node-FD2). Remarkably, the values recorded by the smart retrofit meter on the analog device closely align with those obtained from the digital meter. The primary distinction lies in the precision of the values. However, this difference becomes negligible when we round off the precision of the digital meter's values from three digits to one decimal place, making them similar to the readings of the smart analog meters.

Date (DD/MM)	22/10	23/10	24/10	25/10	26/10	27/10	28/10	29/10	30/10	31/10	01/11
Retrofit Meter	0	11.4	0	0	0	0	2.1	0	0	2.3	0
Digital Meter	0	11.605	0.003	0.004	0.003	0.003	2.164	0.005	0.004	2.397	0

Table 5.4 Table for comparison between net water volume flowing in a day for both analog and digital meter.

Node	Average Daily Consumption (kL/Day)
Node-HB1	98.60
Node-HB2	88.95
Node-HB3	27.52
Node-FD1	0.90

Table 5.5 Table for the Average Daily consumption for each node found using the best fitting linear trend line.

5.3.2 Time series data

Fig. 5.4 shows the time series data collected from all the nodes for approximately a month (from 17 September 2022 to 15 October 2022). All the meter readings are plotted here by adjusting them to zero on 17 September 2022 for ease of visualization. The thin solid line indicates the best fit linear trend line for the plotted raw data (thick line of the same colour) for a node. It can be seen that the amount of water used is highest for Node-HB1 and Node-HB2 followed by Node-HB3, and Node-FD1 has the least water usage. Similar observations can also be noted from table 5.5.

Note that Nodes-HB1 and HB2 are for the same building A, where Node-HB1 is on the pipeline between the borewell to sump and Node-HB2 is from sump to OHT. That is the reason they are almost close to each other, except for a small difference which is because of the water storage in sumps. It can be observed that the Nodes-HB1/HB2 and HB3 present in the hostel blocks have the highest volume of water flowing throughout the month along with the highest slopes. Slope (kL per day). This is expected as buildings A and B are hostel blocks with occupancy of around 800 and 425 students, respectively. Nodes HB1 and HB2 are much higher than Node-HB3 also because building A has two messes while building B does not have any. On the other hand, Node-FD1, responsible for supplying drinking water, displays the lowest slope, indicating the least volume of water flowing through it. It is intriguing to note the distinct steps in the curve, implying that the drinking water is supplied at intervals of a day or two, from the sump to the OHT. This suggests that the water in the storage tank remains unused for an extended period before being consumed.

5.3.3 Monthly analysis

Fig. 5.5 shows the monthly supplied water through all the nodes from the month of August to November 2022. In order to find these values, the meter reading at the end of the month was subtracted from the start of the month. From the graph a similar observation as to time series data can be observed, where nodes present at the hostels are showing higher monthly consumption, compared to other nodes. From the figure, observations made are:

- Hostel nodes-HB2 and HB3 show higher water demand for the month of September and November. This behaviour can be related to the high occupancy of the students because of the semester exams conducted in that period.

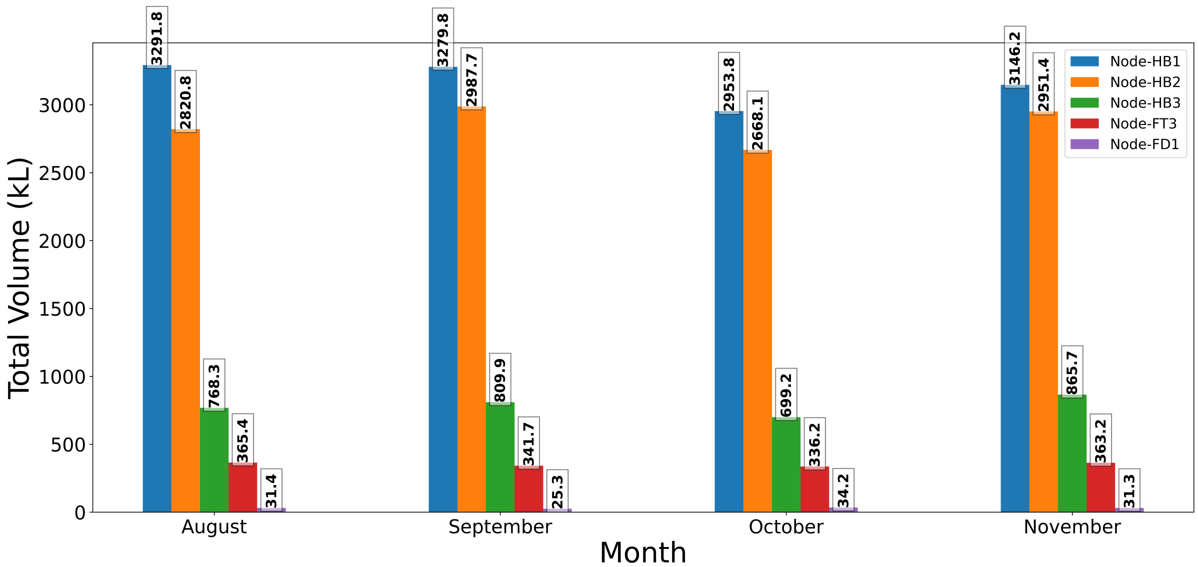


Figure 5.5 Monthly net water supply plot.

- Hostel nodes-HB2 and HB3 also show a reduced water demand in the month of October due to the low occupancy in hostels because of several holidays in that month, when many students go home.

- For Node-FT3 in the faculty and staff block, constant usage of tap water over the whole data can be seen. This can be attributed to the fact that the water demand remains constant as per family's requirements and do not get influenced by exam or small holidays during the semester.

5.3.4 Weekly analysis

Fig. 5.6 illustrates the total volume of water supplied on a weekly basis over a span of four weeks. Specifically, these four weeks were chosen to highlight the impact of consecutive holidays during the third week, from 3^{rd} October to 9^{th} October. During this period, a considerable number of students left the campus, leading to a decrease in active occupancy within the Hostel block. Consequently, the water consumption and supply in this block also decreased. The figure clearly indicates a noticeable dip in the weekly water supply for the Hostel Block nodes during the third week. In contrast, the Faculty/Staff Block experiences only a minimal dip during the same period. This observation underscores the fact that long holidays have a more substantial effect on the active occupancy of the Hostel block compared to the Faculty/Staff block. The reason behind this discrepancy lies in the nature of occupancy within these blocks. The Hostel block serves as a temporary residence for students, leading to a more significant impact on water usage when they are away. On the other hand, the Faculty/Staff block represents a permanent residence for the faculty, resulting in a relatively stable occupancy even during extended holidays.

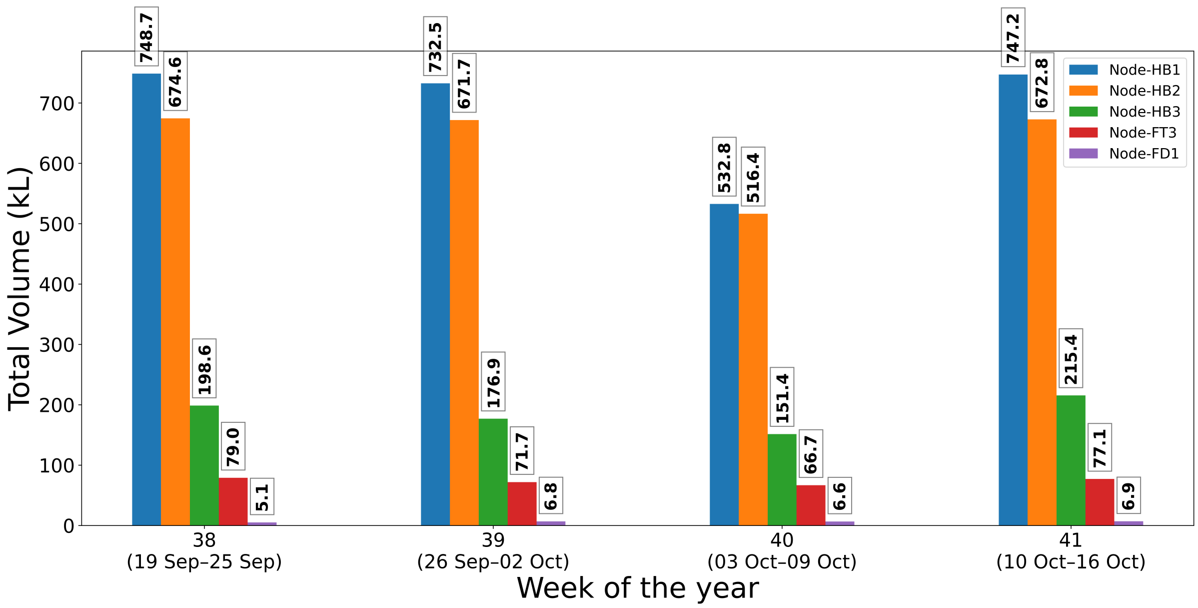


Figure 5.6 Weekly net water supply plot over 4 weeks.

Fig. 5.7 shows average volume of the water supplied for each day of the week for different nodes over available data in 4 months (August to November 2022). The idea behind this plot is that the average supply over a long period would be similar to the average consumption over a long period. The figure shows that the supply for each day of the week has little variation. Some intuitive observations made from the plot are:

- Water usage in the hostel block nodes is lower on weekends compared to other days. This is due to the cleaning staff of hostels having a holiday on Sunday. Students also start their day late by spending more time resting and some students go out of the hostel accounting for the least consumption. Conversely, water usage remains relatively consistent on the other days.
- In Node-FT3 of the faculty and staff block, water usage remains consistently stable across all days. This can be attributed to the families residing in the building, as their water requirements remain constant, unlike the variable needs of students.

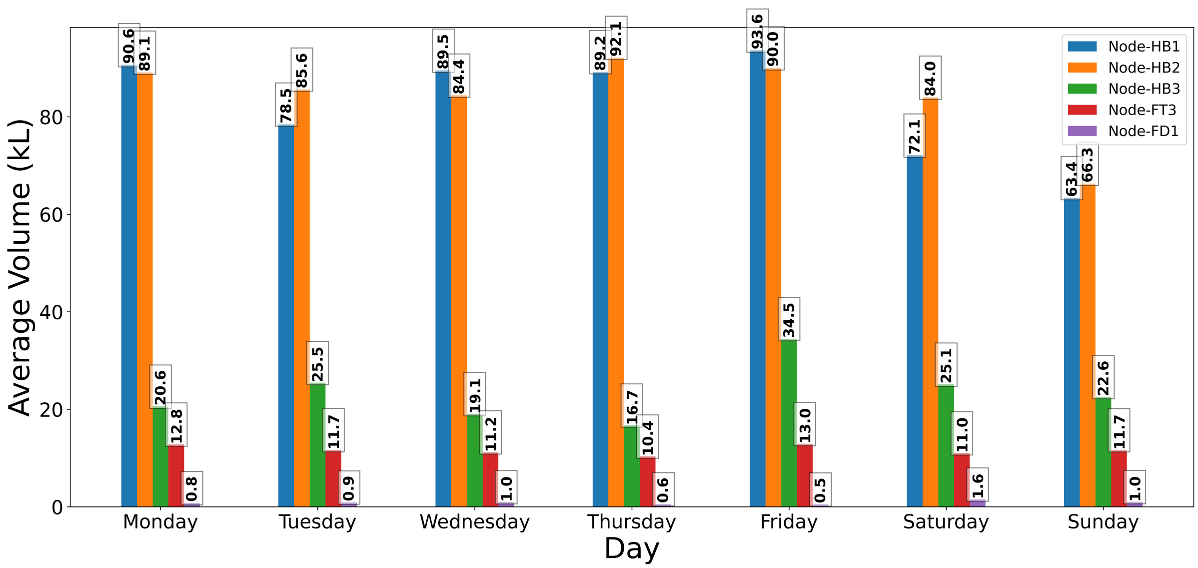


Figure 5.7 Week-day wise net water supply plot.

Chapter 6

Conclusion

The thesis presents an innovative approach to transform traditional analog water meters into smart devices using IoT-based techniques. The retrofit model leverages AI-based digit detection to extract numerical readings from the analog meters, enabling the transmission of these values to a cloud server. A unique and extensive dataset, specifically tailored to Indian water meters, was collected during the study. To gather data, ten nodes were strategically deployed across the campus during the data collection phase. Among these nodes, four were strategically placed to analyze the consumption patterns in the hostel and faculty/staff block, which are the most densely populated buildings on campus. The deployed devices provided valuable time series data, allowing for an in-depth exploration of monthly and weekly consumption patterns throughout one semester. Additionally, the study investigated how consumption patterns varied during holiday periods within the semester.

In Chapter 3 of the thesis, the retrofit model's development is detailed, encompassing its structural design, the equipment employed, and the algorithm utilized. For digit detection, a machine learning model based on RF was adopted. The retrofit model was initially deployed on a single meter, serving as the data collection source for training the ML model. Following data collection and model training, a post-processing step was integrated into the algorithm, resulting in an impressive accuracy of 97.69% for digit recognition.

Chapter 4 builds upon the work conducted in Chapter 3, expanding the retrofit deployment to encompass six meters. This extension resulted in the collection of a substantial dataset from these meters, which was then used to train a DL model achieving over 99% accuracy. Interestingly, it was observed that the previously trained ML model exhibited lower accuracy compared to its earlier performance. To investigate this behaviour, a thorough analysis of the nodes' locations was conducted. The deployed nodes exhibited significant diversity, with some placed in dark enclosures, others in open sunlight, and some in shaded areas. These diverse locations led to variations in the intensity of images captured by the devices, thereby impacting the training features. To gain further insights into how the ML model perceived the images, the concept of Hoggles was employed. Hoggles involved reverting the trained HOG features to visualize how the ML model actually interpreted the images. In order to enhance the detection algorithm, a transfer learning-based ResNet-18 DL model was trained, and a comparative analysis was carried out between the features learned by both the DL and ML models.

Chapter 5 presents a comprehensive behavioral analysis of the collected data set, focusing on the time series data. This analysis delves into the monthly and weekly consumption patterns observed in the hostel and faculty/staff block. Furthermore, the study examines how consumption patterns evolve in response to holidays throughout the semester.

Future directions: Utilize the collected time series data to conduct a comprehensive analysis of motor operation, with the goal of identifying the ideal duration for ensuring an ample water supply at the user-end. Explore the potential benefits of adjusting the motor's ON/OFF cycles to optimize its long-term efficiency and overall health. Additionally, enhance the performance of the storage units by examining water stagnation in the tanks and making necessary adjustments to the floater's level thresholds, thereby preventing any unnecessary water supply through the automatic water motor.

Bibliography

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] M. Marjani, F. Nasaruddin, A. Gani, A. Karim, I. A. T. Hashem, A. Siddiqa, and I. Yaqoob, “Big IoT data analytics: Architecture, opportunities, and open research challenges,” *IEEE Access*, vol. 5, pp. 5247–5261, 2017.

[3] G. Mastorakis, C. X. Mavromoustakis, J. M. Batalla, and E. Pallis, *Convergence of artificial intelligence and the Internet of Things*. Springer, 2020.

[4] W. J. Cosgrove and D. P. Loucks, “Water management: Current and future challenges and research directions,” *Water Resour. Res.*, vol. 51, no. 6, pp. 4823–4839, Jun. 2015.

[5] C. P. C. B. India, “Report on AQI, control of urban pollution series,” https://app.cpcbccr.com/ccr_docs/FINAL-REPORT_AQI_.pdf, 2015.

[6] S. Datta, I. H. Mafat, and R. Saxena, “Sensitivity analysis of water wastage in indian households,” *Materials Today: Proceedings*, vol. 77, pp. 111–116, 2023, advances in Water Treatment and Management. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214785322066585

[7] E. Vallino, L. Ridolfi, and F. Laio, “Measuring economic water scarcity in agriculture: a cross-country empirical investigation,” *Environmental Science Policy*, vol. 114, pp. 73–85, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S146290111931562X

[8] F. Malin, “Growing water scarcity in agriculture: future challenge to global water security,” *Phil. Trans. R. Soc. A*, 2013.

[9] M. Singh and S. Ahmed, “IoT based smart water management systems: A systematic review,” *Materials Today: Proceedings*, vol. 46, pp. 5211–5218, 2021, international Conference on Innovations in Clean Energy Technologies (ICET2020). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214785320364701

[10] ——, “IoT based smart water management systems: A systematic review,” *Materials Today: Proceedings*, vol. 46, pp. 5211–5218, 2021, international Conference on Innovations in Clean Energy Technologies (ICET2020). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214785320364701

[11] T. Perumal, M. N. Sulaiman, and C. Y. Leong, “Internet of things (IoT) enabled water monitoring system,” in *IEEE 4th Global Conference on Consumer Electronics (GCCE)*, 2015, pp. 86–87.

[12] N. R. Moparthi, C. Mukesh, and P. Vidya Sagar, “Water quality monitoring system using IoT,” in *Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, 2018, pp. 1–5.

[13] R. K. Kodali and A. Sahu, “An IoT based soil moisture monitoring on losant platform,” in *2nd International Conference on Contemporary Computing and Informatics (IC3I)*, 2016, pp. 764–768.

[14] R. et al., “An efficient IoT-based smart water meter system of smart city environment,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 8, 2021.

[15] A. Ray and S. Goswami, “IoT and cloud computing based smart water metering system,” in *2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC)*, 2020, pp. 308–313.

[16] T. Randall and R. Koech, “Smart water metering technology for water management in urban areas,” *Water eJ*, vol. 4, pp. 1–14, 2019.

[17] S. C. Hsia, S.-H. Wang, and S.-W. Hsu, “Smart water-meter wireless transmission system for smart cities,” *IEEE Consumer Electronics Magazine*, vol. 10, no. 6, pp. 83–88, 2021.

[18] M. Parto, C. Saldana, and T. Kurfess, “A novel three-layer iot architecture for shared, private, scalable, and real-time machine learning from ubiquitous cyber-physical systems,” *Procedia Manufacturing*, vol. 48, pp. 959–967, 2020, 48th SME North American Manufacturing Research Conference, NAMRC 48. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2351978920315870

[19] Mashal and et al., “Choices for interaction with things on internet and underlying issues,” *Ad Hoc Networks*, vol. 28, pp. 68–90, 2015.

[20] O. Said and M. Masud, “Towards internet of things: Survey and future vision,” *International Journal of Computer Networks*, vol. 5, no. 1, pp. 1–17, 2013.

[21] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, “Future internet: the internet of things architecture, possible applications and key challenges,” in *2012 10th international conference on frontiers of information technology*. IEEE, 2012, pp. 257–260.

[22] Ansar and et al., "Security in IoT layers: Emerging challenges with countermeasures," in *Computer Vision and Robotics*, P. K. Shukla, K. P. Singh, A. K. Tripathi, and A. Engelbrecht, Eds. Singapore: Springer Nature Singapore, 2023, pp. 551–563.

[23] M. Kumar, K. S. Raju, D. Kumar, N. Goyal, S. Verma, and A. Singh, "An efficient framework using visual recognition for IoT based smart city surveillance," *Multimed. Tools Appl.*, vol. 80, no. 20, pp. 31 277–31 295, Jan. 2021.

[24] A. K. Lall, et al., "Making analog water meter smart using ml and IoT-based low-cost retrofitting," in *2021 8th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2021, pp. 157–162.

[25] M. Kashyap, V. Sharma, and N. Gupta, "Taking mqtt and nodemcu to IoT: Communication in internet of things," *Procedia Computer Science*, vol. 132, pp. 1611–1618, 2018, international Conference on Computational Intelligence and Data Science. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918308585

[26] M. Iglesias-Urkia, A. Orive, and A. Urbieta, "Analysis of coap implementations for industrial internet of things: A survey," *Procedia Computer Science*, vol. 109, pp. 188–195, 2017, 7th International Conference on Sustainable Energy Information Technology, SEIT 2017, 16-19 May 2017, Madeira, Portugal. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050917309870

[27] R. Herrero, "Analytical model of IoT coap traffic," *Digital Communications and Networks*, vol. 5, no. 2, pp. 63–68, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352864816300840

[28] J. Haxhibeqiri, "A survey of LoRaWAN for IoT: From technology to application. " sensors," *Sensors*, vol. 18, 2018.

[29] K. Haque, "Comprehensive performance analysis of zigbee communication: An experimental approach with XBee S2C module. " sensors," *Sensors*, vol. 22, 2022.

[30] A. K. Lall, et al., "Improving IoT-based smart retrofit model for analog water meters using dl based algorithm," in *2022 9th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2022, pp. 207–212.

[31] N. Nilesh, J. Narang, A. Parmar, and S. Chaudhari, "IoT and ml-based aqi estimation using real-time traffic data," in *IEEE 8th World Forum on Internet of Things (WF-IoT)*, 2022, pp. 1–6.

[32] N. Nilesh, I. Patwardhan, J. Narang, and S. Chaudhari, "IoT-based aqi estimation using image processing and learning methods," in *2022 IEEE 8th World Forum on Internet of Things (WF-IoT)*, 2022, pp. 1–5.

[33] *National Air Quality Monitoring Programme (NAMP)*, accessed 25 March 2023, https://cpcb.nic.in/about-namp/.

[34] P. P. Ray, "A survey of IoT cloud platforms," *Future Computing and Informatics Journal*, vol. 1, no. 1, pp. 35–46, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2314728816300149

[35] C. R. Reddy and et al., "Improving spatio-temporal understanding of particulate matter using low-cost IoT sensors," in *IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, 2020, pp. 1–7.

[36] V. Bali, S. Mathur, V. Sharma, and D. Gaur, "Smart traffic management system using IoT enabled technology," in *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, 2020, pp. 565–568.

[37] S. Tiwari, J. Rosak-Szyrocka, and J. Żywiołek, "Internet of things as a sustainable energy management solution at tourism destinations in india," *Energies*, vol. 15, no. 7, p. 2433, 2022.

[38] K. Darshan and K. Anandakumar, "A comprehensive review on usage of internet of things (IoT) in healthcare system," in *2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*. IEEE, 2015, pp. 132–136.

[39] M. H. Kashani, M. Madanipour, M. Nikravan, P. Asghari, and E. Mahdipour, "A systematic review of IoT in healthcare: Applications, techniques, and trends," *Journal of Network and Computer Applications*, vol. 192, p. 103164, 2021.

[40] "Vice President calls for addressing the shortage of doctors and paramedical workers in a mission mode," https://www.pib.gov.in/PressReleasePage.aspx?PRID=1758000, accessed: 07 May 2023.

[41] H. Zhang, J. Li, B. Wen, Y. Xun, and J. Liu, "Connecting intelligent things in smart hospitals using nb-IoT," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1550–1560, 2018.

[42] M. He and J. Shi, "Circulation traceability system of chinese herbal medicine supply chain based on internet of things agricultural sensor," *Sustainable Computing: Informatics and Systems*, vol. 30, p. 100518, 2021.

[43] F. Jamil, S. Ahmad, N. Iqbal, and D.-H. Kim, "Towards a remote monitoring of patient vital signs based on IoT-based blockchain integrity management platforms in smart hospitals," *Sensors*, vol. 20, no. 8, p. 2195, 2020.

[44] A. Shamayleh, M. Awad, and J. Farhat, "IoT based predictive maintenance management of medical equipment," *Journal of medical systems*, vol. 44, pp. 1–12, 2020.

[45] A. O. Khadidos, S. Shitharth, A. O. Khadidos, K. Sangeetha, and K. H. Alyoubi, “Healthcare data security using IoT sensors based on random hashing mechanism,” *Journal of Sensors*, vol. 2022, p. 8457116, Jun 2022. [Online]. Available: https://doi.org/10.1155/2022/8457116

[46] J. Xu, B. Gu, and G. Tian, “Review of agricultural IoT technology,” *Artificial Intelligence in Agriculture*, vol. 6, pp. 10–22, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2589721722000010

[47] R. Akhter and S. A. Sofi, “Precision agriculture using IoT data analytics and machine learning,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, Part B, pp. 5602–5618, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157821001282

[48] O. Unold and et al., “IoT-based cow health monitoring system,” in *Lecture Notes in Computer Science*, ser. Lecture notes in computer science. Cham: Springer International Publishing, 2020, pp. 344–356.

[49] B. Li and J. Yu, “Research and application on the smart home based on component technologies and internet of things,” *Procedia Engineering*, vol. 15, pp. 2087–2092, 2011, cEIS 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877705811018911

[50] Taryudi, D. B. Adriano, and W. A. C. Budi, “IoT-based integrated home security and monitoring system,” *Journal of Physics: Conference Series*, vol. 1140, no. 1, p. 012006, dec 2018. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1140/1/012006

[51] H. Khayyam, B. Javadi, M. Jalili, and R. N. Jazar, *Artificial Intelligence and Internet of Things for Autonomous Vehicles*. Cham: Springer International Publishing, 2020, pp. 39–68. [Online]. Available: https://doi.org/10.1007/978-3-030-18963-1_2

[52] K. S. Viswanadh and et al., “Cv and IoT-based remote triggered labs: Use case of conservation of mechanical energy,” in *9th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2022, pp. 100–106.

[53] “MIT iLabs,” http://icampus.mit.edu/projects/ilabs/, accessed: 28 March 2023.

[54] “RT Labs - NITK,” http://rtlabs.nitk.ac.in/?q=page/rt-lab, accessed: 28 March 2023.

[55] “UNILabs,” https://unilabs.dia.uned.es/, accessed: 28 March 2023.

[56] P. Lea, *IoT and Edge Computing for Architects - Second Edition*. Packt, 2020. [Online]. Available: https://www.oreilly.com/library/view/{I}o{T}-and-edge/9781839214806/

[57] D. Hanes and et al., *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things*. Cisco Press, 2017. [Online]. Available: https://ptgmedia.pearsoncmg.com/images/9781587144561/samplepages/9781587144561_CH08.pdf

[58] G. Ihita, K. Viswanadh, Y. Sudhansh, S. Chaudhari, and S. Gaur, "Security analysis of large scale IoT network for pollution monitoring in urban india," in *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, 2021, pp. 283–288.

[59] Y.-W. Lee, S. Eun, and S.-H. Oh, "Wireless digital water meter with low power consumption for automatic meter reading," in *2008 International Conference on Convergence and Hybrid Information Technology*, 2008, pp. 639–645.

[60] Y.-S. Hong and C. Lee, "A design and implementation of low-power ultrasonic water meter," *Smart Water*, vol. 4, 11 2019.

[61] E. B. Rizzo, F. A. Cousin, R. M. Lucca, and S. R. Lautenschlager, "Autonomous metering system for monitoring water consumption," *AQUA - Water Infrastructure, Ecosystems and Society*, vol. 70, no. 6, pp. 797–810, 07 2021. [Online]. Available: https://doi.org/10.2166/aqua.2021.049

[62] "Creative engineers insertion type electromagnetic flow meter, display: Digital," https://www.indiamart.com/proddetail/insertion-type-electromagnetic-flow-meter-1237102633.html, accessed: 07-08-2023.

[63] "Water ultrasonic flowmeter, for industrial," https://www.indiamart.com/proddetail/ultrasonic-flowmeter-22236502597.html, accessed: 07-08-2023.

[64] "Turbine flow meter," https://www.indiamart.com/proddetail/turbine-flow-meter-20423545073.html, accessed: 07-08-2023.

[65] S. Alvisi and et al., "Wireless middleware solutions for smart water metering," *Sensors*, vol. 19, no. 8, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/8/1853

[66] N. Pimenta and P. Chaves, "Study and design of a retrofitted smart water meter solution with energy harvesting integration," *Discover Internet of Things*, vol. 1, pp. 1–15, 2021.

[67] M. J. Mudumbe and A. M. Abu-Mahfouz, "Smart water meter system for user-centric consumption measurement," in *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*, 2015, pp. 993–998.

[68] H. J. C. Barreto, Kurtulan, S. İnci, M. Nakıp, and C. Güzeliş, "Converting utility meters from analogue to smart based on deep learning models," *2020 Innovations in Intelligent Systems and Applications Conference*, 2020.

[69] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[70] Y. LeCun *et al.*, "Lenet-5, convolutional neural networks," *URL: http://yann. lecun. com/exdb/lenet*, vol. 20, no. 5, 2015.

[71] M. Vanetti, I. Gallo, and A. Nodari, "Gas meter reading from real world images using a multi-net system," *Pattern Recognition Letters*, vol. 34, no. 5, pp. 519–526, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167865512003832

[72] S. Bajaj, C. Teja S., and P. K. Yemula, "Computer vision based energy monitoring system using meter image capturing system (MICAPS)," in *First International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2020.

[73] M. Suresh, U. Muthukumar, and J. Chandapillai, "A novel smart water-meter based on IoT and smartphone app for city distribution management," in *IEEE Region 10 Symposium (TENSYMP)*, 2017, pp. 1–5.

[74] C. Son, S. Park, J. Lee, and J. Paik, "Deep learning–based number detection and recognition for gas meter reading," *IEIE Transactions on Smart Processing & Computing*, vol. 8, pp. 367–372, 10 2019.

[75] S. Lingireddy and T. R. Bhatt, *Essential Tools for Effective Transition from Intermittent to Continuous Water Supply*, pp. 267–277. [Online]. Available: https://ascelibrary.org/doi/abs/10.1061/9780784480632.020

[76] E. Kumpel and K. L. Nelson, "Intermittent water supply: Prevalence, practice, and microbial water quality," *Environmental Science & Technology*, vol. 50, no. 2, pp. 542–553, 2016, pMID: 26670120. [Online]. Available: https://doi.org/10.1021/acs.est.5b03973

[77] S. Mohapatra, A. Sargaonkar, and P. K. Labhasetwar, "Distribution network assessment using epanet for intermittent and continuous water supply," *Water resources management*, vol. 28, pp. 3745–3759, 2014.

[78] P. Klingel, "Technical causes and impacts of intermittent water distribution," *Water Supply*, vol. 12, no. 4, pp. 504–512, 07 2012. [Online]. Available: https://doi.org/10.2166/ws.2012.023

[79] T. R. Gurung, R. A. Stewart, A. K. Sharma, and C. D. Beal, "Smart meters for enhanced water supply network modelling and infrastructure planning," *Resources, Conservation and Recycling*, vol. 90, pp. 34–50, Sep. 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921344914001347

[80] K. A. Nguyen, R. A. Stewart, H. Zhang, O. Sahin, and N. Siriwardene, "Re-engineering traditional urban water management practices with smart metering and informatics," *Environmental Modelling & Software*, vol. 101, pp. 256–267, Mar. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364815217305893

[81] J. E. Pesantez, E. Z. Berglund, and N. Kaza, "Smart meters data for modeling and forecasting water demand at the user-level," *Environmental Modelling & Software*, vol. 125, p. 104633, Mar. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364815219303457

[82] Raspberry-Pi micro-controller module, accessed 21 Feb. 2021. [Online]. Available: https://raspberrypi.org/

[83] ThingSpeak: Cloud data collection and analysis using MATLAB, accessed 27 Feb. 2021. [Online]. Available: https://thingspeak.org

[84] JioFi JMR540: LTE based portable WiFi hotspot, accessed 27 Feb. 2021. [Online]. Available: https://www.jio.com/shop/en-in/router-jmr540-black-/p/491193576

[85] Bureau of Indian Standards: Product certification, accessed 12 Mar. 2021. [Online]. Available: https://bis.gov.in/index.php/product-certification/

[86] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. USA: Prentice-Hall, Inc., 2006.

[87] L. Breiman, "Random forests," *Machine learning 45.1 (2001): 5-32*, 2001.

[88] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[89] S. van der Walt and et al., "scikit-image: image processing in Python," *PeerJ*, accessed 10 Feb. 2021. [Online]. Available: https://scikit-image.org

[90] Scikit-Learn: Machine learning in Python, accessed 10 Feb. 2021. [Online]. Available: https://scikit-learn.org

[91] OpenCV: Python library for real-time computer vision, accessed 10 Feb. 2021. [Online]. Available: https://opencv.org

[92] S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures," *IEEE Access*, vol. 6, pp. 64 270–64 277, 2018.

[93] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[94] J. Deng, W. Dong, R. Socher, L.-J. Li *et al.*, "Imagenet: A large-scale hierarchical image database," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[95] A. Paszke, S. Gross, F. Massa, A. Lerer *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *33rd Conference on Neural Information Processing Systems*, pp. 8024–8035, 2019.

[96] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba, "HOGgles: Visualizing object detection features," *IEEE International Conference on Computer Vision*, pp. 1–8, 2013.

[97] A. Mahendran and A. Vedaldi, “Understanding deep image representations by inverting them,” *IEEE conference on computer vision and pattern recognition*, pp. 5188–5196, 2015.