ISSN: 2093-5374 / E-ISSN: 2093-5382

Machine Learning Based Vehicle Traffic Patterns Prediction Model (ML-VTPM) With Mobile Crowd Sensing for Transportation System

V. Mohammed Hussain[1], Dr.A. Abdul Azeez Khan[2*], Dr. Javubar Sathick[3], Dr. Arun Raj[4], and Dr.A. Haja Alaudeen[5]

[1]Research Scholar, Department of Computer Applications, B.S. Abdur Rahman Crescent, Institute of Science and Technology, Chennai, India. hussain28.ios@gmail.com, https://orcid.org/0009-0009-7706-3329

[2*]Associate Professor, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India. abdulazeezkhan@crescent.educaton https://orcid.org/0000-0001-6960-752X

[3]Associate Professor, Department of Computer Applications, B.S Abdur Rahman Crescent Institute of Science and Technology, Chennai, India. ja,vubar@crescent.education https://orcid.org/0000-0002-2248-8380

[4]Associate Professor, Department of Computer Science and Engineering, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India. arunraj@crescent.education. https://orcid.org/0000-0001-8181-5022

[5]Assistant Professor, Department of Computer Science and Engineering, B.S. Abdur Rahman Crescent Institute of Science and Technology, India. hajaalaudeen@crescent.education https://orcid.org/0000-0002-9710-373X

Received: October 15, 2024; Revised: December 04, 2024; Accepted: January 03, 2025; Published: March 31, 2025

Abstract

The new paradigm of the Internet of Things (IoT), Mobile Crowd Sensing (MCS), can be used to manage traffic congestion, deliver more convenient services, and relieve the issues associated with traffic. Today, the roads with the lowest capacity and the oldest infrastructure cannot accommodate the volume of cars that flow, which results in vehicular congestion. Most traffic congestion happens during peak hours, between eight and ten in the morning, when people are going to their places of employment when students are attending educational institutions, and between four and eight in the evening when they return home. Hence this paper proposed Machine learning-based Vehicle Traffic Patterns Prediction Model (ML-VTPM) for urban transportation systems accompanied by cloud-assisted MCS architecture. Internet of Things (IoT) based sensing data devices gathered continually from many cell phones carried by drivers provide cloud-assisted MCS with the ability to regulate traffic congestion. The MCS architecture can make real-time predictions about traffic based on the information gathered from smartphones (such as speed, direction, and position). After that, the K-means algorithm makes it possible to partition the traffic into smaller groups using clustering. The weights of each cluster are then computed using the convex hull technique. The proposed ML-VTPM technology can correctly calculate the route, allowing for the fastest travel time. An offline

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 16, number: 1 (March), pp. 1-25. DOI: *10.58346/JOWUA.2025.I1.001*

*Corresponding author: Associate Professor, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India.

technique based on machine learning models to forecast the mobility of vehicles in the days and weeks to come. Compared to conventional systems, the collected findings show that the suggested system offers a shorter distance and time for travel in various traffic circumstances.

Keywords: Machine Learning (ML), Mobile Crowd Sensing (MCS), K-means Clustering, Internet of Things (IoT).

1 Introduction of Vehicle Traffic Prediction Using Machine Learning For Transportation System

i) Urban Transportation System

Urban and transportation infrastructures are facing significant problems due to the growing number of people living in urban areas and the rising demand for mobility (Gohar & Nencioni, 2021). Congestion, side-street accidents, and traffic problems are rising due to city overpopulation and more people owning and operating personal automobiles (Mahrez et al., 2021). "Internet of Things" (IoT) describes a network of interconnected computing devices that can collect and process data wirelessly. This includes smart wearables, self-driving cars, smartphones, domestic appliances, and machinery (Ponnusamy & Alagarsamy, 2023). An Internet of Vehicles (IoV) exists whenever automobiles function as an ad hoc network via Internet connectivity. An important part of traffic management is using Internet of Things (IoT) devices. These devices typically have powerful processors set up to provide sufficient cycle time and green times for each signal, and they also have detectors to ensure that priority vehicles get extra attention (Musa et al., 2023). People increasingly consider improving transportation services using Intelligent Transportation Systems (ITS) (Montoya-Torres et al., 2021). With communication, sensing, and data processing technologies, ITS is anticipated to provide many practical applications. These include improving road safety, monitoring traffic conditions, and optimizing routes (Montoya-Torres et al., 2021). ITS aims to alleviate the growing number of transportation-related issues and increase efficiency (Raza et al., 2021). ITS aims to reduce energy consumption, improve air quality, and increase efficiency and safety by giving traffic authorities and drivers real-time traffic information. Traffic prediction is vital to any intelligent transportation system (Yuvaraj et al., 2022).

ii) Issues on Transportation System

Road surveillance cameras and loop detectors are examples of the specialized sensing infrastructures used to gather traffic-related data. Such infrastructures' expensive construction and maintenance costs make them impractical for use in low-resource nations and hinder their worldwide implementation despite the rich and dense data they may give regarding ongoing traffic (Chan et al., 2021). An endless number of sensing devices spread out around the city can detect the health of different urban infrastructure components (Lv et al., 2021). Due to resource and power constraints, sensing devices often cannot handle this kind of data locally and must depend on the data centre for storage and analysis. Consequently, the data centre must receive and evaluate the sensor data in real time before making judgments (Shaaban et al., 2021). Obtaining up-to-the-minute traffic data is one of the difficulties of ITS platforms. It would be impossible to provide such ITS applications without reliable, real-time traffic data set (Yu et al., 2021).

iii) Mobile Crowd Sensing and Machine learning-based Vehicle Traffic Monitoring

Modern developments in AI and ML have enabled smart city monitoring devices to track environmental variables more accurately, allowing for better management of pollution, traffic, and other negative impacts (Zhao et al., 2022). Using ML, intelligent traffic congestion control systems (TCCSs) may be mechanized to boost efficiency and reduce travel costs more precisely and reliably. One common method for dealing with data scarcity is Mobile Crowdsensing (MCS), which allows end-users, or individuals using mobile devices with sensors, to contribute data reporting (Agrawal et al., 2023). Since there is no need to install the sensors in the field immediately, infrastructure-less systems might be described as those that rely on crowdsensing for the Internet of Things (IoT). Scalability, flexibility, time savings, and, most significantly, cost-effectiveness are all improved by MCS, which also increases the monitoring coverage area. Collecting data from many mobile users is possible using MOBILE crowd sensing (Yuan et al., 2022). Integrated sensing and data collection from the environment is made possible by users' mobile phones. Mobile vehicles' ubiquitous, mobile, and connected nature has led to their recent use for crowd-sensing activities (Saleem et al., 2022). The transportation system now has a new tool to combat traffic congestion due to merging mobile interaction and intelligent terminal technologies (Ren et al., 2023). This tool, called Mobile Crowd Sensing (MCS), harnesses the power of multiple mobile devices, including mobile devices and automobiles equipped with sensors (Supangkat et al., 2023). Drivers and other participants in this sensing paradigm may transmit data collected from their mobile devices to a cloud-based traffic monitoring system. After that, the traffic status is communicated to drivers or relevant traffic administrators by examining traffic data (Jan et al., 2023).

Motivation for the Paper

Conventional roadside equipment, including loop detection systems and wayside cameras, is used for traffic monitoring applications. It would be difficult for infrastructure-based sensors to record the mode of transit. Smartphone sensors may be able to collect more data than some specialist sensors because people always have their phones on them. The proposed ML-VTPM with MCS-based cloud system for urban transportation networks. First, build an interactive MCS architecture with cloud-assisted capabilities to create an intelligent urban transportation system that benefits drivers, passengers, and relevant traffic authorities. Incentives, traffic recommendations, and social networks are a few key components of the service that ML-VTPM has examined. Secondly, it provides a novel use case for improving urban transportation's Quality of Service (QoS) via cloud-assisted traffic congestion management systems. Third explores the problems and difficulties of cloud-assisted MCS in urban transportation. These include things like system design, resource restrictions, security and privacy, and the concerns surrounding the system.

The Primary Objective of this Paper

i) Designing the proposed ML-VTPM for urban transportation systems accompanied by cloud-assisted MCS architecture.

ii) The capacity to control traffic congestion is provided by cloud-assisted MCS using sensing data continuously collected from many drivers' mobile phones.

iii) The proposed ML-VTPM technology can accurately determine the route, enabling the quickest possible journey time.

iv) The experimental results show that the ML-VTPM achieves high performance and reduces traffic in urban environments compared to other traditional methods.

The upcoming section 2 discusses the literature review, section 3 describes the proposed methodology, section 4 deliberates the numerical results and discussion and Section 5 concludes the overall paperwork.

2 Literature Review

Some scholars have suggested traffic mitigation strategies to improve the current traffic state. The most recent developments in traffic monitoring and prediction technology are discussed here.

2.1. Urban Traffic Monitoring Systems

A predictive target tracking method (PTTA) may accurately follow a moving target using very little power (Hussain et al., 2023). To predict a source's mobility using time-delayed data readings collected from IoT sensors, a convolutional long short-term memory (CNN-LSTM) model is suggested. IoT networks' scalability, simplicity of implementation, and real-time monitoring capabilities have made them valuable in environmental surveillance applications, such as target tracking. The proposed work uses a mathematical model established to optimise IoT devices' data throughput and reduce vehicle queue times, incorporating an artificial intelligence algorithm (Al-Ani et al., 2023). Maximizing the efficacy of the Internet of Things (IoT) edge computing and AI algorithms by decreasing the occurrence of control function faults. A comprehensive assessment of research on improving the efficiency of urban traffic management via the use of CVs and CAVs (connected and autonomous vehicles) in a mixed-traffic environment (Li et al., 2023). Future studies on urban traffic management in mixed-traffic environments may benefit from this research's useful road map. A deep network-based multi-object multi-camera tracking system (DNMO-MCTS) was suggested for monitoring traffic (Zaman et al., 2024). Proving its efficacy in precise and reliable vehicle tracking, the suggested framework attains competitive performance with an IDF1 score of 0.8289, accuracy of 0.9026, and recall of 0.8527. A CNN is used to segment aerial images in the proposed customized pyramid pooling (CPPM) module for smart traffic monitoring (Rafique et al., 2023). The proposed system demonstrated an impressive 95.78% vehicle detection rate through the experiment's assessment. This rate encompasses a wide range of tasks, such as identifying automobiles in traffic, recognizing the quantity of traffic at junctions and on roads, detecting different types of motor vehicles, and paving the way for pedestrians. The Traffic Responsive Control Framework (TRCF) is a centralized technology that simultaneously calculates the network decision variables (Storani et al., 2023). On a signalized arterial, the whole framework is put through its paces, with many evaluations carried out to calibrate the Model Predictive management (MPC) method and assess the traffic management approach employing fixed and adaptive management techniques.

2.2. Traffic Congestion Prediction

Real-time vehicle identification, tracking, and feature selection using the Edge-enhanced Cooperative Multi-camera Sensing (ECoMS) System (Yang et al., 2023). Regarding cooperative Internet of Things (IoT) architecture, ECoMS is the first system to follow vehicles and monitor traffic using several cameras. The EMVLight framework is a decentralized RL system that combines traffic signal pre-emption with collaborative dynamic EMV routing (Su et al., 2023). Using a new pressure-based incentive function and a creative design of multi-class RL agents, this system tackles the link between EMV navigation and traffic signal management. A traffic congestion-aware graph-based vehicle rerouting framework (TC-AG-VRF) that uses aerial images to determine optimal pathways for vehicles (Bayraktar et al., 2023). With the help of MaVefAI, the accuracy of the fine-tuned segmentation model for vehicle groups exceeds 98%. Extensive testing has shown that every algorithm takes the same route.

Optimization of traffic using a closed-loop control system (CLCS) that relies on license plate recognition (LPR) data (Li et al., 2023). In addition to demonstrating the overall benefit of real-time optimization, feedback, and control, the created model enhances the system's operational efficiency. Local traffic management centres may be able to use the suggested framework to enhance the control of traffic signals. A safety-prioritized receding horizon control framework (S-PRHCF) for the formation of human-driven vehicle (HDV) platoons, with CAVs operating at the forefront (Mahbub et al., 2023). To predict where autonomous cars will travel in the future, the system uses a data-driven prediction model that pulls from the recursion minimum squares algorithm and the principle of constant time distance relative speed car-following model. The combination compromise solution (CoCoSo) technique for ranking the tactics for reducing these risks and the improved fuzzy step-wise weight assessment ratio analysis (IMF-SWARA) method for identifying the most significant risk variables for transportation accidents (Badi & Bouraima, 2023). Other nations have used the IMF-SWARA technique to identify important risk variables and create good management systems.

Summary and Findings

After reviewing the literature, they concluded that no studies have ever considered enlisting automobiles to gather sensor data to predict their mobility using machine learning. By predicting the movement of each vehicle and striving to optimize the sensing data with a restricted budget, the proposed system may improve the amount of acquired sensing data, outperforming previous techniques.

3 Proposed Methodology

The busiest times of day for traffic are between eight and ten in the morning, when most people are heading to work or school, and four or eight in the evening when most people are heading home. This research suggested ML-VTPM with the help of cloud-assisted MCS architecture for urban transportation systems. These electronic equipment and sensors are set up in prominent places like roads, parking lots, traffic signals, and intersections. The cell phones of the drivers taking part may be one of these devices, along with infrared and ultrasonic sensors. The Internet of Things (IoT) and mobile crowd sensing play important roles. This entails collecting data from specific users by taking advantage of smartphones' GPS, accelerometer, and other sensors. Users might choose to install certain mobile applications that allow them to participate in this data collection. While users navigate the road network, these applications may either gather data invisibly or encourage them to report traffic conditions manually. Cloud-assisted MCS can control traffic congestion by constantly collecting data from many drivers' mobile phones. The MCS architecture can provide real-time traffic forecasts using data collected from cell phones. Then, clustering and the K-means method divide the traffic into smaller portions. Next, use the convex hull method to calculate the weights of every cluster. The proposed ML-VTPM system can ensure the quickest possible trip time by accurately calculating the route. Machine learning-based offline method for predicting future vehicle mobility.

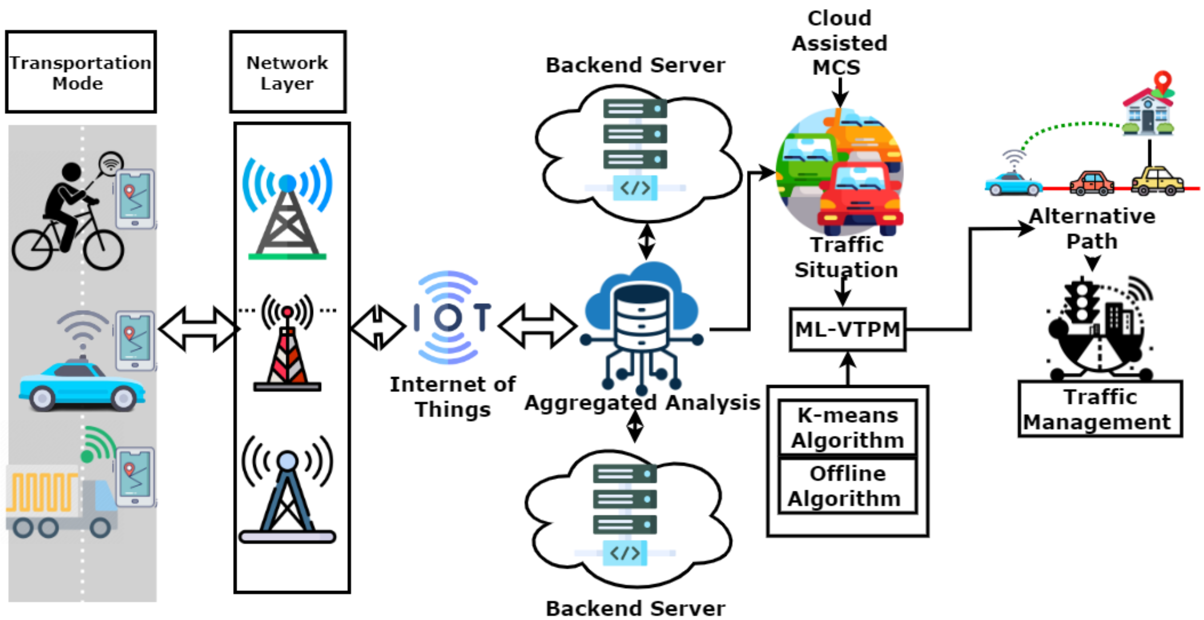


Figure 1: Proposed ML-VTPM

Figure 1 expresses the proposed ML-VTPM. There are several traffic scenarios that current transportation system traffic congestion control models (TCCMs) consider when planning how to handle traffic. Analytical and numerical techniques are used to study the dynamic models. They provide a vehicle motion equation-based dynamic model of traffic congestion. A concept for managing traffic congestion in metropolitan areas based on MCS was presented. There is no need for a specialized sensor network to gather data on traffic. With the help of the proposed MCS android mobile app (GetLocation), drivers can link their smartphones or in-car GPS sensors to the system, which then gathers real-time traffic data and sends it to the traffic management subsystem to forecast how many vehicles will be on each road. Find the number of cars on each route and see how they stack up against the capacity of the routes. An adaptive algorithm is used by the traffic management subsystem to control vehicle traffic effectively. Urban traffic congestion management systems aim to reduce average waiting times and prevent congestion. The MCS-based system for managing urban traffic congestion is shown in Figure 1 by its tiered functionalities.

- **Data Sensing Layer:** Users of the data sensing layer can collect traffic-related data in real time using their mobile devices. This information includes vehicle acceleration, density, and flow. Users who desire to submit their data can collect this information. In addition to that, this data could also contain things like readings from an accelerometer, locations from a GPS, and other information.
- **Network Layer:** Data is collected from a broad range of sensors strategically placed across the transportation network. Types of sensors include intersection light sensors, loop detectors, and management sensors. These sensors also give additional information about the traffic circumstances in certain locations. Data sensing layers are tasked with collecting data, and it is the network layer's task to relay that data to the technology that handles central processing or servers. One important phase in this process is establishing and regulating the paths for data transfer between mobile phones and tablets, sensors, and the computer's CPU.
- **Internet of Things (IoT):** Vehicle count, speed, congestion level, and even environmental variables like weather are just a few of the metrics that the Internet of Things (IoT) sensors are constantly

collecting data on. Aside from Internet of Things (IoT) devices, mobile crowd sensing is also important. The idea is to collect consumer data using cell phone sensors, such as GPS, accelerometer, and others. Users consent to be contacted in this way by downloading specialised mobile applications. The applications in question may either do nothing as the user navigates the road network or aggressively encourage the user to report traffic conditions manually. The combined data from mobile applications and Internet of Things (IoT) devices provides a thorough picture of the current traffic situation. Effective cleaning, merging, and analysing such diverse data often involve machine learning methods.

- **Cloud Layer:** The cloud layer provides a stable and expandable storage infrastructure, which stores enormous volumes of data collected from various sources, including mobile crowd sensing. Data that is both structured and unstructured are included in this group of information. The data under the heading above are traffic records, real-time instrument measurements, and documentation.
- **Machine Learning:** Various computer resources housed in the cloud are utilized to process and analyze the collected data. Data analytics techniques, statistical models, and machine learning algorithms must be implemented to achieve this goal. The plan includes finding patterns, drawing conclusions, and offering suggestions on traffic behaviour.
    - **K-means algorithm:** K-means clustering is used to categorize comparable traffic patterns or behaviours using features gathered from the data. Depending on factors such as density, speed, and direction, it is possible to create groups of automobiles. The K-Means algorithm may be used to aggregate geographical and temporal data to aid in detecting bottlenecks in transportation networks. This data may be of tremendous use in developing strategies for optimizing routes and controlling traffic.
    - **Offline algorithm:** Algorithms for machine learning that are trained using historical data without an internet connection are called offline algorithms. This training is conducted before deploying the algorithms for real-time prediction. Features technology, the practice of identifying valuable attributes from historical traffic data to use as inputs to prediction models, is frequently used in offline algorithms. Time of day, weather, route features, historical traffic patterns, and traffic volume are all potential contributors.
- **Traffic Congestion Analysis:** It is recommended that dashboards or interactive maps be used to provide transportation authorities, city planners, and other affected individuals with a visual depiction of the anticipated traffic patterns.
- **Alternate Path Prediction:** The route forecasts and alternative pathways should be continuously updated depending on the real-time data received via mobile crowd sensing. Other sources of data should also be considered. It is recommended that route suggestions be dynamically adjusted to consider any changes in traffic conditions, accidents, road closures, or other unanticipated occurrences.
- **Traffic Management:** Combining mobile crowd sensing with machine learning-based traffic pattern prediction allows for effective route recommendations and efficient anticipation of alternative routes. Because of this, vehicles will be able to avoid traffic, reduce travel time, and enjoy better traffic flow.

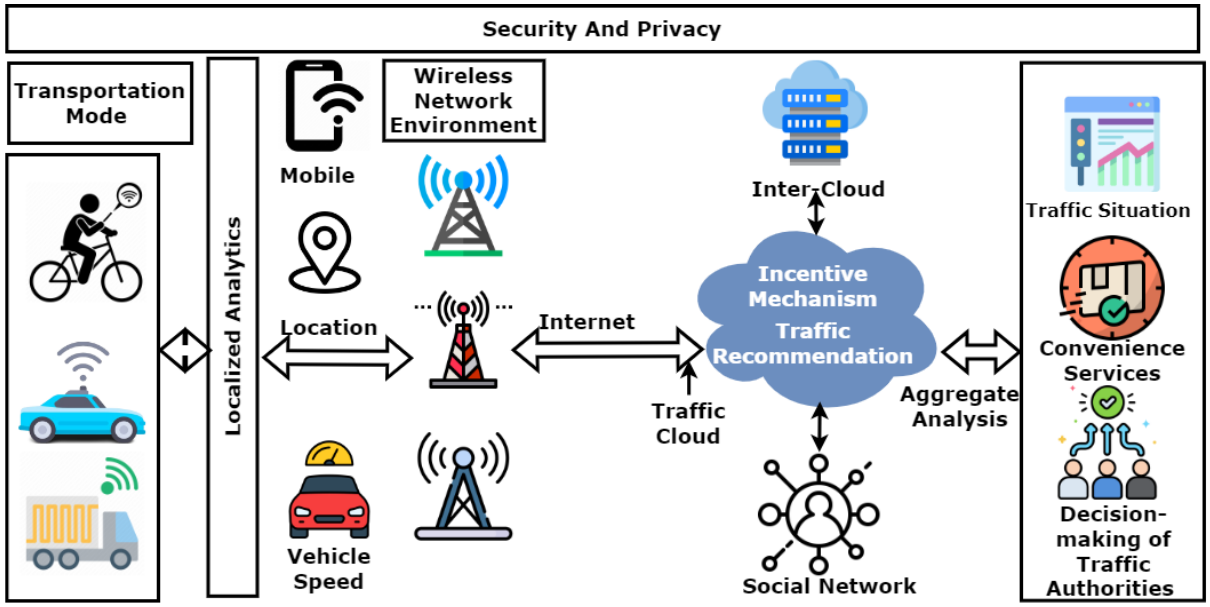


Figure 2: Cloud-assisted MCS Design

A cloud-assisted MCS design for urban transportation is shown in Figure 2. Computing on mobile devices, computing on location, and computing in the cloud are the three interacting layers that make up the computing structure. The essential parts of the service and the suggested design are here. When looking at it through the lens of Internet of Things applications, this design also has three levels. Smartphones or in-car sensors provide the perceptual layer with data (such as position and velocity). Data collection occurs at the transmission layer, communicating with the traffic cloud. At the computational layer in the cloud, for instance, traffic data mining tools are used to generate new applications from crucial traffic features like flow and concentrations. It is well known that sensors like GPS and cameras are part of the computational layer of mobile devices. Figure 1 shows that every tuple has at least five parameters for every mobile terminal: ID, position, speed, instructions, and mode. To cut down on data traffic and energy usage, it is recommended to perform localized analytics on the gathered data before sending it to a mobile base station. Another issue that has to be addressed is the forward frequency, which should be balanced with factors like energy consumption, real-time replies, data traffic, incentive mechanisms, and more. It should be easy for cell phones to communicate with roadside devices at the location computational layer. Sharing traffic reports and media libraries is another perk of having nearby cell phones. Due to MCS technology, new services (including shared taxis, carpooling, and traffic congestion management) are slowly being completed at the cloud computing layer. Multiple cloud systems that exchange resources form an inter-cloud environment, as shown in Figure 2. In this study, we focus on the traffic suggestion and incentive mechanisms, two of the numerous service components of the traffic cloud. After collecting analytics, it is possible to provide mobile users, particularly participants, with real-time traffic services.

There are several vital parts to the suggested cloud-assisted MCS design. Several essential parts of the service are detailed below:

- **Local Analytics:** Using local analytics reduces bandwidth, energy consumption, and storage strain on the mobile cloud backend. Thus, to fulfill the criteria above primarily focused on developing new algorithms for use in local analytics.

- **Aggregated Analysis:** Municipal traffic authorities may optimize traffic conditions and pay greater attention to the geographical distribution of traffic hotspots in aggregate analytics. Everyone behind the wheel probably keeps a careful eye on traffic conditions, such as the amount of congestion.
- **Incentive Mechanism:** More power and data transfer are required since drivers are involved in the MCS apps. As a result, some drivers may decide not to participate. As a result, we need to create reasonable incentive systems (like the MCFS or Mechanism of more Contributions and more Feedback Services) to entice potential consumers and encourage their involvement. The next part presents a case study of cloud-assisted MCS traffic congestion reduction using suggested MCFS.
- **Traffic Recommendation:** A journey from A to B is in the works. Anyone may access the cloud-assisted MCS platform's real-time traffic data via his cell phone before the start. Better adaptive tactics, such as moving to shared cars (public transit, ridesharing, etc.) or other forms of mobility like bicycling, maybe his choice after assessing the traffic situation by considering several options.
- **Mobile computing systems (MCS)** provide a novel approach to sensing by harnessing the capabilities of portable electronic devices like smartphones and vehicles outfitted with sensors. A participant, such as a passenger, in urban transportation may transmit traffic data collected from their smartphone to a stationary server. Transportation data mining aims to inform traffic authorities and drivers about the current traffic conditions.

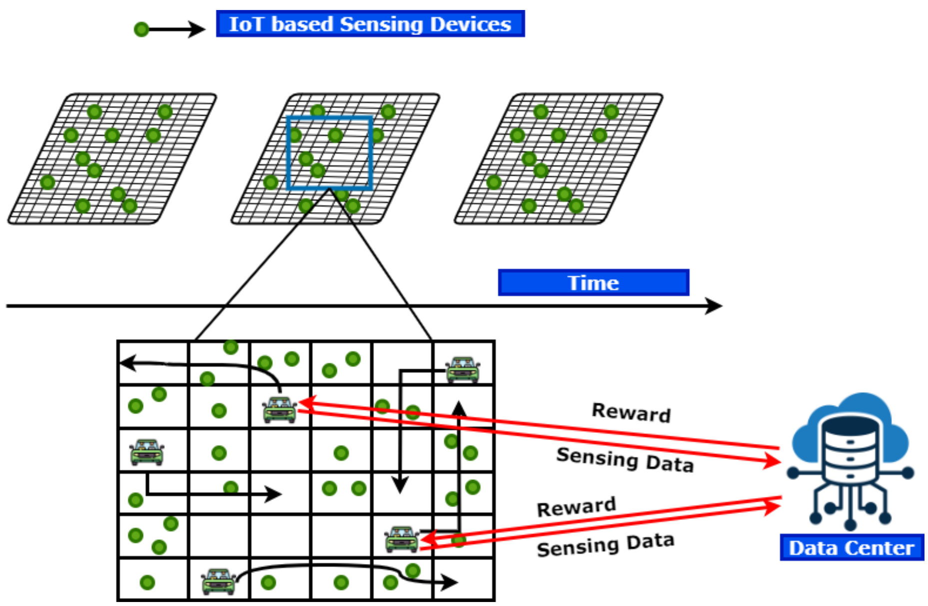


Figure 3: Vehicle Data Collection

Three types of things are considered: data centres, mobility vehicles, and sensing devices. i) Data centre: With a constrained budget, the data centre must assemble a fleet of mobile vehicles to complete the sensing data gathering mission. ii) Wheeled carriages: Smart city roadways are crowded with mobile vehicles. The data centre chooses which cars will gather sensing data; such vehicles will continue on their routes while using opportunist communication to gather data from other IoT-based sensing devices in the area. Following this, the cars transmit the data collected by their sensors to the data centre, where they will be rewarded. iii) IoT-based sensors: In smart cities, sensors detect and gather data about their immediate surroundings.

Figure 3 displays the plan for gathering data using mobile vehicles. Vehicles go over a two-dimensional map representing the sensing region dotted with sensing devices. Each grid comprises unique IoT-based sensing devices and produces unique sensing data types for each time slot. The map is divided into several grids, and we assume that vehicles can cover all the IoT-based sensing devices in one grid. The proposed Ml-VTPM plan considers sensor deployment and creates a sensor map. After processing, the vehicle dataset is converted into a vehicle trajectory matrix using the same format as the reference. These establish the vehicle-sensor coverage using the trajectory and sensor map. This article aims to help readers make the most of their limited budgets by hiring mobile vehicles to collect sensor data.

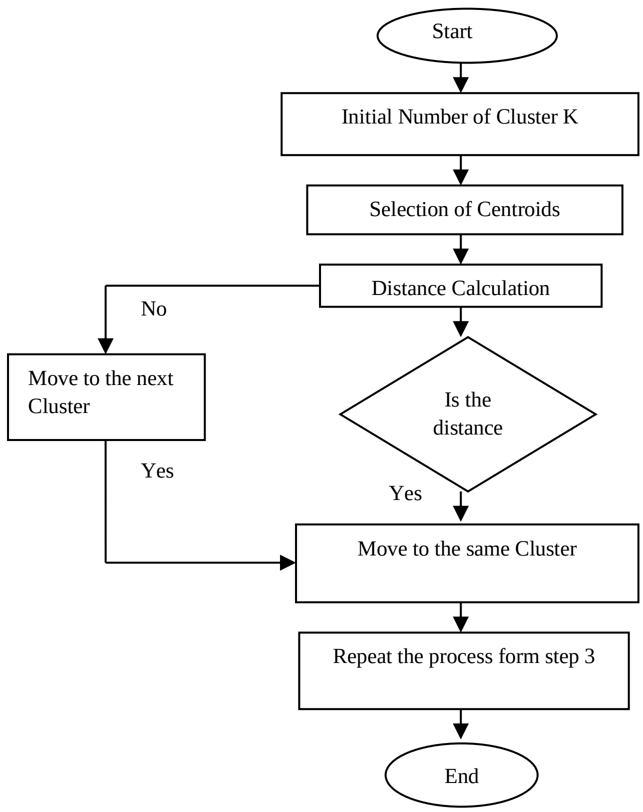


Figure 4: K-Means Algorithm

Specifically, our suggested system is built for highway topologies and incorporates a neural network-based method for initial cluster head selection and a modified K-Means algorithm for clustering. According to the suggested methodology, a cluster head is responsible for receiving, collecting, and aggregating data from all cluster nodes and disseminating this data to other cluster heads. In this way, the nodes that make up each cluster will communicate with the cluster head in a one-hop fashion. All nodes participate in route discovery by communicating with their cluster head exclusively through routing packets. According to the suggested strategy, the result is fewer network overhead and fewer control packets. Algorithm 1 demonstrates the K-means algorithm

Algorithm 1: K-Means Algorithm

Algorithm 1: K-means Algorithm
Input: Quantity of Centroids l
Coordinate system M; Centroids have been chosen at random D_l
Output: A collection of clusters together with their centres
Begin
Repeat
Every data point in M do
Find the distance from every point of information to the cluster centroid utilizing equation (1)
Find the closest centroid and assign it to the data point
End for
For each cluster in D_l do
Use equation (2) to determine the updated centroid location
End for
Repeat until either all data points are in the same cluster or the maximum number of iterations is achieved
End

Among the many clustering methods available, the K-Means approach stands out. Data mining, sensor networks, and ad hoc networks are just a few areas that use it extensively. An unsupervised learning algorithm divides a dataset into K-predefined groups. Its primary goal is minimising the gap between the cluster nodes and the leader. Figure 4 shows that the method selects a starting number of clusters l. Rearranging a set of points y_i where $1 \leq i \leq M$ into l clusters is the goal. In this case, K-Means randomly chooses l points xi from the dataset, where $1 \leq j \leq l$, to serve as centroid points, with each centroid belonging to a cluster D.

After that, the algorithm uses the closest centroid to assign each data point. As demonstrated in Figure 4 this procedure is predicated on an objective function that, for each cluster, determines the total of all squared distances within the cluster. The objective function is used to conduct the calculation:

$$avgmin_d \sum_{j=1}^{l} \sum_{y_i \in D_i} c(y_i, V_j) = avgmin_d \sum_{j=1}^{l} \sum_{y_i \in D_i} \left|y_i - V_j\right|^2 \qquad (1)$$

Equation (1) shows that the objective function has been computed. Where the distance separating the point and the geographic centre of the cluster is denoted as $(y_i, V_j) = \left|y_i - V_j\right|^2$. The point's position y_i and the centroid's position V_j are defined for where l is the number of clusters$i = 1, .., l$.

The results of the cluster assignments were made using the K-Means method. The algorithm uses (2) to update the position of each centroid:

$$V_j = \frac{1}{|d_j|} \sum_{i \in d_i} y_i \, , \forall j \qquad (2)$$

As found in equation (2), the position of the centroid update has been examined. The formation of clusters is the last step as seen in figure 4.

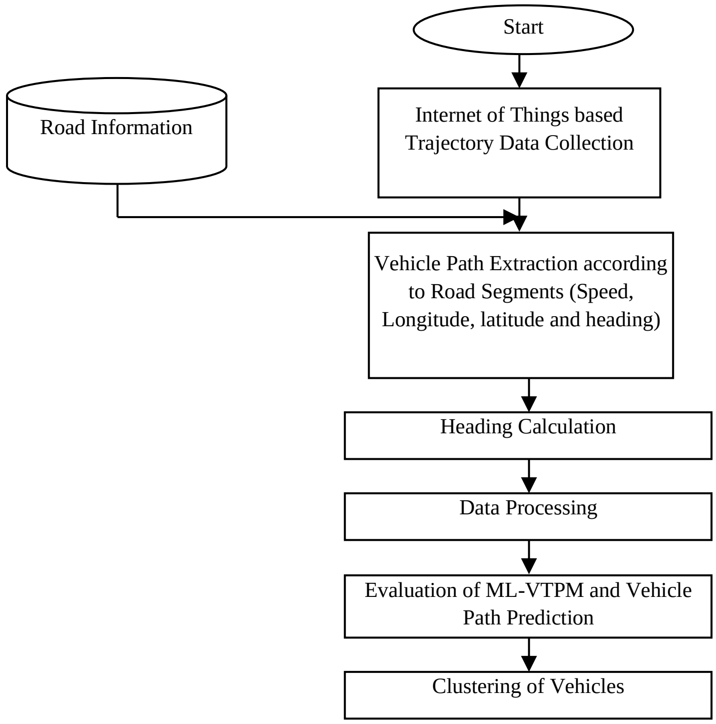


Figure 5: Alternative Path Selection

The primary goal of this study is to provide a method for designing position prediction systems for automobiles by using a trajectory-based strategy and grouping vehicles travelling in the same direction. There are four distinct stages to the method:

- In the first stage, "data collection," car GPS sensors record information about their movements. Mobile crowd sensing involves interacting with consumers via mobile apps, which augments data gathered from IoT sensors. Participation is voluntary and accomplished via specialized mobile applications that access the user's handsets' GPS, accelerometer, and other sensors. Users' speed, route selections, and the present state of traffic are only some of the data collected by these applications as they navigate the road network.
- The four characteristics of IoT devices used to collect GPS data are vehicle motion, speed, longitude/latitude, and yaw angle. The GPS data is a continuous route that starts a few meters from the junction and finishes there.
- The navigational direction considers the vehicle's heading during the path prediction phase. It is possible to deduce the vehicles' future movements and directions from their current motions.
- The K-mean algorithm groups vehicles into clusters based on their trip duration and similarity to better schedule and route trips for motor vehicles travelling in the same direction. This clustering provides a route planner that offers users an alternative to the shortest path-based method.
- The direction of the work path is shown in Figure 5. In the first case, statistics on the road network and its cars are calculated, including the vehicles' speeds, longitude and latitude coordinates, yaw angle, and heading directions. In the fourth case, there is an effort to group cars going in the same direction so that they can see the quickest route.

To generate models in machine learning, it is common practice to split source datasets into two parts: training and testing. To construct the ML model, data acquired from the GPS sensor tracking the vehicle's path is used as training data. Therefore, the dataset for vehicle grouping and route prediction is the most important part of machine learning since it is this dataset that is used to train the model. Additionally, evaluate the machine's ability to anticipate the new reaction using test data. Cross-validation of data is also performed to ensure that the method used to train the machine is accurate and efficient. Data collection, data preparation for the training set, model selection, model training, model evaluation, parameter tweaking, and prediction are the steps involved in a machine-learning technique. Algorithm 2 describes the alternative path selection algorithm.

Algorithm 2: Alternative Path Selection

Algorithm 2: Alternative Path Selection Algorithm

Input: D is the current location, C_d is the coordinates of the destination
Begin
N is map area
$while$ Source ≠ Destination
Collect user-provided traffic statistics in real-time
Allocating weights and classifying traffic data
Determine where traffic is most dense on a certain map
IF (suggest path=traffic jam)
Find another path
else
Suggest other fastest route
end if
Again, fetch the source location
end while
end

Beginning with the user's present position D and the coordinates of the destination C_d that the user has specified, Algorithm 2 begins processing the data. The system retrieves the area map after the user inputs the traffic parameters. Next, the convex hull approach is used to apply weights after using K-means clustering to create a cluster of traffic data. The shortest route, free of obstructions, is anticipated based on weighted clusters. After a certain amount of time, this method will get the present location, traffic statistics in real-time, and the shortest route. Once the user reaches their goal, the procedure will repeat. Algorithm 3 discusses the offline training.

Algorithm 3: Offline Training

Algorithm 3: Offline Training
Input: O 1. Partition O_j into $O_j^{k-\mu+1\sim k}$ and $O_j^{k+1\sim 1\sim\mu}$ 2. For batch of $O_j^{k-\mu+1\sim k}$ and $O_j^{k+1\sim 1\sim\mu}$ do 3. Find the $O_j^{k-\mu+1\sim k}$ and into G_μ 4. Calculate G_μ and obtain X_j 5. Calculate vehicle trajectory in equation (3) 6. Update ML-VTPM Model **7. End For**

8. For each $O_j^{k-\mu+1\sim k}$ do
9. Compute X_j using ML-VTPM model
10. End for
11. $O' = \varphi$
12. For each $V_j$do
13. $O' = O' \cup X_j$
14. End For

Output: $\boldsymbol{O'}$

The historical trajectories of all vehicles V_j are subject to the sliding windowZ_μ. Each car has Z sliding windows, and there are n vehicles. The loss function is characterized as follows, and the two-dimensional cross-entropy function is minimized to simulate the historical vehicle trajectory:

$$Vehicle\ historical\ trajectory = -\frac{1}{m\times z}\sum_{j=1}^{m}\sum_{i=1}^{z} O_{j,i} \times log\left(X_{j,i}\right) \qquad (3)$$

As deliberated in equation (3), the trajectory of the vehicle history is expressed. The specifics of the offline training procedure are presented in Algorithm 3. As inputs compute the training loss using the ground-truth trajectory $O_j^{k-\mu+1\sim k}$ and the trajectory $O_j^{k-\mu+1\sim k}$. Drop and revise the forecasting model. It is common for the prediction model to become more accurate gradually during offline training. At last, the training model is in our possession. Using the trained model foretell when and where each vehicle will be in the future. Once we have the projected vehicle locations X_j for each vehicle $U_j$insert them into the j-th column of the expected vehicle trajectory matrix O'. Assume that all the areas are represented as $H = \{H_l|H_1, H_2,\cdots, H_q\}$; then divide the map into several grids. Easy to create and scalable, the grid division may change its width and length to provide varying granularity control levels.

There are an extensive number of sensors spread out over the city. The sensor flag T_l to ε with $\varepsilon = 0$, if grid H_l includes ε IoT based sensing devices. It is safe to set $T_l = 0$ if grid H_l is sensor-free. $T = \{T_l|T_1, T_2,\cdots, T_q\}$ is the representation of the sensor map.

$$T_l = \begin{cases} \varepsilon & if\ region H_l exists\ \varepsilon\ sensors \\ o & otherwise \end{cases} \qquad (4)$$

As explored in equation (4) IoT sensor map has been examined. The sensing data is collected by many vehicles that circulate the city. Let us assume that the vehicles are represented as $U = \{U_j|U_1, U_2,\cdots, U_m\}$. S is the set of all possible time intervals, represented as $\{S_j|S_1, S_2,\cdots, S_m\}$. If vehicle $U_j$comes over grid H_q in time slot S_j, where H_l is a subset of H assign $O_{ji}$the value of H_l. On the other hand, if the vehicle U_j doesn't come across any grid in S_j we set $O_{ji}\ to\ \emptyset$. $R = \{O_{ji}\}$ is the notation for the vehicle's trajectory.

$$O_{ji} = \begin{cases} H_q & if\ vehicle\ U_j passes\ by\ H_l \\ \emptyset & otherwise \end{cases} \qquad (5)$$

As deliberated in equation (5), the total number of vehicle trajectories is described. Suppose there are m cars in total and each one recruited requires D_j rewards. The variables D representing the recruit costs are$\{D_j|D_1, D_2,\cdots, D_m\}$. Vehicle-sensor penetration is the total number of sensors that the cars cover in a certain period. For every $H_l \in H'$ and every$T_l \in T'$, where $H' \subseteq H$ and $T' \subseteq T$ establish a map$G: H' \subseteq T'$. It is our

$$G(H_l) = T_l \qquad (6)$$

As calculated in equation (6), the total number of vehicle IoT sensor penetrations has been discussed. Given that the data centre uses a subset of the cars in the set $U'$for recruitment purposes U, the coverage of vehicle IoT sensors during the time interval $S' \subseteq S$ is represented as

$$UT = \sum_{s_i \in S'} G\left(\bigcup_{U_j \in U'} O_{ji}\right) \quad (7)$$

As computed in equation (7), the time travel of vehicle IoT sensors has been found. Considering the following: regions G, sensor map T, vehicles U, time S, trajectory R, vehicle prices D, and budget A. To optimize the vehicle-sensor coverage UT, recruit several cars U, considering that the overall recruiting cost cannot exceed A.

$\arg\max_{U'} UT$

$$\sum_{s_i \in S'} D_u \leq A \quad (8)$$

The MCS system uses image processing to optimise traffic and transform a labelled map into an unlabeled one. The matrix is created by converting the road map $(n * n)$ and $(n^2 * n^2)$. This matrix shows the accessibility of the linked routes. Each node in the road network's DC_d route has its weight set to 1 at the outset. Here is a definition of the road network graph theory from equation (8):

$$H = (U, F) \quad (9)$$

As discussed in equation (9) road network map conversion has been discussed. Classifies the roads along a route from point D to point C_d, where U is a vector representing $(U_1, U_2, U_3, \dots, U_m)$a junction on the road network and F is an edge or path on the road network.

$$O = (f_1 \rightarrow U_1, U_2, f_2 \rightarrow U_2, \dots, en \rightarrow U_m, U_m + 1) \quad (10)$$

As explored in equation (10), road path identification was examined. Where the total amount of highways is m and O is the j-th road. An $(n * n)$ two-dimensional space matrix has been used to store the user-received traffic parameters. The system then uses K-means clustering to group the traffic depending on how similar or different the metrics are. This approach for clustering in M-dimensional space, based on Euclidean geometry, involves splitting a collection of m points into m groups. Representing n points $(n_1, n_2, n_3, \dots, n_m)$ and m clusters $(b_1, b_2, b_3, \dots, b_m)$ in a set Q according to equation (11) then,

$O_j \neq \vartheta$ for $(j = 1,2,3, \dots, m)$

$O_j \cap O_i \neq \vartheta$ for $(j = 1,2,3, \dots, m)$

$(j = 1,2,3, \dots, m)$ and $j \neq i$ and

$$\bigcup_{j,i=1}^{m} O_j, O_i = Q \quad (11)$$

As found in equation (11), Euclidean geometry was computed. All users' traffic data was shown via the map's traffic flow. The cluster's weight indicates the amount of traffic on the route. Equation (12) defines the road networks the system uses to determine the traffic density$C$.

$$C = \begin{cases} free\ flow & 1 & if \frac{O_k}{V_{ji}} \leq 1 \\ oridinary\ flow & \frac{O_k}{V_{ji}} & if 1 < \frac{O_k}{V_{ji}} < 10 \\ congestion & \infty & if \frac{O_k}{V_{ji}} > 10 \end{cases} \quad (12)$$

As demonstrated in equation (12) traffic density has been examined and the proposed method improves accuracy. A vehicle's average journey time V_{ji}between junctions i and j, divided by the road's hourly flow rate O_k, yields three traffic conditions: free flow, normal flow, and congestion. Overload sets in when the hourly flow rate/average journey time speed ratio exceeds 10. The passenger's overall trip time is calculated in this model. The connection between the number of vehicles per square centimetre of road traffic $C_j$and the average speed of the road t_j is assumed to be linear. This need is outlined in equation (13).

$$t_j = t_i \left(\frac{1-C_j}{C_j}\right) \quad (13)$$

As computed in equation (13) vehicle road average speed has been formulated and the proposed method improves precision. Where C_j is the traffic density at the bottleneck and t_i is the segment's speed flow. The predicted transit time in a given region i, j may be determined using equation (14)

$$S_j = \frac{k_j}{t_j} \quad (14)$$

As equation (14) shows, travel time was expressed to identify traffic congestion. Where k_j is the length of the trip and S_j is the anticipated travel time. The convex hull approach is used to calculate the traffic density for each cluster that has been detected. It shows how congested some parts of the road are. Then, our suggested system determines the alternative, fastest, and congestion-free routeK_p. Due to the high traffic volume in neighbourhood matrices roads are often disregarded to prevent gridlock. Further, the total time S_s required to go from the starting point to the final destination via the fastest route may be calculated using equation (15).

$$S_s = \frac{K_p}{t_j} \quad (15)$$

As described in equation (15), the alternate fastest route identification is expressed. This study proposed ML-VTPM for urban transportation networks using cloud-assisted MCS architecture. Internet of Things (IoT) sensors are placed strategically at several sites throughout the road network, including highways, crossroads, and major streets. Various technologies that can detect and measure traffic flow, vehicle speed, and congestion levels may be used as sensors. These devices include cameras, infrared sensors, ultrasonic sensors, and more. Information gathered in real-time by the Internet of Things (IoT) sensors includes traffic density, the number of cars going through, and their speeds. By involving individual users via mobile apps, mobile crowd sensing augments the data acquired from IoT sensors. Installing specialized applications that use GPS, accelerometer, and other sensors on users' cellphones allows them to join if they like. These applications track users' movements as they navigate the road system, recording details like their speed, the routes they've chosen, and the traffic conditions. Additional information, such as the ability to manually record accidents, road closures, or construction zones, may also be requested from users. Traffic congestion is managed by a cloud-assisted MCS that continuously gathers data from many drivers' mobile devices. The MCS architecture can provide up-to-the-minute traffic forecasts using data collected from mobile phones. This model offers a dynamic and all-encompassing method of vehicle traffic monitoring by integrating data from Internet of Things (IoT) sensors with mobile crowd sensing. This allows for better transportation systems and more effective traffic management.

4 Numerical Analysis

Around eight and ten in the morning, when the majority of people are on their way to work or school, along with between four and eight in the evening, when the majority of people are on their way home, are the periods of the working day when traffic is at its busiest. This work used cloud-assisted MCS architecture to propose ML-VTPM for city transportation networks. The road network strategically uses Internet of Things (IoT) sensors placed at key locations like highways, crossroads, and major streets. When users manually report incidents, road closures, or construction zones, for example, they may be asked to provide further information. This methodology allows for better transportation systems and more effective traffic management by integrating data from mobile crowd sensing with information from Internet of Things (IoT) sensors to provide a holistic and dynamic approach to monitoring vehicle traffic. Through continuous data collection from a large number of drivers' mobile devices, cloud-assisted MCS manages traffic congestion. The MCS architecture may use mobile phone data to provide real-time traffic predictions. The traffic is then divided into smaller parts using clustering and the K-means algorithm. Next, the weights of each cluster will be computed using the convex hull approach. The suggested ML-VTPM method can guarantee the shortest travel time by precisely estimating the route. Machine learning-based offline approach to future mobility prediction of vehicles.

Dataset Description: This dataset includes a video example demonstrating using YOLOv8 and ByteTracker for vehicle recognition and counting (Su et al., 2023). Using the YOLOv8 algorithm, cars are detected in every video frame, which depicts traffic on a busy street. To get a headcount of all the cars in the video, ByteTracker follows the recognized automobiles. Here are the files that make up the dataset: Video Evaluation. Below is an example video of heavy traffic on a major street in mp4 format. This is the mp4 file that YOLOv8 and ByteTracker created as an output. It displays the number of cars detected in each frame.

Experimental Setup: Two scenarios were considered throughout the simulation's execution: one in which the number of cars is increased from 100 to 300, and the other in which the speed of vehicles is decreased from 50 to 100 km/h. We have considered a 4-kilometre route with two lanes for both situations. There is bidirectional movement of the vehicles. A speed limit of 100 km/h is in effect for all vehicles. So, in a highway situation, the maximum speed may be anywhere from 50 to 100 in the simulation, but in reality, it's the actual speed of the car. Following ten iterations of the simulation, we averaged the outcomes shows in table 1.

Table 1: Details all of the Simulation Settings

Parameter	**Value**
Transmission Range	350 m
Vehicle Maximum Speed	120Km/h
Vehicle Minimum Speed	50 Km/h
Queue Length	100 Packets
Simulation Time	120s
Number of Simulations Runs	10
Critical Density	150
Number of IoT Devices	100, 200, 300, 400,500
Number of Vehicles	50,100, 150, 200, 250,300
Number of Lanes	3
Road Length	4 km
Topology	Highway

i) Performance Ratio (%)

Validation datasets are used to measure the efficacy of the trained model. Several metrics measure the model's capacity to detect vehicle traffic, including accuracy and precision correctly. Test the ML-VTPM using validation datasets to see whether it can correctly detect vehicle traffic patterns.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (16)$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (17)$$

$$Congestion\ Analysis = E(Y, \theta) \quad (18)$$

$$Vehicle\ Path\ (P) = E(Y) + F \quad (19)$$

Accuracy: The amount of accurately predicted occurrences of traffic patterns is represented by TP, which stands for True Positives. The amount of non-traffic pattern events that were accurately predicted is represented by TN (True Negatives). FP stands for "False Positives," which is the total number of traffic patterns that were mistakenly anticipated. False Negatives, or FN, is the sum of all the times non-traffic patterns were wrongly anticipated.

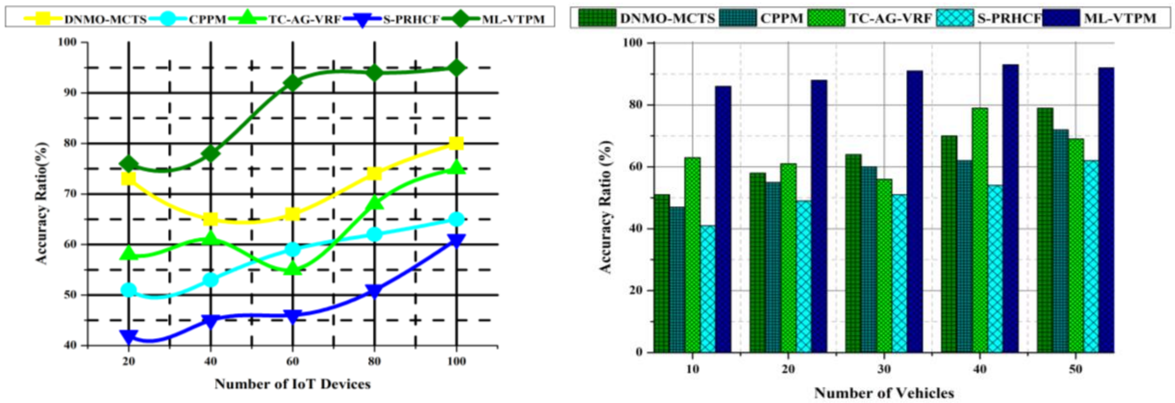
Precision: The number of accurate forecasts of traffic patterns or congestion levels is represented by TP, which stands for True Positives. False Positives, or FP, are the number of times traffic patterns or congestion levels are incorrectly predicted.

Traffic Congestion Analysis: The model's learnt parameters are denoted by θ. Here, "$congestion$" stands for the anticipated degree of traffic jams. The function that the ML-VTPM model learnt is denoted by E.Ystands for variables, such as speed and vehicle density, associated with the vehicle. Time (e.g., the hour of the day, the day of the week) is symbolized by S. D stands for elements specific to the situation, such as the current weather or the state of the roads. T denotes geographical elements, such as the arrangement of roads and neighbouring landmarks.

Path Prediction: The vehicle's expected route is called its path P. E represents the regression function that the machine learning model has learnt. The present environment and vehicle are described by X, a vector of input data that includes things like (Vvehicle characteristics, S traffic condition, D contextual factor, t spatial factors). To account for any noise or ambiguity in the forecast, the error term is represented by F.

ii) Accuracy Ratio (%)

ML-VTPM's data comes from various IoT-based monitoring sources, such as cameras, GPS devices, databases about road infrastructure, weather stations, and traffic sensors. A great deal of information regarding things such as vehicle density, speed, road conditions, weather, and other topics may be gleaned from the data obtained from various sources.



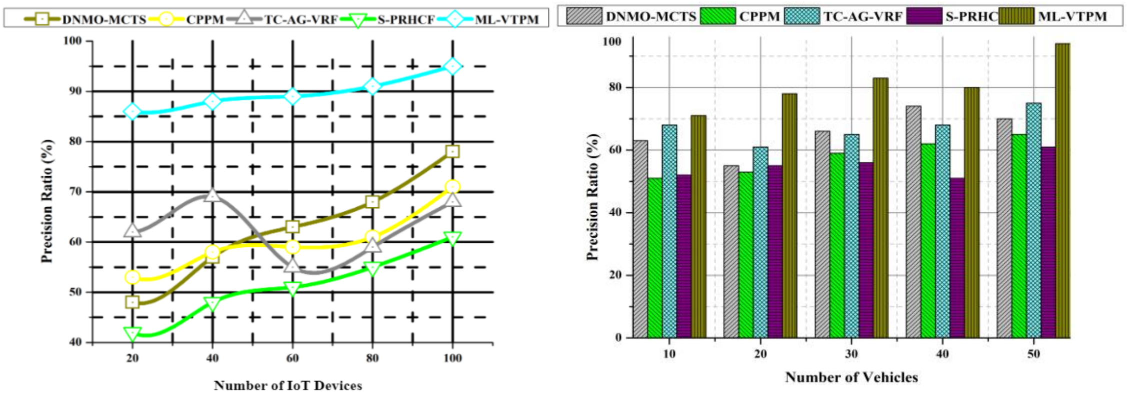
(a) (b)

Figure 6: Accuracy Ratio (%)

Figure 6 (a) and (b) explore the accuracy based on the number of IoT devices and vehicles using equation (12). The proposed model may represent many different elements of vehicle traffic by gathering appropriate attributes from the collected data. These qualities might include various parameters, including but not limited to vehicle speed, acceleration, density, lane occupancy, circulation patterns, road types, time of day, weather, and other elements. When training the ML-VTPM to predict vehicle traffic patterns, supervised machine learning is used to train the system on a problem. Identified historical data is used for this training. To enhance its capacity for prediction, the model acquires the ability to correlate input qualities with the traffic patterns that are associated with them. This formula (16) was used to determine what fraction of events was accurately anticipated. An increased accuracy percentage enhances the ML-VTPM model's ability to forecast vehicle traffic patterns.

iii) Precision Ratio (%)

Raw data collected from many sources, such as traffic sensors, cameras, and GPS devices, might have valuable properties extracted from it. Factors such as vehicle speed, flow density, road conditions, weather, and time of day are examples of such features. They must be normalised to ensure the characteristics have the same size and magnitude. Clustering algorithms like K-means need to have this. It is recommended to use the K-means approach to cluster the data into groups with commonalities in vehicle traffic patterns. Every cluster represents a unique traffic pattern or level of congestion. To find the optimal number of traffic patterns accurately reflecting the data, experiment with different values for L, where L is the number of clusters. Training the ML-VTPM on the clustered data will allow the building of a prediction model that can precisely identify vehicle traffic patterns.



(a) (b)

Figure 7: Precision Ratio (%)

Figure 7 (a) and (b) discuss the precision ratio using equation (13). The ML-VTPM may be trained using the clustered data in one of two ways: as labels or features, depending on the model's requirements. A combined approach of K-means clustering and ML-VTPM models is required to provide real-time forecasts of traffic patterns involving vehicles. Using historical data and the currently assigned clusters, the ML-VTPM may forecast future traffic patterns; this process begins with clustering incoming information into traffic patterns using the K-means algorithm. The ML-VTPM model forecasts traffic patterns or congestion levels that may be quantified using this equation (17). More accurate predictions are likely to be made by the model with a higher precision value, while more false positive predictions are possible with a lower precision number.

iv) Traffic Congestion Analysis (%)

Mobile devices constantly collect data about the vehicle's position, velocity, and acceleration with various IoT sensors, including GPS, accelerometer, gyroscope, and camera. The data is regularly sent to the cloud infrastructure to be stored and processed. Noise and outliers could be present in the raw sensor data retrieved from mobile devices. Data cleansing and outlier identification are examples of the pre-processing procedures that guarantee high-quality data.

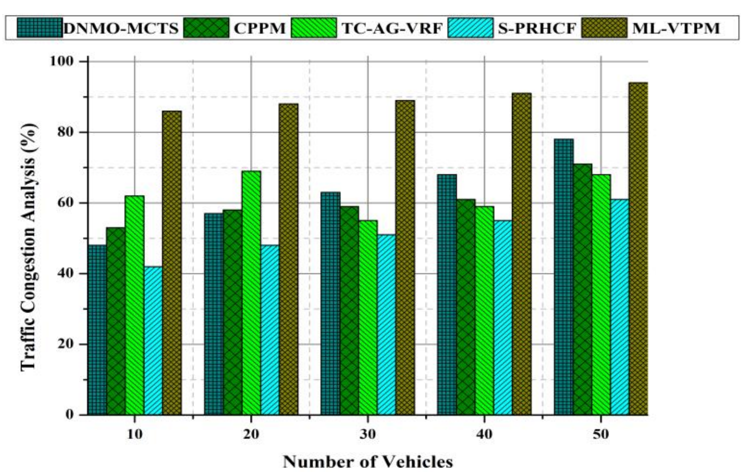


Figure 8: Traffic Congestion Analysis (%)

Figure 8 expresses the traffic congestion analysis using equation (14). With the help of distributed computing resources, the cloud architecture can manage massive amounts of IoT sensor data gathered from various mobile devices. The technology is hosted in the cloud and offers real-time visualizations and dashboards that show trends and patterns in vehicle traffic. The ML-VPTM technology may be enhanced using crowd-sourced input from mobile device users to verify and improve the precision of traffic pattern detection. With cloud computing, machine learning, and mobile crowd sensing, the model can detect and evaluate traffic patterns in real time, which is great for transportation optimization, city planning, and traffic management.

v) Path Prediction Ratio (%)

Gather information on past traffic, such as average speeds, degrees of congestion, road conditions, and other routes. Data must be preprocessed for analysis, including cleaning, filtering, and normalizing. Use the K-means algorithm to sort past traffic records by congestion levels or traffic patterns. Different clusters indicate traffic structures, such as light, moderate, or high congestion. Determine whether routes or places are most congested by analyzing the clusters. Find out which roads are prone to the worst traffic at specific times of the week or day.

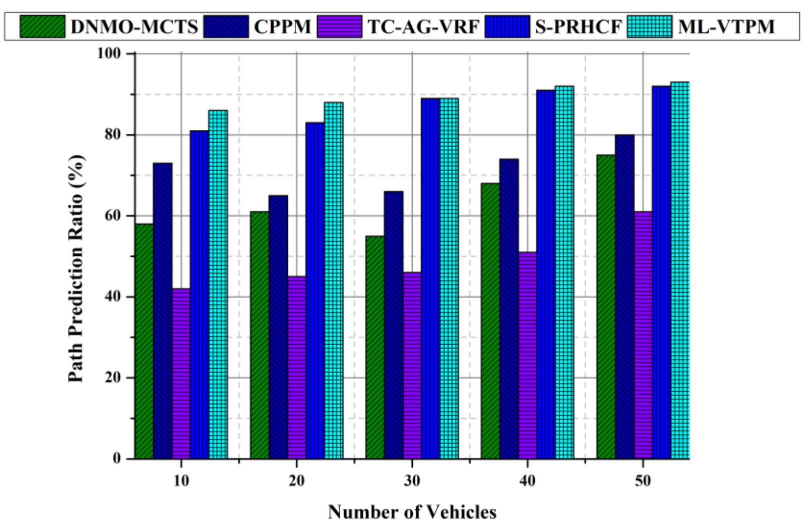


Figure 9: Path Prediction Ratio (%)

Figure 9 deliberates the path prediction ratio using equation (15). Utilize the past traffic data and K-means clusters to train an ML-VTPM. Based on characteristics including time of day, the environment, and road infrastructure, the ML-VTPM should be able to learn to anticipate future traffic patterns and congestion levels. Make advantage of the ML-VTPM to forecast the probable congestion levels along various routes given a certain origin-destination pair and the present traffic state. To monitor traffic flow, vehicle speed, and congestion levels, these sensors may use various technologies, including cameras, infrared, ultrasonic, and more. Internet of Things (IoT) sensors constantly gather data in real-time, including traffic density, vehicle speed, and the number of cars passing through. To supplement the data acquired from IoT devices, mobile crowd sensing involves interacting with individual users via mobile apps. Users may participate by downloading specialized applications that use their phone's GPS, accelerometer, and other sensors. While users navigate the road network, these applications record speed, route selections, and traffic data. The prediction model is a system that monitors traffic conditions in real time and gives drivers other routes based on what it finds.

Endnotes

These days, traffic jams are a real problem since the oldest highways and ones with the lowest capacity can't handle the amount of automobiles on the road. Therefore, ML-VTPM with cloud-assisted MCS architecture was developed in this article for use in urban transportation systems. An Internet of Things (IoT) and mobile crowd-sensing-based system are a state-of-the-art method for monitoring vehicle traffic conditions. This system uses networked sensors and the power of crowdsourcing via mobile devices to acquire real-time data. The Internet of Things (IoT) and mobile crowd sensing play important roles. This entails collecting data from specific users by taking advantage of smartphones' GPS, accelerometer, and other sensors. Cloud-assisted MCS provides the capacity to control traffic congestion by sensing data continuously collected from many drivers' mobile phones. The MCS architecture can provide real-time traffic forecasts by analysing smartphone data—including location, direction, and speed. The next step is to utilize clustering to divide the traffic into smaller portions using the K-means method. The convex hull method is then used to calculate the weights of every cluster. The advertised ML-VTPM technology can accurately determine the route, enabling the quickest possible journey time. A machine learning-based offline method for predicting road traffic in the next days and weeks. The experimental results show the proposed method accuracy ratio of 95.23% and 93.1%, the precision ratio of 94.36% and 95.61%, traffic congestion analysis of 97.61% and the prediction ratio of 96.7% compared to other methods.

References

[1] Agrawal, A., Choudhary, S., Bhatia, A., & Tiwari, K. (2023). Pub-SubMCS: A privacy-preserving publish–subscribe and blockchain-based mobile crowdsensing framework. *Future Generation Computer Systems*, *146*, 234-249. https://doi.org/10.1016/j.future.2023.04.018

[2] Al-Ani, A. K., Ul Arfeen Laghari, S., Manoharan, H., Selvarajan, S., & Uddin, M. (2023). Improved transportation model with internet of things using artificial intelligence algorithm. *Computers, Materials & Continua*, *76*(2). https://doi.org/10.32604/cmc.2023.038534

[3] Badi, I., & Bouraima, M. B. (2023). Development of MCDM-based frameworks for proactively managing the most critical risk factors for transport accidents: a case study in Libya. *Spectrum of engineering and management sciences*, *1*(1), 38-47.

[4] Bayraktar, E., Korkmaz, B. N., Erarslan, A. U., & Celebi, N. (2023). Traffic congestion-aware graph-based vehicle rerouting framework from aerial imagery. *Engineering Applications of Artificial Intelligence*, *119*, 105769. https://doi.org/10.1016/j.engappai.2022.105769

[5] Chan, R. K. C., Lim, J. M. Y., & Parthiban, R. (2021). A neural network approach for traffic prediction and routing with missing data imputation for intelligent transportation system. *Expert Systems with Applications*, *171*, 114573. https://doi.org/10.1016/j.eswa.2021.114573

[6] Gohar, A., & Nencioni, G. (2021). The role of 5G technologies in a smart city: The case for intelligent transportation system. *Sustainability*, *13*(9), 5188. https://doi.org/10.3390/su13095188

[7] https://www.kaggle.com/datasets/vivek603/vehicle-detection-sample-and-output-videos

[8] Hussain, L. A., Singh, S., Mizouni, R., Otrok, H., & Damiani, E. (2023). A predictive target tracking framework for IoT using CNN–LSTM. *Internet of Things*, *22*, 100744. https://doi.org/10.1016/j.iot.2023.100744

[9] Jan, M., Khattak, K. S., Khan, Z. H., Gulliver, T. A., & Altamimi, A. B. (2023). Crowdsensing for Road Pavement Condition Monitoring: Trends, Limitations, and Opportunities. *IEEE Access*, *11*, 133143-133159.
[10] Li, J., Yu, C., Shen, Z., Su, Z., & Ma, W. (2023). A survey on urban traffic control under mixed traffic environment with connected automated vehicles. *Transportation research part C: emerging technologies*, *154*, 104258. https://doi.org/10.1016/j.trc.2023.104258
[11] Li, R., Wang, S., Jiao, P., & Lin, S. (2023). Traffic control optimization strategy based on license plate recognition data. *Journal of traffic and transportation engineering (English edition)*, *10*(1), 45-57.
[12] Lv, Z., Li, Y., Feng, H., & Lv, H. (2021). Deep learning for security in digital twins of cooperative intelligent transportation systems. *IEEE transactions on intelligent transportation systems*, *23*(9), 16666-16675. https://doi.org/10.1109/TITS.2021.3113779
[13] Mahbub, A. I., Le, V. A., & Malikopoulos, A. A. (2023). A safety-prioritized receding horizon control framework for platoon formation in a mixed traffic environment. *Automatica*, *155*, 111115. https://doi.org/10.1016/j.automatica.2023.111115
[14] Mahrez, Z., Sabir, E., Badidi, E., Saad, W., & Sadik, M. (2021). Smart urban mobility: When mobility systems meet smart data. *IEEE Transactions on Intelligent Transportation Systems*, *23*(7), 6222-6239. https://doi.org/ 10.1109/TITS.2021.3084907
[15] Montoya-Torres, J. R., Moreno, S., Guerrero, W. J., & Mejía, G. (2021). Big data analytics and intelligent transportation systems. *IFAC-PapersOnLine*, *54*(2), 216-220. https://doi.org/10.1016/j.ifacol.2021.06.025
[16] Musa, A. A., Malami, S. I., Alanazi, F., Ounaies, W., Alshammari, M., & Haruna, S. I. (2023). Sustainable traffic management for smart cities using internet-of-things-oriented intelligent transportation systems (ITS): challenges and recommendations. *Sustainability*, *15*(13), 9859. https://doi.org/10.3390/su15139859
[17] Ponnusamy, M., & Alagarsamy, A. (2023). Traffic monitoring in smart cities using internet of things assisted robotics. *Materials Today: Proceedings*, *81*, 290-294. https://doi.org/10.1016/j.matpr.2021.03.192
[18] Rafique, A. A., Al-Rasheed, A., Ksibi, A., Ayadi, M., Jalal, A., Alnowaiser, K., ... & Park, J. (2023). Smart traffic monitoring through pyramid pooling vehicle detection and filter-based tracking on aerial images. *IEEE Access*, *11*, 2993-3007. 10.1109/ACCESS.2023.3234281
[19] Raza, A., Bukhari, S. H. R., Aadil, F., & Iqbal, Z. (2021). An UAV-assisted VANET architecture for intelligent transportation system in smart cities. *International Journal of Distributed Sensor Networks*, *17*(7), 15501477211031750.
[20] Ren, Y., Wang, T., Zhang, S., & Zhang, J. (2023). An intelligent big data collection technology based on micro mobile data centers for crowdsensing vehicular sensor network. *Personal and ubiquitous computing*, *27*(3), 563-579.
[21] Saleem, M., Abbas, S., Ghazal, T. M., Khan, M. A., Sahawneh, N., & Ahmad, M. (2022). Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques. *Egyptian Informatics Journal*, *23*(3), 417-426. https://doi.org/10.1016/j.eij.2022.03.003
[22] Shaaban, K., Shamim, M. H. M., & Abdur-Rouf, K. (2021). Visible light communication for intelligent transportation systems: A review of the latest technologies. *Journal of traffic and transportation engineering (English edition)*, *8*(4), 483-492. https://doi.org/10.1016/j.jtte.2021.04.005
[23] Storani, F., Di Pace, R., & De Schutter, B. (2023). A traffic responsive control framework for signalized junctions based on hybrid traffic flow representation. *Journal of Intelligent Transportation Systems*, *27*(5), 606-625. https://doi.org/10.1080/15472450.2022.2074790
[24] Su, H., Zhong, Y. D., Chow, J. Y., Dey, B., & Jin, L. (2023). EMVLight: A multi-agent reinforcement learning framework for an emergency vehicle decentralized routing and traffic

signal control system. *Transportation Research Part C: Emerging Technologies*, *146*, 103955. https://doi.org/10.1016/j.trc.2022.103955

[25] Supangkat, S. H., Ragajaya, R., & Setyadji, A. B. (2023). Implementation of digital geotwin-based mobile crowdsensing to support monitoring system in smart city. *Sustainability*, *15*(5), 3942. https://doi.org/10.3390/su15053942

[26] Yang, H. F., Cai, J., Liu, C., Ke, R., & Wang, Y. (2023). Cooperative multi-camera vehicle tracking and traffic surveillance with edge artificial intelligence and representation learning. *Transportation research part C: emerging technologies*, *148*, 103982. https://doi.org/10.1016/j.trc.2022.103982

[27] Yu, G., Wang, Y., Hu, M., Shi, L., Mao, Z., & Sugumaran, V. (2021). RIOMS: An intelligent system for operation and maintenance of urban roads using spatio-temporal data in smart cities. *Future generation computer systems*, *115*, 583-609. https://doi.org/10.1016/j.future.2020.09.010

[28] Yuan, T., da Rocha Neto, W., Rothenberg, C. E., Obraczka, K., Barakat, C., & Turletti, T. (2022). Machine learning for next-generation intelligent transportation systems: A survey. *Transactions on emerging telecommunications technologies*, *33*(4), e4427. https://doi.org/10.1002/ett.4427

[29] Yuvaraj, N., Praghash, K., Raja, R. A., & Karthikeyan, T. (2022). An investigation of garbage disposal electric vehicles (GDEVs) integrated with deep neural networking (DNN) and intelligent transportation system (ITS) in smart city management system (SCMS). *Wireless personal communications*, *123*(2), 1733-1752. https://doi.org/10.1007/s11277-021-09210-8

[30] Zaman, M. I., Bajwa, U. I., Saleem, G., & Raza, R. H. (2024). A robust deep networks based multi-object multi-camera tracking system for city scale traffic. *Multimedia Tools and Applications*, *83*(6), 17163-17181.

[31] Zhao, C., Lv, Y., Jin, J., Tian, Y., Wang, J., & Wang, F. Y. (2022). DeCAST in TransVerse for parallel intelligent transportation systems and smart cities: Three decades and beyond. *IEEE Intelligent Transportation Systems Magazine*, *14*(6), 6-17. https://doi.org/ 10.1109/MITS.2022.3199557

Authors Biography

V. Mohammed Hussain, born in 1989 in Tamil Nadu, India, is a research scholar at B.S. Abdur Rahman Crescent Institute of Science and Technology. He holds a master's degree in Computer Applications, completed in 2013. With over 11 years of experience in the IT industry, he currently serves as a Technical Staff Member in an IT company. His expertise includes Apple product development (iOS, macOS, and watchOS), as well as automation and IoT.

Dr.A. Abdul Azeez Khan was born in the year 1982, Tamil Nadu, India. He obtained his doctorate degree in the field of computer science in the year 2018. He is having 20+ years of experience put together in industry and academia. At present he is working as Associate Professor in the Department of Computer Applications at B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai-600048. His papers are published in many International Journals & Conferences. His area of interest includes Artificial Intelligence, Machine Learning, IoT, E-learning and Knowledge Management.

Dr.K. Javubar Sathick was born in the year 1984, Tamil Nadu, India. He obtained his doctorate degree in the field of computer science in the year 2018. He is having 17+ years of experience in academics. At present he is working as Associate Professor in the Department of Computer Applications and Deputy Director (Establishment) at B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai-600048. His papers are published in many International Journals & Conferences. His area of interest includes Artificial Intelligence, Machine Learning, IoT, E-learning and Knowledge Management.

Dr.L. Arun Raj is a Associate Professor in Department of CSE at B.S. Crescent Institute of Science and Technology, Chennai, India. He has 15 years of experience in teaching, research and administration. He has published more then 50+ publications in highly cited Journals and Conferences and 5 book chapters. His research interests are machine Learning, Health Care, Multimedia Applications, Next Gen. of Wireless Networks and IoT. He has delivered several guest lectures, seminars and chaired a session for various International and Conferences. He is serving as a Reviewer and Editorial Board Member of many reputed Journals and acted as Session chair and Technical Program Committee member of National conferences and International Conferences at Thailand, Singapore and China.

Dr.A. Haja Alaudeen is presently working as an Assistant Professor at the Crescent Institute of science and Technology, where he teaches courses on Mobile Application Development, Artificial Intelligence and IoT. With over 23 years of experience, he has held various academic and industry positions, including Business Head consultant for leading IT solution provider in Abudhabi, UAE. In addition to his teaching role, he is a reviewer for the most reputed journals in National and International level and also received best reviewer award in the year of 2023.He has completed his Undergraduate B.Sc from Bharathidasan University, Postgraduate M.Sc Software Engineering from Annamalai University and PhD in Mobile Computing from Anna University in the year 2018.