Engineering Research Express

PAPER

ANN based traffic congestion analysis applied to parking recommendation system for electric three-wheelers

RECEIVED
20 December 2024

REVISED
25 April 2025

ACCEPTED FOR PUBLICATION
7 May 2025

PUBLISHED
16 May 2025

Suman Haldar[1], **Arindam Mondal**[2,*], **Rajkumar Maity**[3] **and Rajib Banerjee**[4]

[1] Department of Electronics and Communication Engineering, Neotia Institute of Technology Management and Science, West Bengal, 743368, India
[2] Department of Electrical Engineering, Dr B.C Roy Engineering College, Durgapur, West Bengal, 713206, India
[3] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, West Bengal, 700108, India
[4] School of Computer Science, UPES, Dehradun, Uttarakhand, 248007, India
[*] Author to whom any correspondence should be addressed.

E-mail: sumanececkt@gmail.com, arininstru@gmail.com, rajkumar.maity.work@gmail.com **and** rajib123banerjee@gmail.com

Keywords: battery monitoring, data logger, deep learning, artificial neural network, internet of things, electric three-wheeler

Abstract

The increasing adoption of Electric Vehicles (EVs) is transforming transportation systems with an emphasis on sustainability. In the Asia-Pacific region, Electric Three Wheelers (E3W) have gained significant market share due to their affordability and eco-friendly nature. However, the rise in E3W usage has also led to road congestion and parking challenges, particularly in suburban and rural areas. This paper presents a smart roadside parking recommendation system that utilizes a proposed Artificial Neural Network (ANN) to forecast traffic status and recommend nearby parking options. The system uses vehicle dynamics data such as voltage, current, and timestamps collected via an IoT-based data logger integrated with a cloud-based and map-based visualization interface. By analyzing the collected data, the system predicts real-time traffic status for parking recommendations, enabling an efficient parking solution that saves both time and energy. The experimental result demonstrates that the proposed ANN configuration gives the most optimal result in terms of accuracy, loss, MAE, MSE, RMSE, and R Square on purely unseen data in real-time scenarios. Moreover, the trained ANN model effectively categorizes the parking recommendation as 'Recommended,' 'Waiting State,' or 'Not Recommended' for specific parking areas. To validate the model performance, experiments are conducted in two distinct regions with unique routes and traffic patterns. The result is visualized through map API and demonstrates optimal parking recommendations as compared to real-world traffic scenarios, thereby improving decision-making for E3W drivers.

1. Introduction

The urban transport crisis in India arises from ongoing population growth, urbanization, suburban expansion, rising incomes, and a sharp increase in motor vehicle ownership and usage. Major cities in India face severe and escalating transport issues, including air pollution, noise, congestion, parking shortages, and high energy consumption [1]. Consequently, a shift to more sustainable transportation solutions is a crucial step to mitigate the negative impacts of the fossil fuel-dependent transport system [2].In this context, EVs sales are projected to grow substantially in the coming years. By 2035, the global market share for EVs is anticipated to reach 42.5%. In 2018, 2.1 million EVs were sold, resulting in a total year-end inventory of 5.4 million. By 2035, this inventory is expected to expand to 440 million EVs. Globally, it is estimated that 30% of all passenger vehicles will be EVs by 2032 [3]. In the Asia-Pacific region, E3W are gaining popularity due to their eco-friendly nature and low-cost last-mile transportation, especially in suburban and rural areas [4]. India, China, and the Association of Southeast Asian Nations (ASEAN) dominate the global three-wheeler market, with India leading in sales, reaching 19 million units in 2023. Three-wheelers are widespread in India, with 157 per 1,000 people, compared to just 35 passenger cars [5]. Among electric vehicles, E3Ws have shown the fastest growth, making up 57% of

the 1,392,265 electric vehicles on Indian roads as of August 2022, according to data from the Ministry of Road Transport and Highways [6]. This surge is driven by self-employment opportunities for uneducated and semi-skilled migrants moving to cities and suburban in search of livelihood [7]. As part of the para-transit sector, these vehicles contribute to traffic congestion due to weak enforcement of lane discipline and traffic rules, resulting in haphazard movement and worsening congestion. Moreover, the rapid spread of E3Ws is a major factor in traffic congestion and the shortage of outdoor parking spaces, especially in urban and suburban areas where indoor parking is limited. Inefficient roadside outdoor parking is further worsening traffic congestion. Consequently, people spend considerable time searching for suitable parking spots nearby. Therefore, an automated outdoor parking management system is one of the alternative solutions to save the timing and energy consumption of E3W vehicles. Such a solution takes initiative towards building smart traffic management. One effective solution could be an IoT-based smart parking framework integrated with effective data visualization, standardization and analytics, and smartphone integration with IoT technology [8].

This paper presents a system that delivers a smart parking solution by integrating an onboard multichannel data logger (MCDL) with Internet of Things (IoT) technology, a trained deep learning model, cloud storage, and a map-based interface as part of the parking recommendation system. The IoT framework gathers data such as latitude, longitude, voltage, and current consumption with timestamps from selected E3Ws and stores it on a cloud platform. After preprocessing, the data is input into a trained ANN model to predict traffic status around the E3Ws location. Based on these predictions, a color-coded map interface displays nearby roadside parking options and optimal routes, with colors representing different congestion levels along each path. A blue path indicates 'high congestion' and advises against parking in that area, a green path represents 'smooth traffic' and suggests the area as a viable parking option, while a red path signifies 'moderate congestion' and suggests waiting, as traffic may soon clear. This system streamlines the parking search process, reducing roadside congestion, conserving time, and saving battery power by offering efficient parking recommendations.

The paper is organized as follows: related work is discussed in section 2, methodology and implementation in section 3, experimental results and discussion in section 4, and conclusions and future work in section 5.

2. Related work

Smart parking systems powered by IoT technology and deep learning models improve urban mobility by reducing traffic congestion, conserving energy, and enhancing user experience. By efficiently directing drivers to available spots, they help lower carbon emissions and improve safety while providing valuable data for urban planning. Overall, these systems support sustainable, efficient, and organized city transportation. The researcher proposes an urban traffic flow prediction model based on the k-nearest neighbor (KNN) algorithm to address traffic congestion and assist in route planning. The model offers real-time, efficient traffic flow predictions and attains an average prediction time of 1.3 s with an accuracy of 91.1%. This method delivers valuable data for traffic management and helps drivers choose smoother routes [9]. A traffic forecasting model using machine learning is proposed to improve urban traffic management. In this model, five machine learning methods are tested after identifying key variables affecting traffic flow. The results show the KNN method achieving the highest accuracy (96.14%) and Kappa (0.95). The model, combined with a Geographic Information System (GIS), creates a traffic prediction map that assists both users and administrators in efficiently managing traffic [10]. Combining DBSCAN and the Random Forest algorithm a short-term traffic congestion prediction method is proposed to enhance urban travel conditions. The method achieves 94.36% accuracy on U.S. high-speed road data testing and efficiently predicts congestion levels using historical traffic speed and flow data [11]. In another research, traffic congestion is monitored by combining a piecewise switched linear (PWSL) model with a Kalman filter (KF) to simulate traffic flow. The model's outputs are applied to KNN schemes for accurate congestion detection. A closer examination of California's State Route 60 demonstrates the effectiveness of the method in accurately identifying traffic congestion patterns [12]. Traffic congestion forecasting has become a significant area of research in recent years. Especially through the use of machine learning (ML) and artificial intelligence (AI) in this area. The integration of big data platforms with intrusive and non-intrusive sensors and probe vehicle data, along with improvements in AI models, has greatly expanded this field. The review paper explores different methodologies and approaches applied in traffic congestion prediction, focusing primarily on short-term predictions using various machine learning models. The paper systematically summarizes the strengths and weaknesses of these models, categorizing them under different AI branches [13].The researcher focuses on developing a machine learning-based system for traffic congestion control in smart cities. A fusion-based intelligent traffic congestion control system for Vehicular Networks (FITCCS-VN) is utilized for this system. It collects traffic data and optimizes routing, providing drivers with real-time traffic information. The model achieves 95% accuracy, improving traffic flow and reducing congestion compared to previous methods [14]. The researcher develops an IoT-based Intelligent Transportation System (ITS) to manage road traffic

Table 1. Summary of Key findings, methodologies, and limitations from the literature.

References no	Key findings	Method used	Limitations
[9]	•KNN-based model predicts urban traffic flow in real-time with 91.1% accuracy.	•K-Nearest Neighbors (KNN)	•No IoT integration for real-time vehicle data •Limited to urban traffic flow.
[11]	•Short-term traffic congestion prediction using Random Forest and DBSCAN. •Historical speed and traffic flow data for congestion prediction and 94.36% accuracy using PEMs database (U.S.) for validation.	•Random Forest Algorithm, DBSCAN Clustering, Historical Traffic Data Analysis	•Only short-term predictions, lacks long-term traffic forecasting. •Trained on high-speed road data, may not generalize to collector road or mixed-traffic conditions. •Combining two ML Models added extra complexity.
[12]	•Hybrid PWSL-KF Observer for Traffic Congestion Detection. •KF-KNN Hybrid Detector for Precise Congestion Monitoring. •Validated on California SR-60 freeway data.	•Piecewise Switched Linear (PWSL) Model, Kalman Filter (KF), KNN, Kernel Density Estimation	•Designed specifically for freeway traffic, may not generalize to suburban or mixed-use roads. •High computational complexity due to multiple models (PWSL, KF, KNN). •No IoT or real-time vehicle data integration.
[14]	•FITCCS-VN: ML-Based Traffic Congestion Control System. •Vehicular Networks Enable 95% Accurate Traffic Congestion Prediction.	•Machine Learning (ML) Techniques, Vehicular Networks (VNs), Internet of Vehicles (IoV)	•Depends on costly VN infrastructure, limiting applicability in areas without roadside units (RSUs). •The study relies on simulation-based validation and lacks integration of real-time vehicle data.
[15]	•ANN-Based Traffic Congestion Classification Using in-road stationary sensors data. •Integrated into IoT-based Intelligent Transportation System (ITS) for automatic traffic regulation updates.	•ANN, ITS, In-road Stationary Sensors	•Relies on on road fixed sensors, limiting adaptability to real-time vehicle dynamics. •Traffic regulation adjustments focus only on congestion management, not energy efficiency.
[17]	•Quantile-Based ML for Accurate Energy Prediction with Uncertainty Estimation. •Online adaptive QRNN achieves a 5.04% error, improving state-of-the-art models by 35%.	•ML with physics-informed features, QRNN with online adaptation.	•The dataset is limited to 55 battery electric *t*axis of the same brand and model, reducing generalizability. •Does not predict how traffic scenarios impact energy consumption over time.
[18]	•Accurate estimation of EV energy consumption under real-world traffic conditions. •A novel data-driven model decomposes energy consumption into positive and negative kinetic energies, enhancing estimation accuracy.	•ML with real-world driving data and decomposed factors to predict energy consumption.	•Overlooks detailed congestion analyses derived from EV energy consumption patterns. •Not address parking recommendations for EVs based on real-time energy consumption data.
[20]	•Featured with Automated parking, location tracking, real-time invoicing, and payment system. •Integrated with web & Android apps for better user experience.	•Image processing, Raspberry Pi, camera, GSM, YOLOv3-tiny deep learning-based object detection model.	•YOLOv3-tiny may struggle in complex scenarios, requiring costly hardware, good lighting, and large storage. •Limited coverage to selected parts of the city and recommended for congested zones like shopping malls, offices, hospitals, etc
[26]	•Thing-Speak IoT cloud and ultrasonic sensors enable real-time parking detection. •Achieved 1.5 m average error versus 2.3 m (GPS) and 3.55 m (max *a posteriori*), improving accuracy by 35%+ with weighted k-NN.	•Wi-Fi fingerprinting & weighted k-NN, Raspberry Pi, ultrasonic sensors, and ThingSpeak.	•Relies on fixed sensors which may face challenges such as limited range. •W-KNN algorithm estimates user location but does not incorporate vehicular data for congestion analysis or roadside parking recommendations.

Table 1. (Continued.)

References no	Key findings	Method used	Limitations
[28]	•Proposed a smart parking system integrating IoT, WSN, and RFID to help drivers find available spaces in cities. •Supports linear and mass parking with real-time availability near the destination.	•Hybrid, adaptable self-organization algorithm to configure wireless sensor networks (WSNs).	•The system's performance in actual urban environments remains to be tested and evaluated. •High deployment costs for RFID tags, readers, and WSN infrastructure.

congestion by estimating and classifying traffic congestion states of different road segments. Using data from in-road stationary sensors, an ANN model classifies traffic states, enabling the system to automatically adjust traffic regulations and suggest alternate routes [15]. In another paper, the researcher proposes a data-based method to estimate traffic congestion on road segments between bus stops using smart cards and bus trajectory data. The self-organizing map (SOM) clusters traffic patterns and a congestion index ranks the most congested segments. A case study demonstrates the effectiveness of this approach for optimizing public transport networks [16]. In recent studies, the researcher proposes a machine learning-based framework to accurately forecast EV energy consumption by combining physics-informed features with vehicle-specific online adaptation. Tested on real-world EV data, the model improves prediction accuracy, achieving an average error reduction of 5.04% and enhancing the reliability of energy estimates, which aids in reducing range anxiety and optimizing charging strategies [17]. A data-driven model to estimate EV energy consumption at the link level, considering real-world traffic conditions, is proposed in another paper. The model decomposes energy consumption into positive and negative kinetic energy factors, providing more accurate estimations than existing models [18]. The researcher developed a smart car parking system that helps users find available parking spaces while minimizing search time. The system collects raw data locally, applies data filtering and fusion techniques to reduce transmission costs, and provides real-time road traffic congestion information. The transformed data is sent to the cloud, where machine learning algorithms process and evaluate it, offering an efficient solution for parking and traffic management [19]. The researchers developed a prototype featuring automated parking, location tracking, parking management, real-time invoicing, and a payment system. The system uses IoT, image processing, object detection, Firebase, and GPS/GSM modules, and is integrated with both web and mobile applications to enhance user experience [20]. An innovative Smart Parking Solution using Ultra High Frequency (UHF) sensors is proposed to provide real-time updates on available parking places, aiming to minimize traffic congestion in crowded areas. The solution integrated with a user-friendly app allows travelers to find and book parking places, with a cost-effective system design that is easy for organizations to implement [21]. A smart parking management system using IoT technology is proposed to efficiently tackle parking challenges in urban areas. It offers features such as reservation, payment, slot search, and real-time monitoring. By integrating hardware and costly sensors, the system aims to enhance parking management, reduce costs, and improve the quality of life in smart cities [22]. In another work, a smart parking system is proposed. Utilizing IoT technology in smart cities, the system enables drivers to reserve parking spots in advance, reducing congestion and pollution. It features post-trip bookings, automatic cashless billing, and hacking notifications while providing real-time updates to traffic police for better urban traffic management [23]. Using IoT technology and sensors another work proposes a smart parking solution for commercial areas, assisting users in quickly locating available parking spots and avoiding unnecessary travel. The system uses cloud and database technologies to guide vehicles to book spots, enhancing the parking experience in urban environments [24]. A smart parking system using computer vision is proposed to enhance parking efficiency and security in busy urban areas. The system enables precise monitoring within parking spaces by detecting vehicle license plates, helping users locate their parked cars, and improving security [25]. The recent work demonstrates a smart parking system for accurate vehicle positioning and real-time parking detection using Wi-Fi signal strength and a weighted k-nearest neighbors supervised learning algorithm. The system integrates IoT-enabled Raspberry Pi with an Android application. Data from ultrasonic sensors is processed on the Raspberry Pi and transmitted to the Thing Speak IoT cloud via Message Queue Telemetry Transport (MQTT). The Android app guides drivers to available parking spaces, improving convenience and efficiency [26]. A real-time lead-acid battery monitoring system for E3W vehicles, using the Atmega 328 and ESP8266 microcontrollers is demonstrated. The system is capable of asses battery health across various E3W drive cycles [27]. Using an adaptable hybrid self-organization algorithm, the researcher designed a smart parking system based on wireless sensor networks (WSN). This system efficiently manages energy consumption in wireless communication which in turn extends the lifespan of sensor nodes and the overall WSN. It offers innovative services that assist drivers in quickly locating available parking spaces near

their destination, enhancing parking efficiency in urban areas [28]. A summary of the congestion analysis and parking recommendations from the state-of-the-art work is shown in table 1.

2.1. Motivation and contribution

The motivation for this research arises from several key gaps in existing traffic congestion and parking recommendation systems, particularly concerning E3Ws. Most prior research primarily focuses on fossil fuel-powered vehicles and lacks real-time data integration tailored to EV-specific energy consumption patterns. Additionally, many existing models, such as KNN-based urban traffic prediction, Random Forest-DBSCAN short-term congestion models, and ANN-based ITS congestion classification, rely on historical traffic data or fixed roadside sensors. Such dependence is unsuitable for dynamic congestion analysis and real-time energy estimation in E3Ws. Many congestion detection approaches, including Hybrid PWSL-KF Observer and FITCCS-VN, are optimized for high-speed freeway traffic and depend on costly vehicular network infrastructure. These methods do not integrate real-time in-vehicle data such as voltage, current, speed, acceleration, and time sequencing which are critical for accurately predicting E3W energy consumption in suburban roads and highway intersections. Furthermore, existing EV energy prediction models focus on battery electric *t*axis or broad driving data, without addressing how traffic congestion impacts energy consumption. Parking-focused systems such as YOLOv3-based smart parking, IoT-cloud ultrasonic parking detection, and RFID-integrated smart parking rely on fixed infrastructure and are often designed for urban commercial zones, limiting their applicability in suburban and mixed-use roads where E3Ws predominantly operate. Addressing these gaps, this research develops a smart roadside parking recommendation system by analyzing road congestion, that utilizes real-time vehicle dynamics data and a deep learning model to provide an efficient and energy-saving parking solution for E3Ws in suburban areas.

The key contributions of this work are summarized below:

- Development of a smart roadside parking recommendation system specifically designed for E3Ws operating on suburban roads.
- Integration of an IoT-enabled multichannel data logger to collect real-time vehicle dynamics data, including voltage, current, and timestamps.
- Use of a cloud server for data pre-processing and a map-driven interface for real-time traffic status visualization and parking recommendations.
- Implementation of an ANN model optimized by systematically varying model configuration (layers and neurons) to enhance performance.
- Analysis of key performance metrics such as Test Accuracy, Validation Loss, Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Square Error (RMSE), and R^2 score to ensure robust traffic classification.
- Novel application of real-time vehicle data to predict traffic status, analyze energy consumption patterns, and recommend parking for E3Ws.
- Optimization of parking solutions to improve time and energy efficiency for E3Ws, addressing the unique challenges of suburban collector roads.

3. Methodology and implementation

3.1. Proposed system block diagram

The block diagram of the proposed parking recommendation system is shown in figure 1. The proposed system consists of three main components: an Onboard Multichannel Data Logger (MCDL), a Cloud-Based Vehicle Data Collection and Processing Unit, and an Output Visualization Interface. MCDL is designed to collect and monitor real-time voltage and current data from a 4.8 kWh lead-acid battery pack using a 4-channel resistive divider network and a split-hole current sensor. The collected data is processed through a low-pass filter with diode-based protection before being sent to a multiplexer (MUX). An Atmega 328 (8-bit microcontroller) converts the analog signals into digital using an ADC module, while a DS 3231 real-time clock (RTC) log time stamp and Micro SD memory module store battery data locally via an I2C interface. The processed data is then transmitted through an ESP8266 (32-bit microcontroller) via an IEEE 802.11 LAN network to a cloud-based server. The cloud-based vehicle data collection and processing unit receives the transmitted data, including voltage, current, and time stamps, for pre-processing and ML based analysis. Using ML models, the system classifies traffic congestion states and detects patterns in vehicle operation. The output visualization interface

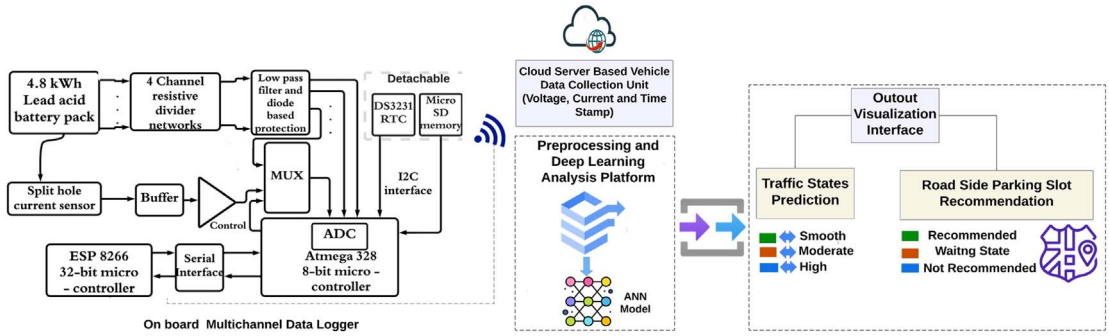

Figure 1. Proposed system block diagram.

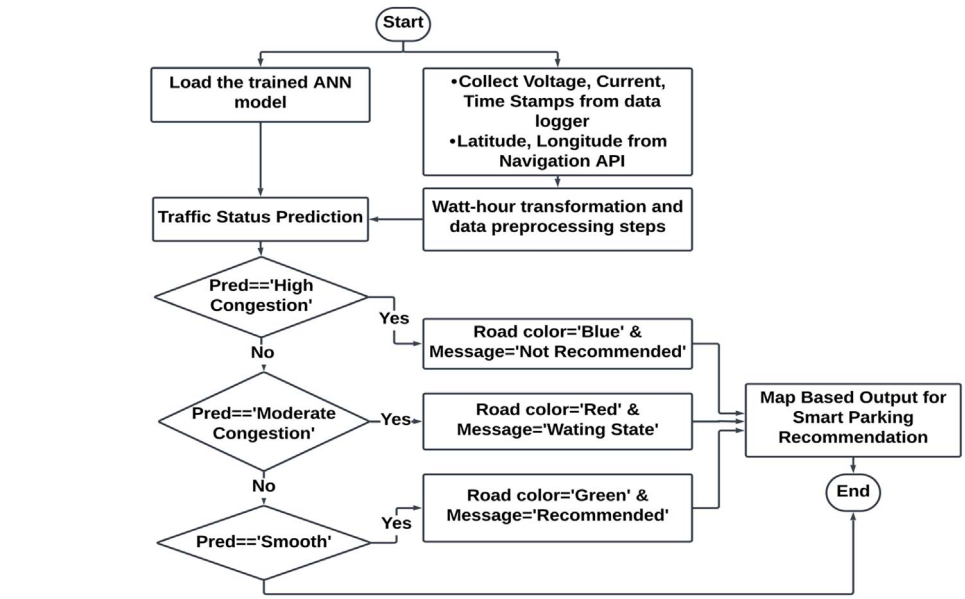

Figure 2. System level flow chart of proposed parking recommendation system.

utilizes the processed data for real-time traffic state prediction and roadside parking slot recommendations. By analyzing traffic conditions during various drive cycles of the E3W, the system provides intelligent decision-making for efficient parking slot allocation. This integrated approach enhances mobility in urban and semi-urban areas, particularly along collector roads, by optimizing parking slot allocation and improving traffic flow management.

3.2. System level flowchart

The system-level flowchart is depicted in figure 2. The flowchart represents a smart parking recommendation system designed for E3Ws based on real-time road congestion data. The workflow begins with data collection from an E3W vehicle that acts as the data host. As the host vehicle traverses a given area, it records voltage, current, and timestamps, along with real-time latitude and longitude coordinates obtained from a navigation API. These raw data points are transformed into watt-hour (Wh) values, which serve as the basis for determining traffic congestion levels through further analysis and preprocessing.

Once the data is preprocessed, a trained ANN model predicts the traffic status in the area, classifying it into one of three categories: Smooth, Moderate Congestion, or High Congestion. Based on the prediction, specific parking recommendations are generated. If the traffic status is predicted as 'High Congestion', the system marks the road in blue, indicating the message 'Not Recommended'. For 'Moderate Congestion,' the road is marked in red with the message 'Waiting State', suggesting the traffic will likely clear soon, making parking possible. If the prediction is 'Smooth', the road is marked in green with the message 'Recommended', signifying it is suitable for parking. These recommendations are visualized on a map, offering users actionable guidance for selecting optimal parking areas based on real-time congestion data. This system utilizes the historical congestion levels

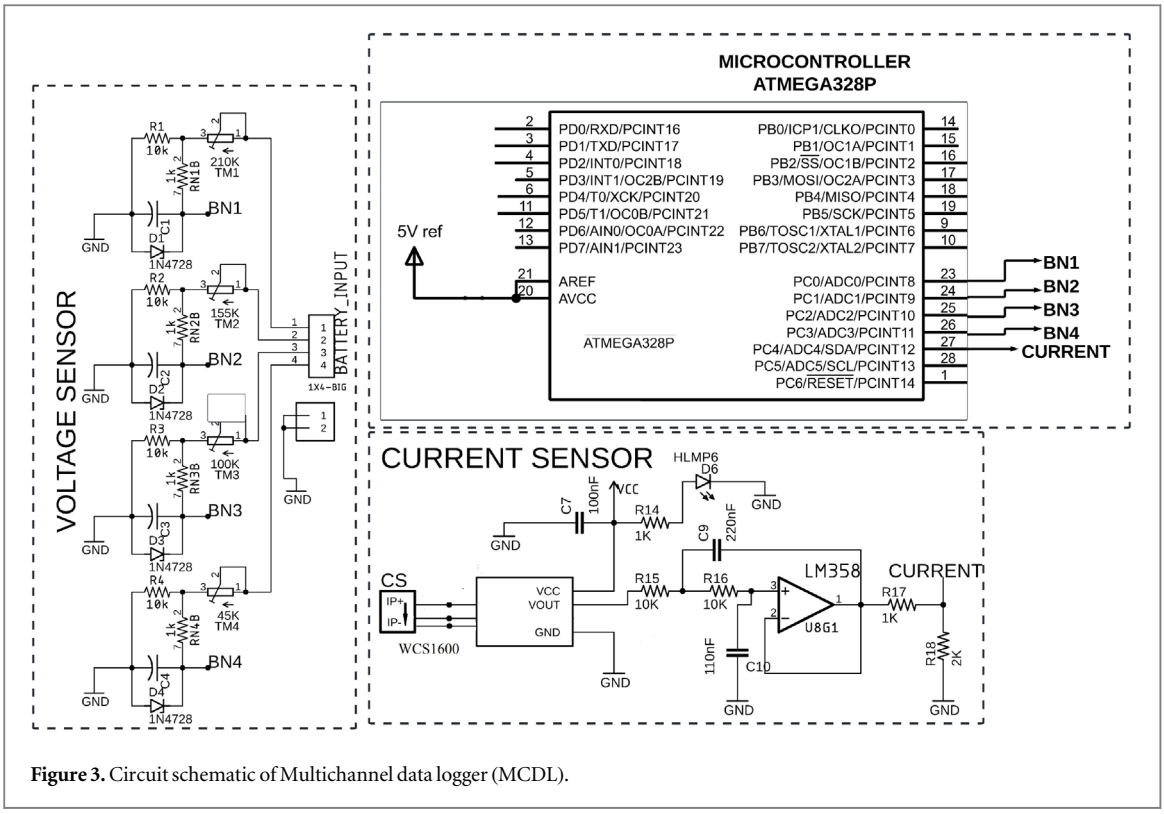

Figure 3. Circuit schematic of Multichannel data logger (MCDL).

encountered by a previously traversing E3W to guide subsequent vehicles in finding optimal parking spots, enhancing efficiency and reducing delays for E3W users.

3.3. Hardware implementation

Figure 3 illustrates the circuit schematic of MCDL. The developed MCDL [29] system monitors a 4.8 kWh lead-acid battery pack by measuring total pack voltage, individual node voltages (BN1–BN4), and bidirectional charging/discharging currents. Data is logged to an SD card for backup and transmitted in JSON format via IEEE 802.11 for real-time cloud monitoring. Voltage sensing is achieved using a terminal subtraction method, where voltages of successive nodes are measured and subtracted to isolate individual cell voltages. Each node is scaled using a dedicated voltage divider with ratio arms of 14:1, 10.5:1, 7.2:1, and 3.6:1 for BN1, BN2, BN3, and BN4, respectively. The scaled voltages are then passed through a 150 Hz low-pass filter before being fed to the ATMEGA-328's 10-bit ADC (VREF = 5 V). Zener diodes (1N4728) safeguard the ADC inputs from overvoltage. Current sensing employs the WCS1600 Hall-effect sensor (0–100 A, 22 mV A^{-1}), combined with a 100 Hz Sallen-Key filter and voltage divider, allowing the system to detect transient current changes of approximately 10 to 20 A s^{-1}, during rapid discharge cycles. This setup accurately monitors battery performance during different E3W drive cycles using C10–C20 PbA batteries.

3.4. Data procurement

Normally the behavior of the battery characteristics is non-linear either in a static or dynamic state. The effective ampere hour discharge or depth of discharge (DOD) has a significant impact on controlling the E3W drive train. The accumulation of currents flowing into and out of a battery and the terminal voltage under different drive cycles are the most popular indicators while analyzing road traffic congestion based on uniform speed, certain acceleration, and deceleration interval. In the proposed system, with the help of the enhanced coulomb counting method, the ampere-hour discharge is calculated in discrete time intervals during each driving lap of the E3W. Using node voltage measurement techniques, the effects of hard acceleration and their impact on batteries are measured.

3.4.1. Data collection process

The proposed MCDL is designed to collect battery-related data from a wide range of E3Ws, provided the battery specifications fall within the device operational limits specifically, a maximum pack voltage of up to 72 V and a peak discharge current of up to 100 A. This makes it compatible with commonly used battery types in E3Ws, such as lithium-ion (Li-ion) and lead-acid (PbA) batteries, as long as they remain within these parameters. An experimental setup was arranged using the developed MCDL module, connected to four series-connected

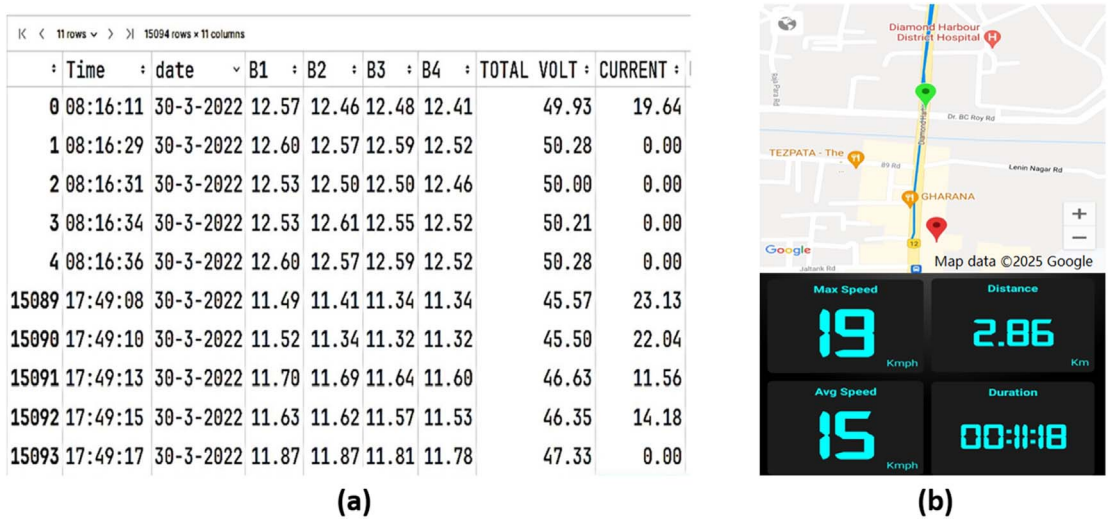

Figure 4. (a) Battery data stored in the cloud server for drive cycle analysis. (b) MAP-based API integration for tracking speed, distance, and duration [30]. Map data ©2025 Google.

tubular PbA batteries (each 12 V, 100 Ah, C20 rating). These batteries, with a total capacity of 4.8 kWh, supply the electrical energy needed to power a 900-watt, 3000 RPM, 48 V BLDC electric motor, enabling the acceleration and cruising of E3Ws. The designed E3Ws have a driving range of up to 80 km and a maximum speed of 25–27 km h^{-1}. Key vehicle operational parameters, including time stamps, speed, net discharging current, individual battery terminal voltages, total pack voltage, and location data, were monitored in real-time. To calculate energy consumption, battery-related data (e.g., time stamp, voltage, and current) was transmitted to a cloud-based vehicle data collection platform via wireless communication at a nominal frequency of 0.5 Hz. To collect time-series data across various traffic conditions, 70 E3Ws were deployed across different road segments, operating at varying times and experiencing different levels of road congestion. The data collection spanned 90 days, covering a 60 km^2 area. The study focused on suburban collector roads and major highway intersections in the southeastern coastal region of West Bengal, India. Different E3W drive cycles were considered, ranging from short laps (10 min) to extended driving laps (210 min). These drive cycles were considered in typical rural and semi-urban driving conditions, characterized by speed variations (idle to peak speeds of 10–27 km h^{-1}), acceleration and deceleration phases, idle periods, and constant-speed segments. To enhance computational efficiency, idle periods and deceleration phases are filtered out. This eliminates low-energy consumption phases that do not significantly contribute to trip-related analysis. Figure 4 shows the dataset used for drive cycle analysis. Figure 4(a) illustrates battery data stored in the cloud server, including individual battery voltages, total pack voltage, and net discharging current recorded with timestamps for energy consumption analysis. Figure 4(b) illustrates MAP-based API integration for tracking speed, distance, and duration. Therefore, integrating MCDL with the cloud and MAP-based API ensures accurate energy consumption calculations, enabling effective analysis of traffic congestion.

3.4.2. Generating voltage data for Watt-hour (Wh) analysis

The measurement of battery node voltages BN_1, BN_2, BN_3, and BN_4 in a series-connected battery configuration is critical for assessing the health and performance of the battery pack. Using equations (1), (3), (5), and (7) the total node voltage is calculated to ensure safe operational limits. Individual battery voltages are derived through a sequential subtraction method using equations (2), (4), and (6), where each voltage. B_i is determined from the node voltages, allowing for precise monitoring of each battery's performance. This systematic approach facilitates effective battery management and enhances the overall reliability of E3W under various driving cycles while maintaining a safe discharge limit of 45.84 V at 80% depth of discharge level as per standard PbA battery manufacturer specifications.

$$BN_1 = \sum_{1}^{4} B_i \; with \; the \; condition \; that \sum_{1}^{4} B_i \leqslant 45.84 \tag{1}$$

$$B_1 = BN_1 - \sum_{i=2}^{4} B_i \tag{2}$$

$$BN_2 = \sum_{i=2}^{4} B_i \tag{3}$$

$$B_2 = BN_2 - \sum_{i=3}^{4} B_i \tag{4}$$

$$BN_3 = \sum_{i=3}^{4} B_i \tag{5}$$

$$B_3 = BN_3 - B_4 \tag{6}$$

$$B_4 = BN_4 \tag{7}$$

3.4.3. Generating current data for depth of discharge and Watt-hour (Wh) analysis

A smooth drive cycle with consistent speed or gentle accelerations and decelerations tends to discharge the battery more evenly. This leads to a moderate and predictable depth of discharge (DOD) over time for a 4 series connected 12 V,100 AH PbA battery pack used in host E3W. Similarly, rapid acceleration requires a large burst of current, leading to a faster discharge and a deeper DOD over a short period. Each rapid acceleration drains a higher amount of charge, pushing the battery to higher DOD levels faster. In traffic, even small accelerations require bursts of current, which means the battery must frequently supply a high current in short periods. Therefore ampere-hour discharge or DOD across various driving cycles of E3W is important to analyze the traffic condition. Using MCDL the time-varying discharging current I_D (t) is measured in operating period Δt using equation (8)

$$\int_{t_1}^{t_n} I_D(t)dt = \frac{\Delta t}{2}[I_D(t_1) + 2I_D(t_2) + 3I_D(t_3) + \ldots I_D(t_n)] \tag{8}$$

Where t_1 and t_n are the time taken for host E3W driving lap D_{L1} and D_{Ln} respectively under different drive cycle scenarios. The discharging depth during vehicle operating period Δt for driving lap D_{Li} …… D_{Ln} is determined through the application of equation (9)

$$\Delta D_{OD} = \frac{\int_{t0}^{t0+\Delta t} I_D(t)dt}{Q_{rated}} \times 100\% \tag{9}$$

In equation (9), the polarity of I_D is negative and indicates the discharging of the battery during the driving period under various road traffic conditions, Q_{rated} is the rated capacity of the battery under test and t_0 is the initial time of the battery discharge.

The instantaneous energy consumption (E_{wh}) under any drive cycle is calculated using equation (10)

$$E_{wh} = \sum_{1}^{4} B_i \times \int_{t_1}^{t_n} I_D(t)dt \times [T(n_i) - T(n_{i-1})] \tag{10}$$

Where, $T(n_i)$ refers to the length of each time interval i within the total measurement period.

By tracking voltage and current fluctuations during varying driving conditions, especially in smooth, moderate, and highly congested traffic, the system stores the Watt-hour data and transmits it to the cloud platform for preprocessing and analysis.

3.5. Dynamic parking recommendation based on multi-factor congestion analysis

The congestion index $C(t)$ derived from energy consumption in watt-hour/km, is represented by equation (11)

$$C(t) = \frac{E_{net}(t) - E_{avg}(t)}{E_{avg}(t)} \tag{11}$$

Where, $E_{net}(t)$ = net energy consumption for the specific time interval, $E_{avg}(t)$ = average energy consumption per km for a given drive cycle.

For N Electric Three wheelers (E3Ws), congestion probability (P_C) is calculated using equation (12)

$$P_C = \frac{\sum_{i=1}^{N} C_i(t)}{N} \tag{12}$$

Where $C_i(t)$ = congestion index for each E3W.

To account for sudden energy spike, the new congestion probability (P_{Cnew}) is represented by equation (13).

$$P_{Cnew} = P_c + \alpha \sum_{i=1}^{N} \frac{dE_i}{dt} \tag{13}$$

where, $\frac{dE_i}{dt}$ = energy consumption rate of each E3W and α = congestion sensitivity to energy changes that may be dynamic or adaptive.

Based on the new congestion probability (P_{Cnew}), the roadside parking slot is recommended for below the upper threshold (U_{TH}) and lower threshold (L_{TH}) limit. Therefore condition for parking recommendation (P_{rec}) is

$$P_{rec} = \begin{Bmatrix} smooth, & P_{Cnew} < L_{TH}\ (recommended) \\ moderate, & L_{TH} \leqslant P_{Cnew} \leqslant U_{TH}\ (waiting\ state) \\ high, & P_{Cnew} > U_{TH}\ (not\ recommended) \end{Bmatrix}$$

Both thresholds vary by region; In this study, $L_{TH} = 0.08$ and $U_{TH} = 0.15$ are applied based on localized traffic and energy consumption characteristics.

Each host E3W is connected to a MAP API and fetch real-time geo coordinates to check the availability of parking within a 500-meter radius (R). If congestion in a Region of Interest (ROI) is too high then the search radius expands dynamically (R_{new}) and represented by equation (14)

$$R_{new} = R + \beta\left(\frac{dE_{net}}{dt}\right) \tag{14}$$

Where β controls the rate at which the search radius expands in response to increasing congestion.

The parking occupancy rate (PO_r) is determined based on P_{Cnew}, L_{TH} and U_{TH} in the ROI using equation (15).

$$PO_r = \min\left(1, \max\left(0, \frac{P_{Cnew} - L_{TH}}{U_{TH} - L_{TH}}\right)\right) \tag{15}$$

This mathematical formulation establishes a robust analytical backing to enhance the accuracy and responsiveness of real-time parking decisions.

3.6. Software implementation

3.6.1. Map API integration for longitude and latitude collection

To facilitate real-time location tracking, the system integrates a Map API for collecting precise longitude and latitude data. As the host E3W vehicle traverses a specific route, the Map API captures its geographic coordinates (latitude and longitude) at 2 s intervals and stores them in a database. These coordinates are linked to the corresponding traffic congestion data, derived from watt-hour transformations of voltage, current, and time inputs during the data preprocessing steps. By associating location information with the corresponding energy consumption data, the system creates a spatially-aware dataset that forms the foundation for recommending optimal and accurate location-specific parking areas to upcoming vehicles.

3.6.2. Data preprocessing

In Machine Learning, data preprocessing is a crucial step to ensure the development of an effectively trained model. As highlighted in section 3.4, which discusses data procurement steps, all data are initially gathered on a cloud platform. After acquisition, initial pre-processing steps are performed to prepare the data for analysis. Specifically, once the Watt-hour data is retrieved, additional preprocessing is conducted to refine the dataset. These advanced preprocessing steps are detailed in the following subsections, which cover Data Normalization and Target Label Assignment, Data Grouping, and Formatting.

3.6.2.1. Data normalization and target label assignment

The analysis shows the datasets have distinct traffic congestion patterns. Moreover, each dataset defines specific upper and lower bounds for Watt-hour consumption. These bounds vary across different drive cycles, are categorized under various traffic states, and serve as features for analyzing traffic dynamics in machine learning models. To account for these variations, a scaling technique called Min-max scaling is applied to normalize all traffic density features, ensuring fair treatment, enhancing algorithmic convergence and performance, and improving resilience to outliers. For supervised learning, traffic density is manually assigned to the output column by closely analyzing real-time on-road traffic conditions to voltage, current, and timestamp values recorded from the host E3W's onboard battery. During data collection, output labels are manually assigned as 'Smooth Traffic'/'High Congestion'/'Moderate Congestion' by observing traffic conditions faced by the E3W in

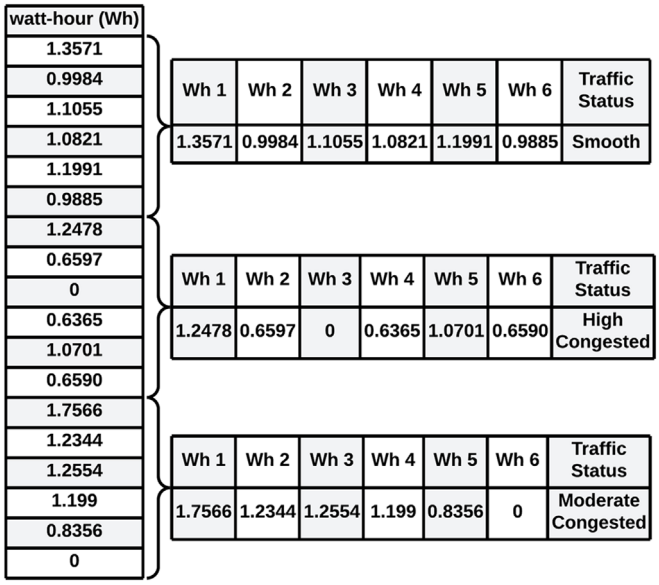

Figure 5. Six watt-hour group paired with their traffic status label.

Table 2. Optimal number of watt-hour values evaluation.

Number of consecutive Wh value	2	3	4	5	**6**	7	8	9	10	11	12
ML model accuracy score	0.7980	0.8037	0.8049	0.8114	**0.8245**	0.8112	0.7724	0.8187	0.8050	0.8056	0.8133

real-time. While recording onboard battery data (voltage, current, and timestamps), traffic conditions labels are simultaneously noted to best represent the traffic conditions experienced by the host E3W. This real-time labeling provides an accurate reflection of varying traffic states, each corresponding to specific operational states of the E3W.

3.6.2.2. Data grouping and formatting

The output labels are assigned by grouping six consecutive rows from the normalized 'watt-hour' column, along with their associated traffic congestion levels. We selected six consecutive watt-hour values as the optimal group size based on empirical evaluation. As shown in table 2, we systematically tested different group sizes, ranging from two to twelve consecutive watt-hour values, and measured their impact on ANN model accuracy. The results demonstrate that a group size of six yielded the highest classification accuracy of 82.45%, indicating its effectiveness in capturing relevant energy consumption patterns for traffic congestion prediction. This result suggests that six sequential watt-hour values provide a sufficient temporal context for the neural network to distinguish between different traffic congestion levels. Using fewer than six values may not provide enough information for the model to recognize meaningful patterns, while using more than six values does not further improve accuracy and, in some cases, even leads to a decline in performance. This diminishing accuracy beyond six values indicates potential redundancy or noise introduced by longer sequences. Thus, the dataset is structured to a group of six consecutive watt-hour values not arbitrarily but based on a systematic evaluation. This approach enables the neural network to learn from an effective watt-hour consumption sequence to predict congestion status effectively.

Figure 5 presents a structured table of watt-hour measurements taken over six-time intervals (Wh 1 to Wh 6), paired with their respective traffic density categories. The top table illustrates 'Smooth' traffic conditions, where watt-hour measurements such as 1.3571, 0.9984, and 1.1991 indicate moderate and stable energy consumption, consistent with smooth traffic. The middle table shows a 'High Congested' traffic scenario, where watt-hour values like 1.2478, 0.6597, and 1.0701 reflect significant energy consumption fluctuations associated with heavy traffic and frequent stops. The bottom table represents 'Moderate Congested' traffic conditions, with watt-hour readings such as 1.7566, 1.2344, and 0.8356, indicating varied energy usage due to intermediate congestion levels and slower vehicle movement. Since data is recorded at 2 s intervals, six consecutive watt-hour

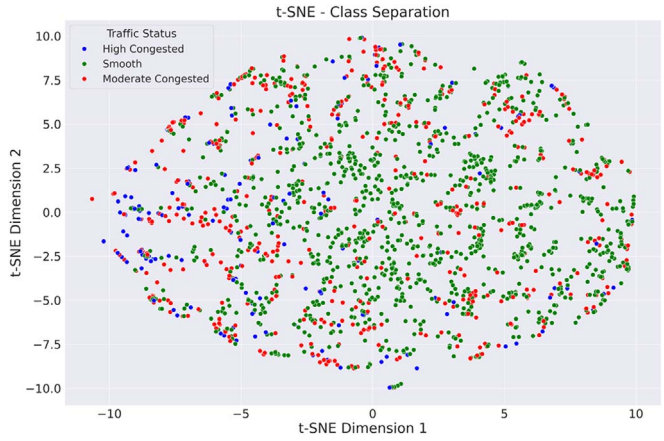

Figure 6. t-SNE plot of six consecutive Watt-hour features.

values represent a 12 s window. Each window is assigned a specific output label either 'Smooth Traffic,' 'High Congestion,' or 'Moderate Congestion' reflecting the traffic conditions experienced by the E3W over the sampling interval of 12 s. This approach captures real-time traffic patterns, helping the model accurately predict road conditions based on recent energy consumption data.

3.6.3. Data visualization and model selection

In this section, a primary visualization technique is used to select the best-fit model for training and real-time traffic status forecasting. Figure 6 represents the t-SNE plot of six consecutive Watt-hour features. The outcome of the plot illustrates the considerable overlap between the output classes and complex nonlinear distribution. Therefore, suggesting that traditional machine learning techniques such as logistic regression, decision trees, or even SVM might struggle to achieve higher classification accuracy for real-time traffic status prediction. These methods typically excel when most of the data points are linearly separable and patterns can be captured with simpler decision boundaries. However, according to the t-SNE plot, the dense intermixing of classes in the dataset suggests that relationships among watt-hour values and the corresponding traffic density categories are highly nonlinear, making it challenging for traditional algorithms to effectively separate the classes. Deep learning, particularly ANN, is highly suited for tasks involving complex, nonlinear relationships due to its multi-layered architecture that enables learning intricate patterns within the data. In this study, we apply an ANN to predict traffic status based on six consecutive watt-hour features, leveraging the model's ability to capture stable dependencies and interactions within these readings for traffic status forecasting.

3.6.4. ANN model development

ANNs are deep learning techniques inspired by the structure and functioning of the human brain [31]. ANNs are highly effective in learning from non-linear datasets, often outperforming traditional machine learning techniques. A typical ANN consists of at least three layers: an input layer, one or more hidden layers, and an output layer. During forward propagation, the input data in each layer is multiplied by an initial weight value, added to a bias of the respective layer, and routed through an activation function such as ReLU, tanh, softmax etc During backpropagation, these weights are adjusted iteratively to minimize the error between predicted and actual outputs.

The general mathematical formulation of the ANN output can be expressed as equation (16)

$$\hat{y} = \varnothing_1\left[b_0 + \sum_{k=1}^{k=m}\left\{w_k \times \varnothing_0\left(b_k + \sum_{i=1}^{i=n} w_{ik}x_i\right)\right\}\right] \tag{16}$$

Where $\hat{y}$ denotes model predicted output.

$\varnothing_0$, $\varnothing_1$ = Activation functions.

b_0, b_k = bias for hidden and output layer, respectively.

w_k = weight between the k^{th} hidden layer neuron and output.

w_{ik} = weight between the i^{th} input and k^{th} hidden layer neuron.

x_i = i^{th} input feature

m = number of neuron in hidden layer

n = number of input feature

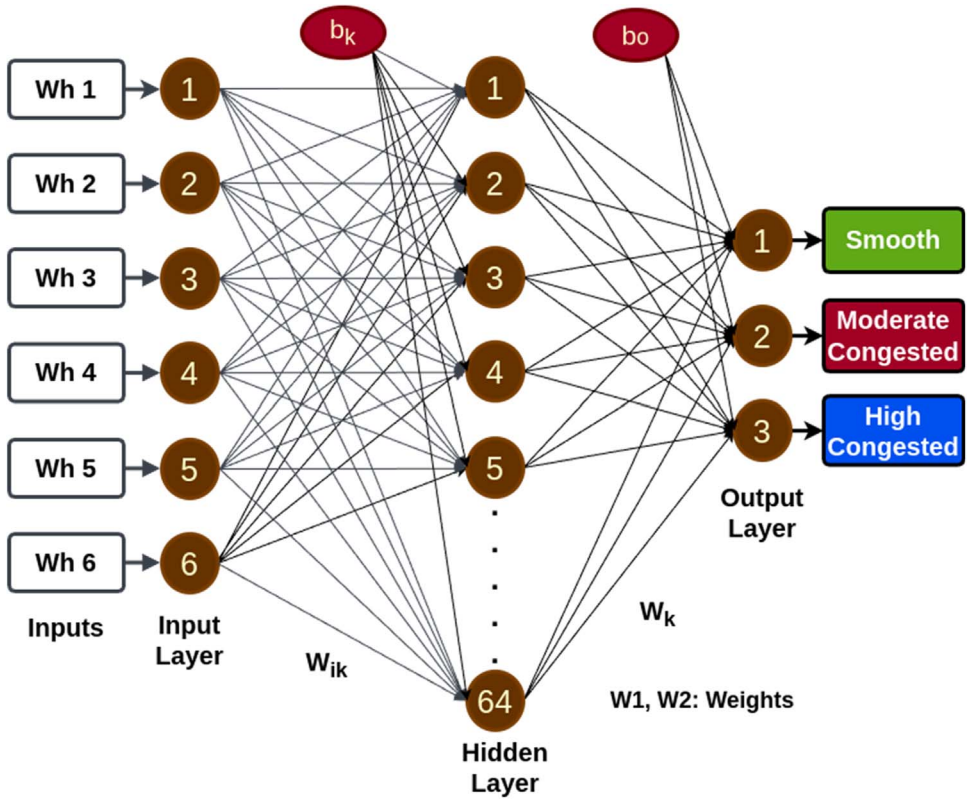

Figure 7. Proposed ANN architecture for traffic status classification.

Table 3. Output shape and total parameter count at each layer of the proposed ANN model.

Layer (type)	Output shape	Parameter
dense_2 (Dense)	(None, 64)	448
batch_ normalization (Batch Normalization)	(None, 64)	256
dense_1(Dense)	(None, 3)	195

Figure 7 illustrates the proposed ANN architecture, developed based on the characteristics of the self-collected dataset. The architecture is designed with an appropriate number of neurons at each layer. The input layer consists of six neurons corresponding to six consecutive watt-hour features (Wh1, Wh2, …, Wh6), the hidden layer contains 64 neurons, and the output layer includes three neurons representing the classification categories (Smooth, Moderate Congested, and High Congested). At each layer, specific weights (w_{ik} in the hidden layer and w_k in the output layer) are applied to the inputs, added to the bias (b_k in the hidden layer and b_0 in the output layer), and passed through activation functions ($\varnothing_0$ in the hidden layer and $\varnothing_1$ in the output layer). The ANN architecture, also summarized in table 3, comprises two dense layers and one batch normalization layer. The hidden layer, with 64 neurons and 448 trainable parameters, captures intricate feature representations, while the batch normalization layer, with 256 parameters, ensures stable learning by normalizing intermediate outputs. The final dense layer, containing 195 parameters, maps the learned features to the three output classes, enabling effective classification of traffic status.

The model applies activation functions, including ReLU in the hidden layer and Softmax in the output layer, to introduce non-linearity and enable the learning of complex patterns. It is compiled using the categorical crossentropy loss function, which is well-suited for multi-class classification, and the Adam optimizer with a learning rate of 0.001 is used, which is tested and selected for its efficient convergence. To prevent overfitting and enhance training efficiency, an early stopping mechanism is implemented, monitoring validation loss and restoring the best weights after 20 epochs of no improvement.

Table 4. Memory distribution of trainable and non-trainable parameters.

Parameter type	Count	Memory size
Total Parameters	899	3.51 kilobyte
Trainable Parameters	771	3.01 kilobyte
Non-trainable Parameters	128	512 byte

In this study, all initial weight values are assigned using the He Normal Initialization technique. This approach ensures, weights are drawn from a normal distribution with a mean of 0 and a standard deviation (σ) calculated using equation (17)

$$\sigma = \sqrt{\frac{2}{fan\_in}} \tag{17}$$

Here, the fan represents the number of input connections to a neuron. This initialization technique ensures efficient weight scaling, and faster convergence, preventing gradient vanishing or exploding issues during training. Table 4 highlights the memory distribution of trainable and non-trainable parameters across the model. With a total of 899 parameters, including 771 trainable and 128 non-trainable parameters. The model is lightweight, consuming only 3.51 kilobytes of memory.

Trainable parameters, including weights and biases, are updated during training, while non-trainable parameters (from the batch normalization layer) maintain inference-time statistics to ensure stability. This compact yet robust design, coupled with the integration of batch normalization and early stopping, makes the model well-suited for small-scale multi-class classification tasks, ensuring computational efficiency and reliable performance on unseen data. The proposed ANN model classifies traffic status into three categories: Smooth, Moderate Congestion, and High Congestion. Based on the classification results, the system is designed to provide dynamic parking recommendations, aimed at alleviating roadside congestion by suggesting optimal parking spots depending on the current traffic conditions.

4. Experimental result and discussion

The experimental results and discussion are organized into six sections: the first outlines the experimental design; the second focuses on feature selection process for the ANN model; the third covers the training and validation of the ANN model; the fourth presents performance characteristics; the fifth discusses model errors and limitations; and the sixth provides visual illustrations of road traffic conditions along with corresponding roadside parking recommendations.

4.1. Experimental design

The experimental design involved deploying the MCDL system across 70 E3Ws, each fitted with 4.8 kWh lead-acid battery packs (4 × 12 V, 100 Ah, C20) that powered 900 W, 48 V BLDC motors. These vehicles, with a range of 80 km and speeds of up to 27 km h^{-1}, were monitored over a period of 90 days in a 60 km^2 area of southeastern West Bengal, India. Real-time data, including timestamps, speed, voltage, current, and GPS location was captured at a frequency of 0.5 Hz and transmitted to a cloud platform. Drive cycles ranged from 10 to 210 min, covering various suburban and rural traffic conditions. A Map API tracked geo-location every 2 s, associating spatial coordinates with watt-hour-based energy data for congestion and performance analysis. A dataset of 15,000 manually labeled records was created, with 70% used for training, 20% for testing, and 10% for validation, enabling the development and evaluation of machine learning models for predicting energy consumption and analyzing drive patterns.

4.2. Experimentation of ANN model feature selection

Table 5 illustrates the experimentation of ANN model features selection method for precise traffic status classification. The real-time experimentation is conducted in a discrete time frame to monitor the traffic congestion status. The energy consumption (Watt-hour), battery pack mean voltage (Mean Volt), DOD and time frame under different traffic status is directly fetched from MCDL. While the distance (km) traveled by E3W under a specific time frame is recorded from a GPS speedometer. The energy consumption rate (Wh/km) is determined as the ratio of watt-hour consumption over a specific time frame to the distance traveled (in kilometers) by the E3W during that same period. From the data set, it has been observed that under highly congested traffic, Wh/km values are generally higher due to frequent stops and accelerations, leading to increased energy consumption per kilometer (S. No 1 to S. No 6). In smooth traffic, Wh/km values remain

Table 5. Features evaluation of ANN model for traffic status classification.

S.no	Traffic status	Watt-hour (Wh)	Distance (km)	Wh/km	Mean volt	DOD	Time frame(Sec)
1	High Congested	21.56	0.394	54.72	49.32	0.413	161
2	High Congested	50	0.657	76.1	49.58	1.01	380
3	High Congested	45	0.743	60.57	49.42	0.917	299
4	High Congested	55.66	0.654	85.11	47.56	1.17	258
5	High Congested	153.73	2.76	55.7	46.64	3.06	560
6	High Congested	45.1486	0.675	66.88	46.07	0.98	188
7	Smooth	39.8	0.771	51.63	48.97	0.813	195
8	Smooth	144.32	2.79	51.72	48.96	2.96	533
9	Smooth	52	1	52	48.74	1.06	211
10	Smooth	24.34	0.47	51.77	48.68	0.5	92
11	Smooth	473.58	9.16	51.69	48.08	9.85	1706
12	Smooth	380	7.36	51.63	47.54	8.009	1385
13	Smooth	138.12	2.67	51.71	46.04	3	437
14	Smooth	157.1048	3	52.37	45.67	3.44	489
15	Moderate Congested	124.34	2.04	60.96	48.86	2.04	485
16	Moderate Congested	46.69	0.9	51.88	49.68	0.94	378
17	Moderate Congested	16.78	0.318	52.76	48.56	0.339	71
18	Moderate Congested	14.49	0.28	51.75	48.79	0.297	73
19	Moderate Congested	106.81	2.06	51.85	47.07	2.27	391
20	Moderate Congested	26.18	0.497	52.68	46.78	0.539	169

relatively steady around 51.7, showing efficient energy use per kilometer due to sustained, continuous driving (S. No 7 to S. No 14). In moderate congested traffic, Wh/km values are slightly higher than in smooth traffic but generally lower than in high congested traffic (S. No 15 to S. No 20). The energy consumption of a battery pack, measured in Watt-hours (Wh), is directly related to the DOD as discussed in sub-section 3.4.2. Therefore Watt-hour data, the driving time, and the corresponding DOD created a significant correlation for congestion classification. A short travel time (380 s) but higher DOD (1.01) suggests more frequent stops and starts, causing increased strain on the battery over a short period (S. No 2). Even in a high congested traffic (S. No 4) a shorter time (258 s), the DOD(1.17) is comparable to smooth traffic's travel times (S. No 9), indicating that congested conditions demand more energy due to frequent acceleration bursts. In contrast for smooth traffic conditions (S.No 11) long travel time(1706 s) with high DOD(9.85), reflecting consistent battery power usage without rapid changes in speed and minimal stops. In the case of moderate congested traffic (S. No 15) moderate DOD (2.04) for an intermediate travel time(485 s), indicating partial acceleration with some periods of sustained driving. A travel time (169 s) with a low DOD (0.539) value of moderately congested traffic (S. No 20), reflecting minimal discharge due to lower energy demand compared to smooth traffic conditions (S. No 7).

The outcome of the above analysis revealed that the Watt-hour feature is highly correlated with traffic status compared to DOD features, providing distinct patterns for different traffic conditions and simplifying the classification process. Therefore, Watt-hour is most relevant to train the proposed ANN model for precise traffic status forecasting and optimal parking recommendation.

4.3. ANN model training and validation analysis

Based on the performance evaluation parameters such as test accuracy, validation loss, Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R^2 the comparative analysis of the proposed ANN model configuration is conducted. During the training process, the Sparse Categorical Crossentropy (SCC) loss function was employed, which is particularly suitable for multi-class classification problems. This loss function measures the difference between the true class labels and the predicted class probabilities and represented by equation (18).

$$SCC\ Loss = -\sum_{i=1}^{N} Y_i \log(\hat{Y}_i) \tag{18}$$

Where N = Total number of instances in the batch, Y_i = True class label for the i^{th} instance , $\hat{Y}_i$ = Predicted probability for the i^{th} instance.

To verify that the model is not over fitting to the training dataset and can effectively generalize to real-world vehicle dynamics data, test accuracy is evaluated. This evaluation demonstrates the reliability of the model for deployment in predicting various traffic states. The test accuracy is represented by equation (19).

$$Test\ Accuracy = \frac{\sum_{i=1}^{N} \mathbb{I}(Y_i = \widehat{Y_i})}{N} \times 100 \tag{19}$$

Where, $\hat{Y}_i$ = predicted class label for the ith instances, $\mathbb{I}(Y_i = \widehat{Y_i})$ = indicator function, which is 1 if $Y_i = \hat{Y}_i$ and 0 otherwise.

To ensure that the model maintains its ability to generalize to unseen data in different traffic status, rather than simply memorizing the training data, validation loss testing is conducted. Validation loss quantifies the model's ability to generalize by measuring prediction errors on the unseen validation set. It is calculated using the SCC loss function applied to the validation set after each epoch during training. A lower validation loss indicates better generalization, meaning the model is more capable of accurately predicting unseen data.

Mean Absolute Error (MAE) is used to calculate the average difference between predicted and actual values, making it easy to assess prediction accuracy. This is important in traffic prediction, as it provides a clear, actionable metric. For E3W vehicle dynamics, MAE helps measure how closely traffic conditions are predicted. It is also less affected by outliers, so occasional sensor errors (such as glitches in voltage or current readings) do not significantly impact performance. The MAE is represented by equation (20).

$$Mean\ Absolute\ Error = \frac{1}{N}\sum_{i=1}^{N}|Y_i - \hat{Y}_i| \tag{20}$$

Where $|Y_i - \hat{Y}_i|$ = Absolute error for the ith instance, representing the magnitude of the difference between the true and predicted values.

In contrast to MAE, Mean Squared Error (MSE) is particularly useful in traffic prediction when large deviations, such as significant traffic congestion or sudden changes in vehicle dynamics require more attention. MSE ensures that the model focuses on correcting larger errors, which is critical in high-traffic areas where accurate predictions are essential for real-time decision-making, such as roadside parking recommendations. The expression for MSE is given below as equation (21).

$$Mean\ Squared\ Error = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \tag{21}$$

RMSE would be considered more meaningful in the proposed system using MCDL, where traffic predictions mainly rely on real-world measurements (e.g., node voltage, net discharging current, energy consumption). RMSE allows the prediction error to be directly compared with the actual values, making the results easier to interpret and apply in practical scenarios, such as roadside parking recommendations and optimization of vehicle dynamics in different traffic conditions. The expression for RMSE is given below as equation (22).

$$Root\ Mean\ Squared\ Error = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2} \tag{22}$$

R^2 is especially useful when comparing multiple model configuration. If different models are tested for traffic prediction based on energy consumption the one with the highest R^2 can be considered the best at explaining the variance in the data, making it more reliable for real-time predictions. R^2 score helps in assessing how much of the variability in the target variable is explained by the model compared to a simple mean-based model. A high R^2 score indicates that the model is good at making predictions. However, in the case of proposed parking recommendation based on traffic status prediction, a higher R^2 means that the model is effectively predicting real-world traffic conditions, which is depends on vehicle performance and provide optimized parking recommendations in various traffic states. The nature of the data is one of the important factor for evaluation of R^2 score. In complex, real-world datasets like electric vehicle dynamics where multiple external factors often involve a lot of noise that are difficult to model perfectly. The R^2 score is represented by equation (23).

$$R^2 Score = 1 - \frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{N}(Y_i - \bar{Y}_i)^2} \tag{23}$$

Where $\bar{Y}_i$ = Mean of the true class label for the ith instances.

The chosen performance metrics comprehensively validate the model's effectiveness for real-world deployment. Test accuracy and validation loss confirm its ability to generalize beyond training data, preventing overfitting and ensuring adaptability to dynamic traffic conditions. Table 6 represents the significance of MAE with other evaluation metrics for sensitive error calculations. With reference to the above table, Mean Absolute Error (MAE) = $(0 + 1 + 1 + 0 + 2)/5 = 0.8$, meaning the ANN's average prediction is 0.8 congestion levels off. This reflects how far, not just whether the predicted traffic state deviates from the true label. This is especially important in real-time vehicle dynamics applications where misclassifying 'High' as 'Smooth' (Sample E) can have operational impacts that are rightly captured by a high 'absolute error 2'. Moreover, misclassifying 'High' as

Table 6. Sample data points illustrating the relevance of evaluation metrics.

Sample	True class	Predicted class	Absolute error
A	0(Smooth)	0(Smooth)	0
B	1(Moderate)	2(High)	1
C	2(High)	1(Moderate)	1
D	1(Moderate)	1(Moderate)	0
E	2(High)	0(Smooth)	2

Table 7. ANN model performance parameters.

S.no	Loss function	Learning algorithm	No of neurons	Test accuracy	Validation loss	MAE	MSE	RMSE	R^2
1	SCC	Adam	4	0.7967	0.5702	0.2823	0.5072	0.7122	0.4669
2	SCC	SGD	4	0.7967	0.5617	0.2799	0.4928	0.7020	0.4820
3	SCC	RMSProp	4	0.8002	0.5735	0.2703	0.4665	0.6830	0.5097
4	SCC	Adam	8	0.8026	0.5081	0.2667	0.4533	0.6733	0.5235
5	SCC	SGD	8	0.7955	0.5793	0.2763	0.4773	0.6908	0.4983
6	SCC	RMSProp	8	0.7907	0.5758	0.2763	0.4581	0.6769	0.5185
7	SCC	Adam	16	0.8038	0.4899	0.2644	0.4486	0.6697	0.5285
8	SCC	SGD	16	0.7990	0.5205	0.2751	0.4785	0.6917	0.4971
9	SCC	RMSProp	16	0.8134	0.4973	0.2620	0.4677	0.6839	0.5084
10	SCC	Adam	32	0.8002	0.5438	0.2703	0.4545	0.6742	0.5222
11	SCC	SGD	32	0.7955	0.5663	0.2787	0.4844	0.6960	0.4908
12	SCC	RMSProp	32	0.8086	0.5486	0.2536	0.4115	0.6415	0.5675
13	**SCC**	**Adam**	**64**	**0.8242**	**0.5116**	**0.2500**	**0.4294**	**0.6553**	**0.5486**
14	SCC	SGD	64	0.8038	0.5331	0.2584	0.4211	0.6489	0.5574
15	SCC	RMSProp	64	0.8086	0.5442	0.2572	0.4342	0.6589	0.5436
16	SCC	Adam	128	0.8134	0.5354	0.2632	0.4761	0.6900	0.4996
17	SCC	SGD	128	0.8062	0.5339	0.2703	0.4833	0.6952	0.4921
18	SCC	RMSProp	128	0.8002	0.5133	0.2739	0.4725	0.6874	0.5034

'Moderate' (Sample C) can lead to the least significant operational impact and perfectly captured by comparatively low 'absolute error 1'.

Since the ANN model uses real-time energy data from voltage, current, and time sensors and the cost of misclassifying congestion levels depends on the closeness of prediction, metrics like MAE, MSE, RMSE, and R^2 are more useful for evaluating model performance in this case.

The results from the experiments across the three tables indicate that the configuration using the Adam optimizer with a single hidden layer and 64 neurons is the most optimal choice among all tested models. The experiments in table 7 are conducted not only for model configuration selection but also for learning algorithm (optimizer) selection, based on the six evaluation parameters (test accuracy, validation loss, MAE, MSE, RMSE, and R^2 score) on test data. Since the Adam optimizer outperformed compared to SGD and RMSProp in terms of these key metrics illustrated in table 7 (S.No 13), further experiments are conducted using the Adam optimizer for model configuration selection. In table 7, this configuration achieved the highest overall test accuracy of 82.42% among single-layer models, with a competitive validation loss (0.5116) and robust performance metrics such as MAE (0.2500), MSE (0.4294), RMSE (0.6553), and R^2 (0.5486). Furthermore, the experiment is conducted based on the Adam optimizer with two hidden layers to analyze the model performance while varying the number of neurons in each layer.

In table 8, the best-performing model configuration is observed with two hidden layers and 8 neurons in each layer (S.NO 8), achieving almost similar test accuracy (81.70%) with validation loss (0.5206), but the added complexity of two layers increases memory usage and computational cost, making it less efficient.

Similarly, the experiment is conducted based on the Adam optimizer with three hidden layers to analyze the model performance while varying the number of neurons in each layer. In table 9 (S. No 12), the configuration with three hidden layers (16, 8, and 16 neurons) achieved the highest test accuracy (82.18%), but showing a higher validation loss (0.5502). This model configuration increased the model complexity, leading to higher computational requirements and potential over fitting. In contrast, the Adam, 64 (single hidden layer) configuration strikes the best balance between performance, generalization, and efficiency. Its single-layer architecture minimizes model complexity and memory usage while maintaining high accuracy and reliable error

Table 8. ANN model performance parameters utilizing two hidden layers.

S. no	Loss function	Learning algorithm	Hidden_1 Neurons	Hidden_2 Neurons	Test accuracy	Validation loss	MAE	MSE	RMSE	R^2
1	SCC	Adam	4	128	0.7895	0.5355	0.2967	0.5287	0.7271	0.4443
2	SCC	Adam	8	64	0.8146	0.5219	0.2787	0.5371	0.7329	0.4355
3	SCC	Adam	16	32	0.8002	0.5580	0.2751	0.4689	0.6848	0.5071
4	SCC	Adam	32	16	0.8050	0.5635	0.2739	0.4892	0.6995	0.4858
5	SCC	Adam	64	8	0.8002	0.5508	0.2871	0.5167	0.7189	0.4569
6	SCC	Adam	128	4	0.8086	0.5828	0.2560	0.4139	0.6433	0.5650
7	SCC	Adam	4	4	0.8050	0.5223	0.2691	0.4773	0.6908	0.4983
8	SCC	Adam	8	8	0.8170	0.5206	0.2512	0.4330	0.6580	0.5449
9	SCC	Adam	16	16	0.8074	0.5398	0.2620	0.4462	0.6680	0.5310
10	SCC	Adam	32	32	0.8122	0.5701	0.2679	0.4880	0.6986	0.4870
11	SCC	Adam	64	64	0.8038	0.5993	0.2691	0.4581	0.6769	0.5185
12	SCC	Adam	128	128	0.8086	0.6415	0.2763	0.4988	0.7063	0.4757
13	SCC	Adam	32	4	0.8122	0.5700	0.2584	0.4426	0.6653	0.5348
14	SCC	Adam	4	8	0.8026	0.5344	0.2739	0.4916	0.7012	0.4833
15	SCC	Adam	16	4	0.8086	0.5119	0.2512	0.4043	0.6359	0.5750
16	SCC	Adam	64	32	0.8086	0.6196	0.2715	0.4964	0.7046	0.4782
17	SCC	Adam	16	8	0.8026	0.4999	0.2811	0.5084	0.7130	0.4657
18	SCC	Adam	8	4	0.8110	0.4961	0.2584	0.4426	0.6653	0.5348

metrics. The Adam optimizer further enhances performance by effectively adapting learning rates and utilizing momentum, ensuring faster convergence and stable results. Therefore, the Adam, 64 configuration emerges as the most practical and optimal choice for this work, providing excellent performance with minimal computational overhead.

4.4. Performance characteristics of selected ANN configurations

Figure 8 depicts the training accuracy versus validation accuracy plot and figure 9 illustrates the training loss versus validation loss plot during the ANN model training with the selected configuration (Adam optimizer with a single hidden layer of 64 neurons). The plots demonstrate a steady improvement in accuracy and a significant reduction in loss as the epoch progresses, indicating that the model effectively learns from the training data. Importantly, the training and validation curves exhibit similar patterns, with no significant divergence, which signifies good generalization to unseen data and minimizes the risk of overfitting.

The loss curve shows a rapid initial decline, stabilizing toward the later epochs, while the accuracy curve steadily increases, reaching a plateau as the model converges. This behavior suggests that the model is not only capturing the energy consumption patterns effectively but also achieving a balance between bias and variance. These results strongly validate the effectiveness of the proposed ANN model configuration in predicting traffic status and providing parking recommendations on new, unseen energy consumption data.

4.5. Model errors and limitations

The primary cause of misclassification in the proposed ANN model is the narrow difference in energy consumption and voltage variation between smooth and moderate traffic conditions. This results in overlapping data patterns, sometimes making it difficult for the model to distinguish between the two classes. The energy consumption difference between smooth (51.82 Wh km^{-1}) and moderate congestion (53.65 Wh km^{-1}) is only 1.83 Wh km^{-1}, while the voltage standard deviation (STDEV) values are 1.01 and 1.13, respectively, making class distinction challenging. Additionally, external factors such as driving behavior, road gradient, and vehicle load significantly impact classification accuracy. Aggressive acceleration and braking during a smooth drive can temporarily increase energy consumption, resembling moderate congestion patterns. Road gradients also play a crucial role where uphill driving increases energy usage, while downhill slopes reduce it, leading to potential misclassification. Similarly, increased vehicle load raises Wh/km, making a smooth condition appear moderate. These external influences introduce variability in energy and voltage patterns, causing occasional misclassification, especially in borderline cases where feature values between classes are closely aligned.

4.6. Map-based visual illustration

Real-time experimentation for traffic status prediction and corresponding roadside parking recommendations is conducted in two different regions of West Bengal, India. The first experiment is carried out in Region A, Diamond Harbour, a suburban area with the geographical coordinates 22°11′33.3‘N, 88°11′21.8‘E. To ensure a rigorous validation process, a secondary experiment is conducted in Region B, Sarisha with the geographical

Table 9. ANN model performance parameters utilizing three hidden layers.

S.no	Loss function	Learning algorithm	Hidden_1 Neurons	Hidden_2 Neurons	Hidden_3 Neurons	Test accuracy	Validation loss	MAE	MSE	RMSE	R^2
1	SCC	Adam	4	4	4	0.8050	0.5191	0.2560	0.4115	0.6415	0.5675
2	SCC	Adam	4	8	16	0.8086	0.5313	0.2572	0.4318	0.6571	0.5461
3	SCC	Adam	4	16	32	0.8086	0.5413	0.2632	0.4617	0.6795	0.5147
4	SCC	Adam	4	32	64	0.8062	0.5835	0.2703	0.4785	0.6917	0.4971
5	SCC	Adam	4	64	128	0.8014	0.9066	0.2727	0.4761	0.6900	0.4996
6	SCC	Adam	8	8	8	0.8086	0.5058	0.2703	0.4833	0.6952	0.4921
7	SCC	Adam	8	8	16	0.8098	0.5177	0.2536	0.4139	0.6433	0.5650
8	SCC	Adam	8	16	32	0.8086	0.5547	0.2739	0.4964	0.7046	0.4782
9	SCC	Adam	8	32	64	0.7990	0.5385	0.2943	0.5359	0.7320	0.4367
10	SCC	Adam	8	64	128	0.8086	0.5693	0.2859	0.5467	0.7394	0.4254
11	SCC	Adam	16	16	16	0.8014	0.5383	0.2787	0.4844	0.6960	0.4908
12	SCC	Adam	16	8	16	0.8218	0.5502	0.2536	0.4522	0.6724	0.5247
13	SCC	Adam	16	16	32	0.8218	0.5502	0.2560	0.4617	0.6795	0.5147
14	SCC	Adam	16	32	64	0.8158	0.7615	0.2703	0.5024	0.7088	0.4719
15	SCC	Adam	16	64	128	0.7907	0.5933	0.2799	0.4593	0.6777	0.5172
16	SCC	Adam	32	32	32	0.7907	0.6410	0.2847	0.4761	0.6900	0.4996
17	SCC	Adam	32	8	16	0.8026	0.5424	0.2679	0.4450	0.6671	0.5323
18	SCC	Adam	32	16	32	0.8158	0.5622	0.2656	0.4833	0.6952	0.4921
19	SCC	Adam	32	32	64	0.8014	0.6059	0.2787	0.4916	0.7012	0.4833
20	SCC	Adam	32	64	128	0.8182	0.5838	0.2500	0.4246	0.6516	0.5537
21	SCC	Adam	64	64	64	0.8146	0.6597	0.2596	0.4581	0.6769	0.5185
22	SCC	Adam	64	8	16	0.8062	0.5467	0.2811	0.5060	0.7113	0.4682
23	SCC	Adam	64	16	32	0.8218	0.7959	0.2548	0.4557	0.6751	0.5210
24	SCC	Adam	64	32	64	0.8122	0.6623	0.2715	0.4964	0.7046	0.4782
25	SCC	Adam	64	64	128	0.7943	0.6181	0.2823	0.4713	0.6865	0.5046
26	SCC	Adam	128	128	128	0.8074	1.3583	0.2811	0.5156	0.7180	0.4581
27	SCC	Adam	128	8	16	0.8110	0.5793	0.2739	0.5012	0.7080	0.4732
28	SCC	Adam	128	16	32	0.7931	0.5810	0.2835	0.4892	0.6995	0.4858
29	SCC	Adam	128	32	64	0.7943	0.5999	0.2775	0.4593	0.6777	0.5172
30	SCC	Adam	128	64	128	0.8026	0.7123	0.2847	0.5167	0.7189	0.4569

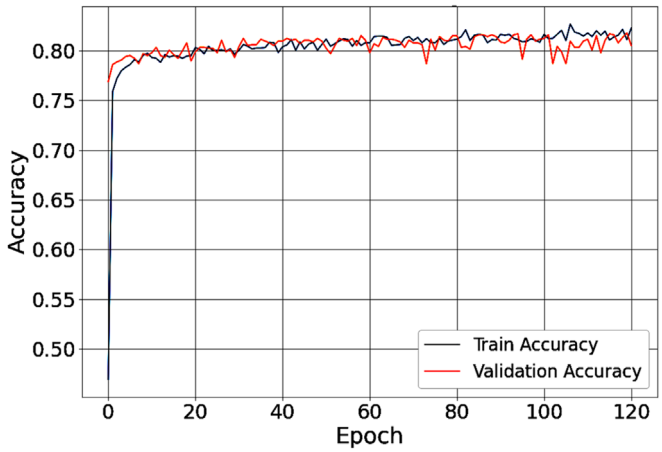

Figure 8. Training accuracy versus validation accuracy plot.

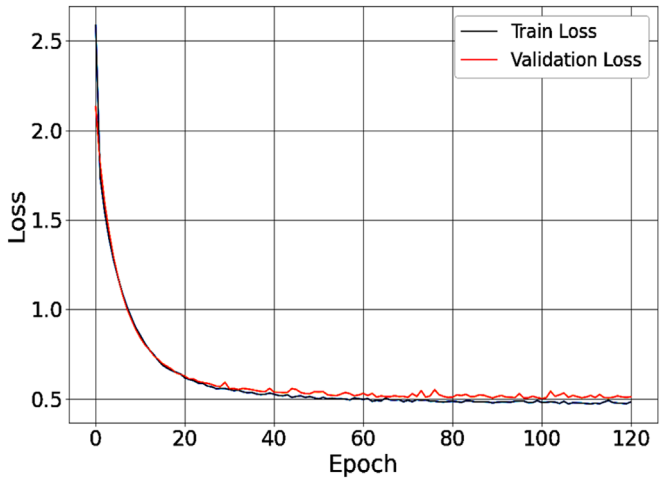

Figure 9. Training loss versus validation loss plot.

coordinates 22°15′19‘N, 88°11′24‘E. Region B featured a distinct route and traffic pattern compared to Region A. Figure 10 represents the road traffic status monitoring of E3W1 and the corresponding roadside parking recommendations for Region A.

The observations are detailed as follows:

- **Real-time traffic monitoring and congestion classification**

  Position P1 represents the starting point of E3W1 in real-time. Using a data logger, the real-time energy consumption pattern of E3W1 is continuously monitored, providing dynamic input to the proposed ANN model. Based on the traffic patterns identified, the ANN model classifies traffic conditions into three categories, Green: Smooth traffic movement with minimal energy consumption, Red: Moderate congestion, leading to increased energy consumption and Blue: High congestion zones, where energy consumption is significantly elevated. At P1 the ANN model identifies the road as moderately congested, marked in red, indicating a zone with higher-than-average energy consumption and reduced operational efficiency. This classification forms the basis for subsequent parking recommendations and route optimization strategies.

- **Parking availability analysis within radius**

  A circular zone with a 500-meter radius is marked around position P_2, representing the operational range of E3W1 for locating available parking spots. Within this radius, the ANN model identified one roadside parking spots, marked in green, as suitable and recommended for E3W1. Other parking spots within this radius are categorized as waiting state and non-recommended spots. Waiting state spots marked in red to indicate occupied or non-accessible parking for a certain time. While Non-recommended spots marked in

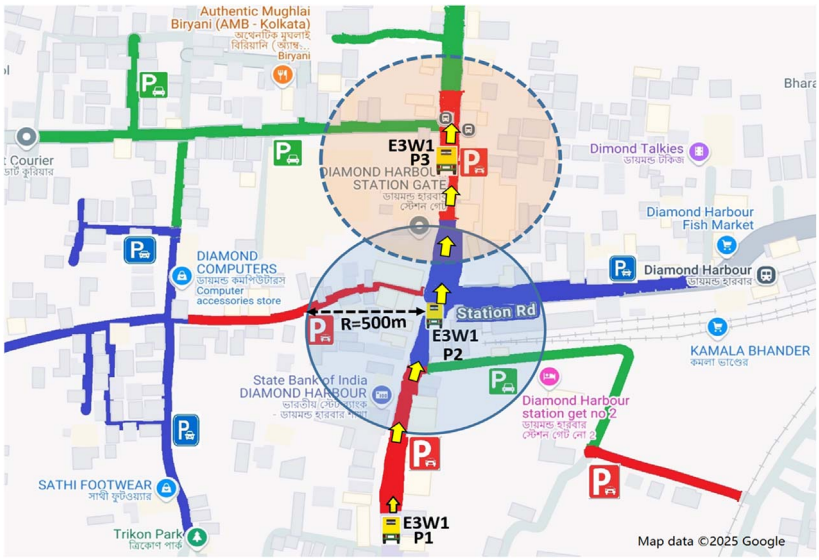

Figure 10. Map showing parking spots, pathways, and coverage areas (500 m radius) with designated journey points (P1, P2, and P3) of E3W1 and directional flow indicators in Region A. Using an ANN model, road traffic status is predicted to recommend roadside parking. Blue, red, and green paths represent high, moderate, and smooth congestion respectively [32]. Map data ©2025 Google.

blue, representing areas excluded due to constraints such as high energy consumption or operational inefficiencies due to high traffic.

- **Relative position analysis at P3**

A dotted circle marks the relative position P_3 of E3W1 with respect to P_2. This position represents the vehicle's subsequent movement influenced by ANN-guided recommendations. At P_3 E3W1 receives a parking recommendation for a spots located along the left-side collector roadway, highlighted in green. This recommendation aligns with the energy-efficient and congestion-aware framework of the ANN model.

The ANN model demonstrated its effectiveness in providing targeted parking recommendations within the operational radius of E3W1. Despite the availability of multiple parking options, the model prioritized energy efficiency, accessibility, and congestion metrics to ensure the optimal selection of a single recommended spots. At P_2 and P_3, the model successfully identified and recommended a parking option that minimized operational energy consumption while adhering to traffic conditions.

Figure 11 illustrates the E3W2 performance under similar congestion conditions but validates the model under potentially different road geometries and parking dynamics. In figure 11, the ANN model identifies two recommended (green) parking spots for E3W2. However, the location of the recommended spots differs slightly due to changes in the distribution of other spots, indicating variability in operational conditions. A similar distribution of not recommended (blue) spots is observed in figure 11. However, the number and spatial arrangement of these spots differ slightly, reflecting the ANN model's adaptability to different traffic and parking scenarios. In figure 11, the number of waiting state (red) spots appears slightly higher than in figure 10, suggesting a more waiting parking environment within the operational radius of E3W2. This validates the ANN model's capability to manage varying levels of parking recommendation based on the degree of road traffic status.

The comparison reveals that the ANN model consistently delivers accurate traffic classification based on real-time traffic status for energy-efficient parking recommendations across both scenarios. While the operational conditions for E3W1 and E3W2 are similar, the parking recommendation outcomes differ slightly due to variations in traffic patterns and parking availability in the respective zones. This demonstrates the ANN model's adaptability and reliability in diverse real-world scenarios.

5. Conclusion and future work

This paper presents a comprehensive framework for the development of a smart roadside parking recommendation system tailored for electric three-wheelers (E3Ws) operating on suburban roads. The proposed methodology integrates an IoT-enabled multichannel data logger to collect real-time vehicle dynamics data, including voltage, current, and time stamps. This data is processed using a cloud server, and the results are

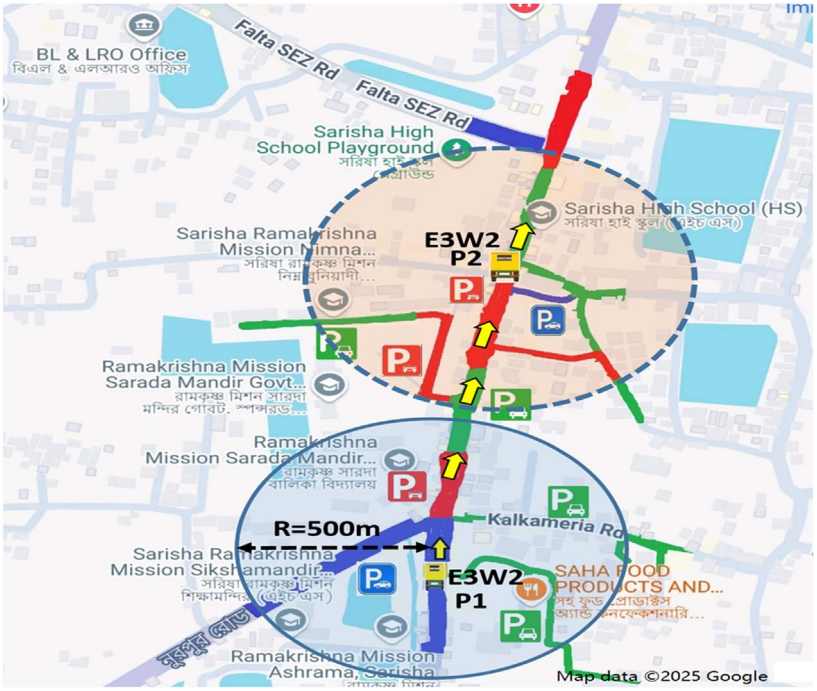

Figure 11. Map depicting parking zones, pathways, and coverage areas (500 m radius) with designated journey points (P1, P2) of E3W2 and directional flow indicators in Region B [33]. Map data ©2025 Google.

displayed on a map-driven interface for real-time traffic status visualization and parking recommendations. The framework employs an Artificial Neural Network (ANN) optimized through systematic adjustments in its configuration, such as layers and neurons, to achieve the best model performance. Key performance metrics are meticulously analyzed to ensure robust traffic classification. Additionally, the system innovatively uses real-time vehicle data to predict traffic conditions, analyze energy consumption patterns, and provide roadside parking recommendations for E3Ws. The developed system demonstrates its effectiveness by optimizing parking solutions to improve time and energy efficiency and addressing the unique challenges associated with suburban collector roads.

Future research will focus on expanding the applicability of the system to include all types of electric vehicles across diverse geographical regions. This will involve refining the ANN model and incorporating advanced features to further enhance the accuracy and scalability of traffic status prediction and parking recommendations. The potential for integrating additional real-time data sources, such as environmental conditions and driver behavior, will also be explored to provide more comprehensive and adaptive solutions.

Funding

The research received no external funding.

Conflicts of interest

The authors declared that they have no conflicts of interest.

Data availability statement

The data cannot be made publicly available upon publication because they are not available in a format that is sufficiently accessible or reusable by other researchers. The data that support the findings of this study are available upon reasonable request from the authors.

ORCID iDs

Arindam Mondal https://orcid.org/0000-0003-3210-1685

References

[1] Pucher J, Peng Z R, Mittal N, Zhu Y and Korattyswaroopam N 2007 Urban transport trends and policies in China and India: impacts of rapid economic growth *Transport Reviews* **27** 379–410

[2] Ansari D and Holz F 2020 Between stranded assets and green transformation: fossil-fuel-producing developing countries towards 2055 *World Development* **130** 104947

[3] Rietmann N, Hügler B and Lieven T 2020 Forecasting the trajectory of electric vehicle sales and the consequences for worldwide CO_2 emissions *J. Clean. Prod.* **261** 121038

[4] Haldar S, Mandal R and Mondal A 2019 Impact of clean transportation systems in rural economy: a study *In AIP Conf. Proc.* Vol. 2091 (AIP Publishing)

[5] IEA 2024 *Global EV Outlook 2024* (IEA) https://iea.org/reports/global-ev-outlook-2024 Licence: CC BY 4.0

[6] Ministry of Heavy Industries 2022 *FAME India Scheme Phase-II* Available: https://pib.gov.in/PressReleasePage.aspx?PRID=1848751

[7] Saxena S N 2019 Revolution in growth of three-wheeler electric vehicles in India Providing job opportunities to semi-skilled and unskilled people *Journal of Global Tourism Research* **4** 117–26

[8] Gupta R, Budhiraja N, Mago S and Mathur S 2020 An IoT-based smart parking framework for smart cities *Data Management, Analytics and Innovation: Proc. of ICDMAI 2020* vol. 1 (Springer) 19–32

[9] Liu Y, Yu H and Fang H 2021 Application of KNN prediction model in urban traffic flow prediction *2021 5th Asian Conf. on Artificial Intelligence Technology (ACAIT)* (IEEE) 389–92

[10] Babaei M and Behzadi S 2023 Spatial data-driven traffic flow prediction using geographical information system *Journal of Soft Computing in Civil Engineering* **7** 132–43

[11] Shenghua H, Zhihua N and Jiaxin H 2020 Road traffic congestion prediction based on random forest and DBSCAN combined model *2020 5th Int. Conf. on Smart Grid and Electrical Automation (ICSGEA)* (IEEE) 323–6

[12] Harrou F, Zeroual A and Sun Y 2020 Traffic congestion monitoring using an improved kNN strategy *Measurement* **156** 107534

[13] Akhtar M and Moridpour S 2021 A review of traffic congestion prediction using artificial intelligence *Journal of Advanced Transportation* **2021** 8878011

[14] Saleem M, Abbas S, Ghazal T M, Khan M A, Sahawneh N and Ahmad M 2022 Smart cities: fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques *Egyptian Informatics Journal* **23** 417–26

[15] Ashifuddin Mondal M and Rehena Z 2019 Intelligent traffic congestion classification system using artificial neural network *Companion Proc. of the 2019 World Wide Web Conf.* 110–6

[16] Gu Y, Wang Y and Dong S 2020 Public traffic congestion estimation using an artificial neural network *ISPRS International Journal of Geo-Information* **9** 152

[17] Zhu Q, Huang Y, Lee C F, Liu P, Zhang J and Wik T 2024 Predicting electric vehicle energy consumption from field data using machine learning *IEEE Transactions on Transportation Electrification* **11** 2120–32

[18] Qi X, Wu G, Boriboonsomsin K and Barth M J 2018 Data-driven decomposition analysis and estimation of link-level electric vehicle energy consumption under real-world traffic conditions *Transportation Research Part D: Transport and Environment* **64** 36–52

[19] Alsafery W, Alturki B, Reiff-Marganiec S and Jambi K 2018 Smart car parking system solution for the internet of things in smart cities *2018 1st Int. Conf. on Computer Applications & Information Security (ICCAIS)* (IEEE) 1–5

[20] Kabir A T, Saha P K, Hasan M S, Pramanik M, Ta-Sin A J, Johura F T and Hossain A M 2021 An IoT based intelligent parking system for the unutilized parking area with real-time monitoring using mobile and web application *2021 Int. Conf. on Intelligent Technologies (CONIT)* (IEEE) 1–7

[21] Somani A, Periwal S, Patel K and Gaikwad P 2018 Cross platform smart reservation based parking system *2018 Int. Conf. on Smart City and Emerging Technology (ICSCET)* (IEEE) 1–5

[22] Farooqi N, Alshehri S, Nollily S, Najmi L, Alqurashi G and Alrashedi A 2019 UParking: developing a smart parking management system using the internet of things *2019 Sixth HCT Information Technology Trends (ITT)* (IEEE) 214–8

[23] Hainalkar G N and Vanjale M S 2017 Smart parking system with pre & post reservation, billing and traffic app *2017 Int. Conf. on Intelligent Computing and Control Systems (ICICCS)* (IEEE) 500–5

[24] Kanteti D, Srikar D V S and Ramesh T K 2017 Smart parking system for commercial stretch in cities *2017 Int. Conf. on Communication and Signal Processing (ICCSP)* (IEEE) 1285–9

[25] Lookmuang R, Nambut K and Usanavasin S 2018 Smart parking using IoT technology *2018 5th Int. Conf. on Business and Industrial research (ICBIR)* (IEEE) 1–6

[26] Sayeed M S, Abdulrahim H, Razak S F A, Bukar U A and Yogarayan S 2023 IoT raspberry Pi based smart parking system with weighted k-nearest neighbours approach *Civil Engineering Journal* **9** 1991–2011

[27] Haldar S, Mondal S, Mondal A and Banerjee R 2022 State of health and life cycle prediction of in-vehicle lead acid battery *2022 Int. Interdisciplinary Conf. on Mathematics, Engineering and Science (MESIICON)* (IEEE) 1–6

[28] Hilmani A, Maizate A and Hassouni L 2018 Designing and managing a smart parking system using wireless sensor networks *Journal of Sensor and Actuator Networks* **7** 24

[29] Haldar S, Gol S, Mondal A and Banerjee R 2024 IoT-enabled advanced monitoring system for tubular batteries: enhancing efficiency and reliability *E-Prime-Advances in Electrical Engineering, Electronics and Energy* **9** 100709

[30] Google 2025 Diamond Harbour, South 24 Parganas, West Bengal [Google Maps]. Retrieved May 6, 2025, from https://google.com/maps/@22.1962542,88.1917402,16.75z

[31] Zou J, Han Y and So S S 2009 Overview of artificial neural networks *Artificial Neural Networks: Methods and Applications* **458** 14–22

[32] Google 2025 Diamond Harbour, South 24 Parganas, West Bengal [Google Maps]. Retrieved May 6, 2025, from https://google.com/maps/@22.1924088,88.1883701,18z

[33] Google 2025 Sarisha, South 24 Parganas, West Bengal [Google Maps]. Retrieved May 6, 2025, from https://google.com/maps/@22.2479594,88.1831,17.25z