Synthesis: A Multidisciplinary Research Journal
Volume 4, Issue 1, January– March 2026, Pp: 34-45
© 2026, All Rights Reserved @ Macaw Publications

Research Paper

Edge-AI Enabled Uncertainty-Aware Intelligent Traffic Management System Using Graph-Based Forecasting for Smart Cities

[1] Sakhamuru Amulya, [2*] M.Sri Lakshmi [3] M Bhavsingh

[1] *Engineer IV, Collegeboard 11955 Democracy Dr, Reston, VA 20190, USA, Email: asakhamuru@collegeboard.org*
[2*] *Professor, Department of Computer Science and Engineering, G. Pullaiah College of Engineering and Technology (Autonomous), Kurnool, Andhra Pradesh, India, Email: srilakshmicse@gpcet.ac.in*
[3] *Associate Professor, Department of Computer Science and Engineering, Ashoka Women's Engineering College, Kurnool, Andhra Pradesh, India, Email ID: bhavsinghit@gmail.com*

**Corresponding Author(s): srilakshmicse@gpcet.ac.in*

Article Info
Received: 10/11/2025
Revised: 16/02/2026
Accepted:23/03/2026
Published:31/03/2026

Abstract
The high rate of urbanization has brought about a lot of traffic congestion to the smart cities, requiring smart and scalable traffic control solutions. Conventional signal control systems are not responsive to changing traffic situations, whereas current prediction-based models do not take into account the uncertainty factor, and hence unreliable decisions are made. To discuss the issue of the real-time optimization of traffic, this paper offers a solution to it through the creation of an Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS). The framework combines the edge-based traffic perception, graph based spatio-temporal prediction and uncertainty-conscious adaptive signal control into one architecture. At the edge the traffic descriptors extracted include the number of vehicles, the length of a queue and occupancy of a lane which are then modeled on a dynamic traffic graph to provide congestion patterns in the short term. A risk conscious control method has built in predictive uncertainty to enable sound decision making according to different traffic conditions. Our experimental analysis shows that the proposed method reduces the average waiting time by 19.2%, queue length by 21.9%, and increases the throughput by 18.9% compared to prediction-based control without uncertainty. Further, the system is real time with an average inference time of 28.5 ms. These findings demonstrate the efficiency of the combination of edge intelligence and uncertainty-aware control as a solution to the next-generation smart city traffic management systems, which is scalable and reliable.

Keywords: Edge Artificial Intelligence, Intelligent Traffic Management, Spatio-Temporal Graph Modeling, Uncertainty-Aware Control, Smart Cities, Adaptive Traffic Signal Control.

Copyright: © 2026 Sakhamuru Amulya, M.Sri Lakshmi and M Bhavsingh. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license.

1. Introduction

The high rate of urban population increase and the density of the vehicles has increased traffic congestion in the contemporary cities and this has increased the delays in the travel, the fuel consumption, and the impact on the environment. The Intelligent Traffic Management Systems (ITMS) have become one of the most important elements of smart city infrastructure, which is supposed to optimize traffic flow based on data-driven decision-making and adaptive control strategies [1], [2]. The recent breakthrough in artificial intelligence, edge computing, and connected infrastructure has made it possible to create more responsive and scalable traffic control systems, in which data can be processed nearer to the source to minimize latency and communication overhead [3], [4]. Specifically, Edge-AI paradigms enable the real-time processing of traffic conditions with the help of local computational resources, which is why they can be used in time-sensitive applications like traffic signal control [5].

In spite of these developments, the current methods of traffic management have a number of limitations. Conventional fixed-time and actuated signal management plans are not dynamic to the changing road traffic situation, thus leading to poor use of the road capacity [6]. Although recent machine learning and deep learning-based schemes have enhanced the prediction of traffic and adaptive control, most of them are centralized and fail to consider spatial relationships among intersections [7]. Moreover, predictive-based control systems commonly expect to have a perfect model reliability, and disregards uncertainty in the prediction of traffic, which may result in unstable or non-optimal control responses in the face of noisy or rapidly changing environments [8].

This paper is aimed to overcome these shortcomings by introducing a framework called Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS), which combines edge-based perception, graph-based spatio-temporal forecasting with uncertainty-aware adaptive signal control. The main goal of the work is to create a scalable and stable traffic management system, which can be used in real-time settings, considering the dynamics of traffic and predictability. The suggested framework uses small scale traffic descriptors obtained at the edge and processes them using a dynamic graph framework to generate the interdependencies between intersections and hence making them more capable of accurately and context sensitive prediction of traffic [9].

Nevertheless, there are a number of research challenges that arise in the design of such a system. First, the models must be efficient to ensure low-latency and precision of traffic perception at the edge in terms of computational constraints and detection performance. Second, the traffic dynamics across the intersections require the efficient spatio-temporal learning processes with the ability to capture both local and global dynamics. Third, incorporating uncertainty into the control decisions should be supported by strong formulations capable of avoiding unreliable courses of action at the expense of adaptability. Lastly, end-to-end coordination of the system in perception, prediction, and control is also a problem in system consistency and scalability [10].

In order to beat these challenges, the following are the key contributions of this paper:

- Hierarchical Edge-AI based architecture of traffic perception in real-time and decentralized processing of data.
- A temporally adaptable graphical forecasting model of traffic relationships within intersections.
- A risk estimation mechanism that is risk-averse to improve the quality of the traffic predictions.
- A signal control policy that is adaptive and has a fallback mechanism that is safety conscious to guarantee stable operation in uncertain conditions.
- An integrated assessment model by applying real world data and simulated environments to confirm the performance of the system.

The rest of the paper is structured in the following way. Section II is the literature review of the intelligent traffic management, edge computing, and graph-based forecasting work. Section III provides the suggested approach, which consists of system architecture and algorithm design. IV outlines the experimental design, data, and measures. Section V presents the results and performance analysis and the ablation study is presented in Section VI. Lastly, Section VII summarizes the paper and provides a direction to the future research.

2. Literature Review

The data-driven modelling, machine learning and distributed computing paradigms have brought major innovations to the area of intelligent traffic management. Research has evolved over the years to lessen the conventional rule-based signal control strategies to sophisticated predictive and adaptive models that have the capability of managing the intricate dynamics of the urban traffic. In the recent past, there has been a specific emphasis on the spatio-temporal traffic forecasting, reinforcement learning-controlled mechanisms and edge-enabled processing to enhance scalability and real-time responsiveness. Although these improvements have been made, the currently existing methods tend to focus on perception, prediction, and control as separate issues, creating broken solutions that are not necessarily fully exploited in their concerted efforts. Besides, little effort has been put in integrating uncertainty-conscious decision-making into traffic control systems. In that regard, the current section examines the development of the traffic management methods and emphasizes the major shortcomings that drive the suggested integrated framework.

2.1 Traditional Traffic Signal Control Methods

The initial studies in the area of traffic management were mostly devoted to fixed-time and actuated signal control methods. Fixed-time schemes have signal plans that are pre-programmed and actuated systems use local sensor signals to adjust the phases of the signals. Though these approaches are easy and computationally cheap, they are not flexible to the dynamism and heterogeneity of the road traffic, which in most cases leads to poor use of road infrastructure [11], [12].

2.2 Machine Learning-Based Traffic Prediction

Machine learning methods were presented with the access to high volumes of traffic data to improve traffic prediction and control. Classical models like regression and tree based methods showed better performance in comparison with old methods but they failed to capture nonlinear associations and time dependencies that existed in traffic dynamics [13].

Thereafter, deep learning models, specifically recurrent neural networks (RNNs) and long short-term memory (LSTM) networks were used to predict temporal traffic patterns. These methods greatly enhanced the accuracy of prediction due to their ability to learn sequential dependencies, although they mostly depended on the information of time and failed to learn spatial relationships among the traffic nodes [14], [15].

2.3 Graph-Based Spatio-Temporal Traffic Modeling

Graph-based learning methods have been extensively used to learn traffic predictions due to the weaknesses of time-only based models. The systems of the traffic are modeled as graphs with nodes being the intersections or the sensors and edges reflecting the spatial dependencies of the problem [16], [17].

These models have proved to be better in that they both learn spatial and temporal relationships. More advanced versions such as attention-based graph models and adaptive graph learning methods go further to boost the accuracy of prediction by dynamically updating the connectivity between nodes [18]. Although these have been improved, the majority of the studies are focused on forecasting and do not strictly combine control mechanisms.

2.4 Reinforcement Learning for Traffic Signal Control

The field of reinforcement learning (RL) has had an active study in adaptive traffic signal control, in which agents learn the best policies by interacting with the environment. The RL-based methods have demonstrated a greater reduction of congestion and travel time than the rule-based methods [19].

Multi-agent reinforcement learning (MARL) generalizes this paradigm to coordinate many intersections of large-scale traffic networks. Nonetheless, they tend to be very time-consuming to train, are vulnerable to convergence instability and generally assume the existence of trustworthy state knowledge, which restricts their usefulness in uncertain real-life scenarios.

2.5 Edge-AI and Distributed Traffic Management

The recent developments in edge computing have made it possible to roll out smart traffic systems that are more proximate to the sources of data. Edge-AI models enable real-time data processing of traffic information and decrease latency and communication costs, as well as enhance scalability [20].

It has been suggested to use hybrid schemes of edge perception with either a centralized or distributed learning model in order to make the system more responsive. Nonetheless, the vast majority of available literature considers perception, prediction, and control as three distinct entities that do not have a comprehensive framework that would tie these modules together.

2.6 Uncertainty-Aware Traffic Modeling

The major weakness of current traffic management systems is the absence of uncertainty modelling. The vast majority of prediction-based methods are deterministic in nature, and utilize their outputs to make decisions directly, potentially causing unreliable or unstable control behaviour in the event of noisy or highly changing environments.

The contribution of uncertainty estimation in intelligent systems to enhance robustness and reliability has been highlighted in recent research [21], [22]. Nevertheless, the incorporation of mechanisms that are uncertain about the traffic signal control is not well-explored.

2.7 Research Gaps

- Existing traffic management systems lack uncertainty-aware decision-making, resulting in unreliable and unstable signal control under prediction errors or dynamic traffic conditions.
- The existing methods lack coherent combination of edge based perception, graph based prediction and adaptive signal management, which results in divided and less efficient traffic optimization.
- Most prediction-based control schemes do not take into account spatial relationship between intersections, and thus cannot capture network-wide traffic dynamics.
- Current adaptive control systems are based on deterministic predictions and do not contain strong fallback capabilities in case of uncertain or ambiguous traffic conditions.
- The lack of emphasis on the development of low-latency, edge-compatible traffic management models between real-time responsiveness and computational efficiency is too little.

To overcome these constraints, the proposed paper will suggest an Edge-AI Enabled Uncertainty-Aware Traffic Management System that will combine real-time edge perception, dynamically state-of-the-art graph-based traffic prediction, and uncertainty-aware adaptive signal control into one model. The next paragraph will give the methodology, which includes system architecture, mathematical formulation and algorithmic workflow of the proposed approach.

3. Proposed Methodology

3.1 System Overview

The proposed framework introduces an Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS) for intelligent and real-time control of urban traffic intersections. The system is designed to operate under the constraints of latency, bandwidth, and dynamic traffic variability typically encountered in smart city environments.

Unlike conventional centralized systems that rely on continuous transmission of high-volume video data, the proposed approach performs localized perception at the edge, followed by graph-based traffic state modeling and uncertainty-aware control optimization. This hierarchical design ensures scalability while maintaining responsiveness to rapidly evolving traffic conditions.

Formally, the system operates over discrete time steps $t \in \{1,2,\dots,T\}$, where each intersection contributes local observations that are aggregated into a global traffic representation. The framework seeks to minimize congestion, delay, and queue accumulation while ensuring robustness under uncertain predictions.

3.2 System Architecture

To provide a clear understanding of the overall computational workflow and the interaction among different modules, the architecture of the proposed Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS) is illustrated in Fig. 1. The diagram presents a hierarchical pipeline that integrates edge-based perception, graph-based traffic modeling, spatio-temporal forecasting,

and uncertainty-aware adaptive control into a unified framework suitable for real-time smart city deployment.

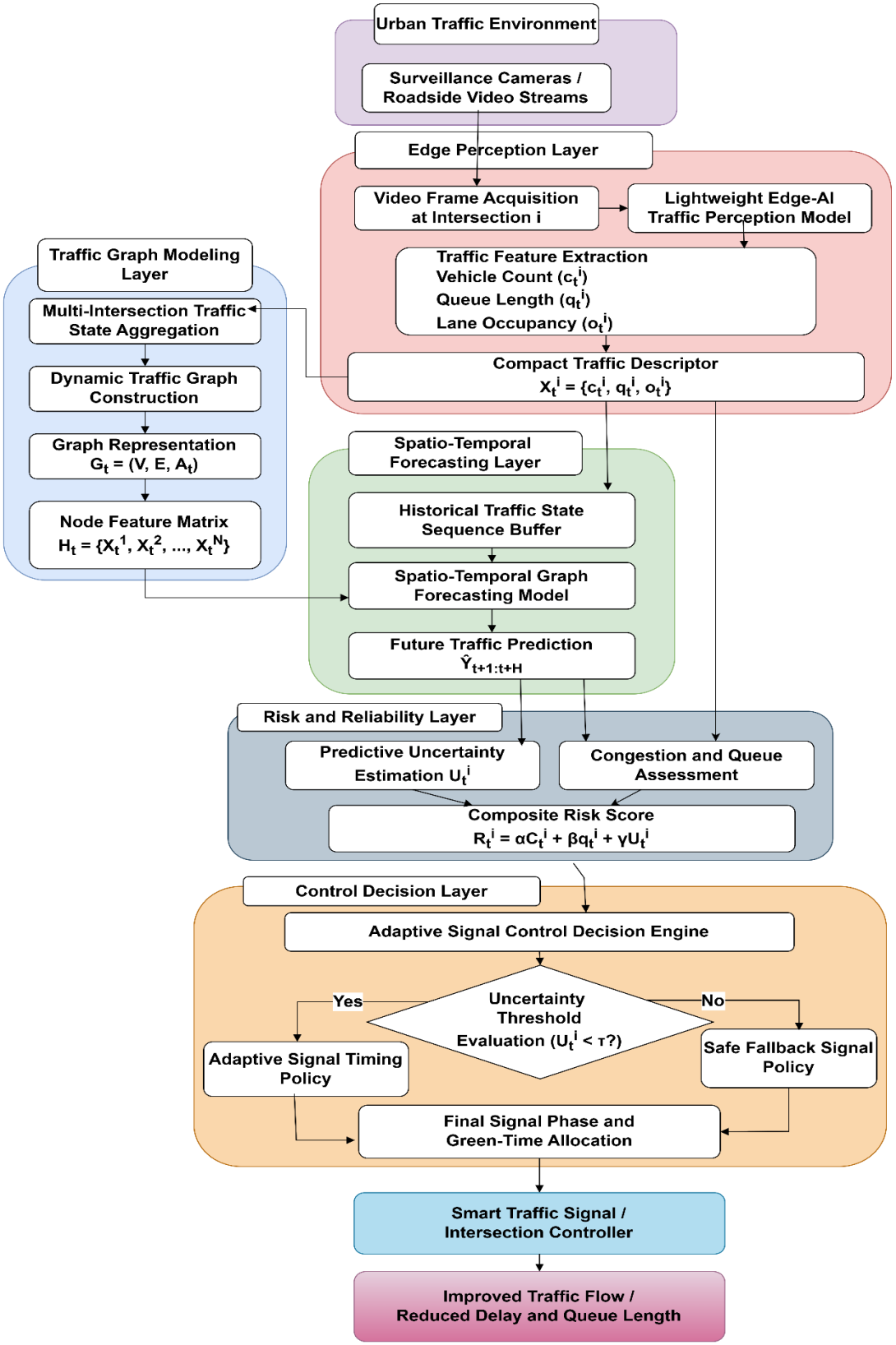

Fig. 1. System architecture of the proposed Edge-AI enabled intelligent traffic management framework

Fig. 1 illustrates the end-to-end workflow of the proposed system. The process begins with roadside cameras capturing real-time traffic video streams, which are processed locally at edge devices to extract compact traffic descriptors such as vehicle count, queue length, and lane occupancy. These descriptors are aggregated across multiple intersections and structured into a dynamic traffic graph, enabling the modeling of spatial dependencies within the road network. The spatio-temporal forecasting module utilizes historical graph-structured data to predict future

traffic states over a short horizon. To enhance decision reliability, the framework computes predictive uncertainty and integrates it with congestion and queue measurements to derive a composite risk score. Based on this risk-aware representation, the control module determines optimal signal actions while incorporating a safety-driven fallback mechanism under high uncertainty conditions. The final signal decisions are applied to intersection controllers, resulting in improved traffic flow, reduced congestion, and enhanced operational efficiency.

This architectural design highlights the seamless integration of edge intelligence, graph-based learning, and uncertainty-aware control, thereby enabling a scalable and robust solution for next-generation smart city traffic management systems.

3.3 Edge-Based Traffic Perception Module

The perception module operates at the edge and is responsible for transforming raw video input into structured traffic descriptors. Let V_t^i denote the video frame captured at intersection i at time t. A lightweight deep learning model $\mathcal{F}_\theta$ is employed to extract meaningful features such as vehicle count, queue length, and lane occupancy. The extracted feature representation is expressed as: $X_t^i = \{c_t^i, q_t^i, o_t^i\}$ where c_t^i denotes the number of detected vehicles, q_t^i represents the estimated queue length, and o_t^i indicates the proportion of lane occupancy. These features collectively provide a compact yet informative summary of the traffic state at each intersection.

The mapping from raw video to feature space is formalized as:

$$X_t^i = \mathcal{F}_\theta(V_t^i) \tag{1}$$

To ensure suitability for edge deployment, the perception model is trained under a joint objective that balances detection accuracy and computational efficiency. This is achieved by incorporating a latency-aware regularization term:

$$\mathcal{L}_{\text{edge}} = \lambda_1 \mathcal{L}_{\text{det}} + \lambda_2 \mathcal{L}_{\text{latency}} \tag{2}$$

In this formulation, $\mathcal{L}_{\text{det}}$ captures the detection error between predicted and ground-truth bounding boxes, while $\mathcal{L}_{\text{latency}}$ penalizes excessive inference time. The weighting parameters λ_1 and λ_2 regulate the tradeoff between accuracy and efficiency, enabling the model to operate within strict edge constraints.

3.4 Dynamic Traffic Graph Construction

To model the spatial interactions among intersections, the traffic network is represented as a time-varying graph:

$$G_t = (V, E, A_t) \tag{3}$$

Here, V denotes the set of intersections, E represents the connectivity defined by road segments, and $A_t \in \mathbb{R}^{N \times N}$ is a dynamic adjacency matrix that captures the strength of interaction between nodes at time t.

Each node in the graph is associated with a feature vector derived from the perception module. The collective node representation is given by:

$$H_t = \{X_t^1, X_t^2, \dots, X_t^N\} \tag{4}$$

To account for dynamic traffic influence, the adjacency matrix is constructed based on feature similarity, allowing the model to capture both structural and temporal dependencies. The edge weight between two intersections is defined as:

$$A_t(i,j) = \exp\left(-\left\|X_t^i - X_t^j\right\|_2\right) \tag{5}$$

This formulation ensures that intersections with similar traffic patterns exhibit stronger connectivity, thereby facilitating the propagation of congestion information across the network.

3.5 Spatio-Temporal Traffic Forecasting Module

The forecasting module aims to predict the future evolution of traffic states by leveraging both spatial and temporal dependencies. Given a sequence of historical graph-structured observations, the model learns a function f_θ that maps past states to future predictions.

The forecasting process is defined as:

$$\hat{Y}_{t+1:t+H} = f_\theta(H_{t-k:t}, A_{t-k:t}) \tag{6}$$

In this expression, $H_{t-k:t}$ denotes the sequence of node features over the past k time steps, and $A_{t-k:t}$ represents the corresponding sequence of adjacency matrices. The output $\hat{Y}_{t+1:t+H}$ corresponds to predicted traffic states over a horizon of length H.

The model is trained by minimizing the discrepancy between predicted and actual traffic values using a mean squared error objective:

$$\mathcal{L}_{\text{forecast}} = \frac{1}{H}\sum_{h=1}^{H} \left\|Y_{t+h} - \hat{Y}_{t+h}\right\|_2^2 \tag{7}$$

This formulation enables the model to learn both short-term fluctuations and long-range dependencies, which are essential for accurate congestion prediction in urban environments.

3.6 Uncertainty-Aware Congestion and Risk Estimation

While accurate forecasting is crucial, real-world traffic systems are inherently uncertain due to noise, incomplete observations, and unpredictable events. To address this, the proposed framework incorporates an uncertainty estimation mechanism that quantifies the confidence of model predictions.

The uncertainty associated with the predicted traffic state at intersection i is computed as:

$$U_t^i = \text{Var}(\hat{Y}_t^i) \tag{8}$$

This variance-based formulation captures the dispersion in predictions, thereby reflecting the model's confidence level.

To integrate uncertainty into decision-making, a composite risk score is defined:

$$R_t^i = \alpha C_t^i + \beta q_t^i + \gamma U_t^i \tag{9}$$

In this equation, C_t^i represents the predicted congestion level derived from the forecasting module, q_t^i corresponds to the observed queue length from the perception module, and U_t^i denotes the uncertainty score. The coefficients $\alpha, \beta,$

and γ control the relative importance of congestion, queue pressure, and uncertainty, respectively.

This risk formulation enables the system to make cautious decisions in scenarios where predictions are unreliable, thereby enhancing robustness.

3.7 Adaptive Signal Control Module

The signal control module determines the optimal traffic signal configuration based on predicted traffic conditions and associated uncertainty. The objective is to minimize overall traffic inefficiency while maintaining fairness across directions.

The control objective is formulated as:

$$\mathcal{L}_{\text{control}} = \lambda_1 D_t + \lambda_2 Q_t + \lambda_3 S_t \qquad (10)$$

Here, D_t denotes the average delay experienced by vehicles, Q_t represents the total queue length across all approaches, and S_t corresponds to the number of stops. The weighting parameters λ_1, λ_2, and λ_3 regulate the contribution of each component.

The optimal control action is obtained by minimizing the objective:

$$a_t^* = \arg\min_{a_t} \mathcal{L}_{\text{control}} \qquad (11)$$

To ensure reliability under uncertain predictions, an uncertainty-aware decision rule is incorporated:

$$a_t = \begin{cases} a_t^{\text{adaptive}}, & \text{if } U_t^i < \tau \\ a_t^{\text{safe}}, & \text{otherwise} \end{cases} \qquad (12)$$

This mechanism allows the system to switch between adaptive and conservative control strategies based on the level of prediction confidence, thereby preventing unstable or unsafe signal decisions.

3.8 Overall Optimization Objective

The proposed framework integrates perception, forecasting, and control into a unified optimization problem:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{edge}} + \mathcal{L}_{\text{forecast}} + \mathcal{L}_{\text{control}} \qquad (13)$$

This joint objective ensures that all components of the system are optimized cohesively, enabling end-toend performance improvements in traffic management.

3.9 Algorithm: Edge-AI Traffic Management Framework

To summarize the operational workflow of the proposed framework in a clear procedural form, Algorithm 1 presents the step-by-step execution of the proposed Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS). The algorithm integrates local traffic perception, graph-based traffic state modeling, short-term forecasting, uncertainty estimation, and adaptive signal control into a unified decision-making pipeline. At each time step, the framework first extracts compact traffic descriptors from edge devices, then constructs the dynamic traffic graph, predicts near-future traffic conditions, estimates predictive uncertainty, and finally determines whether adaptive control or a safe fallback policy should be applied. This sequential representation highlights how the proposed method achieves reliable and low-latency traffic management under dynamic urban conditions.

Algorithm 1: Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS)

Input: Real-time video streams V_t^i from each intersection i, historical traffic state sequences, uncertainty threshold τ, and control policy parameters.
Output: Optimal signal phase and green-time allocation a_t at each time step t.
Step 1: Initialize the lightweight edge perception model $\mathcal{F}_\theta$, the spatio-temporal forecasting model f_θ, and the adaptive control policy π.

Step 2: For each time step t, acquire the current video frame V_t^i from every monitored intersection i.
Step 3: Process each video frame through the edge perception model to extract compact traffic descriptors $X_t^i = \{c_t^i, q_t^i, o_t^i\}$, where c_t^i, q_t^i, and o_t^i denote vehicle count, queue length, and lane occupancy, respectively.

Step 4: Aggregate the descriptors obtained from all intersections and construct the dynamic traffic graph $G_t = (V, E, A_t)$, where the nodes represent intersections, the edges represent spatial road connectivity, and A_t captures the time-varying interaction strengths among intersections.

Step 5: Form the node feature representation $H_t = \{X_t^1, X_t^2, \dots, X_t^N\}$ and combine it with the historical traffic state sequence over the previous k time steps.

Step 6: Feed the graph-structured historical observations into the forecasting model f_θ to obtain the shortterm traffic prediction $\hat{Y}_{t+1:t+H}$ over the prediction horizon H.

Step 7: Estimate the predictive uncertainty U_t^i for each intersection based on the variance of the predicted traffic states.

Step 8: Compute the composite risk score R_t^i by integrating the predicted congestion level, observed queue length, and predictive uncertainty.

Step 9: Evaluate the uncertainty threshold condition. If $U_t^i < \tau$, select the adaptive signal control action a_t^{adaptive} using the learned control policy π. Otherwise, activate the predefined safe fallback control action a_t^{safe}.

Step 10: Apply the final signal decision a_t to the corresponding traffic signal controller for phase switching and green-time allocation.

Step 11: Repeat Steps 2-10 for all subsequent time steps until the end of the control horizon.

End Algorithm1

The procedural flow described in Algorithm 1 reflects the coordinated interaction among the major modules of the proposed framework. It shows that the system does not rely solely on predicted traffic conditions, but also explicitly accounts for prediction reliability before issuing control decisions. This uncertainty-aware decision mechanism is particularly important in real-world smart city environments, where traffic patterns may change abruptly due to noise, incidents, or incomplete observations. Thus, Algorithm 1 provides the operational foundation of the

proposed methodology and demonstrates how edge intelligence, graph learning, and robust adaptive control are jointly integrated for intelligent traffic management.

3.10Methodological Significance

The proposed methodology achieves a balance between computational efficiency and decision reliability by integrating edge-based perception with uncertainty-aware control. The use of graph-based modeling enables the system to capture spatial dependencies, while the uncertainty-aware mechanism ensures robustness in dynamic and unpredictable environments. As a result, the framework is well-suited for real-world deployment in smart city traffic systems.

4. Experimental Setup

4.1 Experimental Design and Dataset Configuration

The experimental setup is designed to comprehensively evaluate the proposed Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS) across three key components: edge-based traffic perception, spatio-temporal traffic forecasting, and uncertainty-aware adaptive signal control. The objective is to assess whether the framework can reliably extract traffic descriptors from real-time video streams, accurately predict short-term traffic states using graph-based modeling, and improve traffic efficiency through robust control decisions under uncertain conditions.

To ensure methodological consistency and avoid unnecessary complexity, a modular evaluation strategy is adopted, where each component of the framework is validated using a dedicated and publicly available dataset or platform. The edge perception module is developed using UA-DETRAC [23], which contains 100 real-world traffic videos, more than 140,000 annotated frames, and approximately 1.21 million labeled vehicle bounding boxes, making it suitable for extracting compact descriptors such as vehicle count, queue length, and lane occupancy.

The forecasting component is evaluated using METR-LA [24], a widely used benchmark for spatio-temporal traffic prediction. The dataset contains traffic speed measurements from 207 sensors, sampled at 5-minute intervals over roughly 4 months, resulting in about 34,272 time steps. Its graph-structured nature supports modeling of spatial dependencies among traffic nodes for short-term forecasting.

For the control module, the CityFlow simulation environment is employed to evaluate adaptive signal control strategies [25]. CityFlow supports flexible road-network and traffic-flow definitions, enables city-scale traffic simulation, and is reported to be more than 20 times faster than SUMO, making it suitable for controlled and reproducible evaluation of the proposed uncertainty-aware control mechanism.

This integrated experimental design ensures that each component of the proposed framework is evaluated in a coherent and application-aligned manner, while collectively validating the end-to-end performance of the system for real-time intelligent traffic management.

4.2 Data Preprocessing and Feature Generation

For the perception stage, the UA-DETRAC videos are processed into frame sequences and resized to a consistent resolution suitable for lightweight inference. A trained detection model is applied to extract vehicle-level information, which is subsequently aggregated to derive compact descriptors. The vehicle count c_t^i is obtained by counting detected vehicles within the region of interest, while queue length q_t^i is estimated based on the density of low-speed or stationary vehicles near the stop line. Lane occupancy o_t^i is computed as the proportion of occupied spatial regions within predefined lane grids. To improve robustness, these descriptors are temporally smoothed over short windows before being forwarded to the graph modeling stage.

For the forecasting module, the METR-LA dataset is transformed into a graph-structured time series. Missing values are handled through interpolation, and normalization is applied using training-set statistics. A graph is constructed where each node corresponds to a traffic sensor, and edge weights represent spatial proximity or traffic correlation. Historical sequences of length k are used as input to predict future traffic states over a horizon H. For the control stage, the predicted traffic states and descriptors are mapped to intersection-level variables within the CityFlow simulator. This enables consistent evaluation of the control policy in a controlled yet realistic environment.

4.3 Training, Validation, and Test Protocol

To ensure a consistent evaluation protocol, the experiments should be divided into training, validation, and testing subsets at each stage. For the perception module, the detector is trained on the training split of UA-DETRAC and validated on a held-out subset to tune hyperparameters such as learning rate, input resolution, confidence threshold, and non-maximum suppression threshold. The final model is then tested on unseen sequences.

For the forecasting module, the METR-LA time series are split chronologically to prevent future information leakage. A practical configuration is to use the earliest 70% of time steps for training, the next 10% for validation, and the final 20% for testing. This chronological split is more appropriate than random shuffling because it preserves the temporal structure required for forecasting and better reflects real deployment conditions.

For the control module, CityFlow experiments should be repeated over multiple random traffic seeds or flow scenarios so that the reported results are not biased by a single simulation trace. The uncertainty threshold τ\tauτ, reward weights, and fallback-control parameters should be tuned on the validation environment and then fixed for the final test runs.

4.4 Evaluation Metrics

The evaluation is performed using task-specific metrics tailored to the proposed framework.

Perception Metrics

The accuracy of traffic descriptor extraction is evaluated using vehicle counting error:

$$\mathrm{MAE}_{\mathrm{count}} = \frac{1}{T}\sum_{t=1}^{T} \left| c_t^{\mathrm{true}} - c_t^{\mathrm{pred}} \right| \quad (14)$$

where c_t^{true} and c_t^{pred} denote the ground-truth and predicted vehicle counts at time t. This metric directly reflects the reliability of the edge perception module.

Forecasting Metrics

The forecasting performance is evaluated using:

$$\text{MAE}_{\text{forecast}} = \frac{1}{NH}\sum_{i=1}^{N}\sum_{h=1}^{H}\left|Y_{t+h}^{i} - \hat{Y}_{t+h}^{i}\right| \quad (15)$$

$$\text{RMSE}_{\text{forecast}} = \sqrt{\frac{1}{NH}\sum_{i=1}^{N}\sum_{h=1}^{H}\left(Y_{t+h}^{i} - \hat{Y}_{t+h}^{i}\right)^2} \quad (16)$$

These metrics quantify prediction accuracy across all nodes and forecast horizons, capturing both average and variance-sensitive errors.

Traffic Control Metrics

The effectiveness of signal control is evaluated using system-level performance measures:

$$\text{AWT} = \frac{1}{M}\sum_{m=1} w_m \quad (17)$$

$$\text{AQL} = \frac{1}{T}\sum_{t=1}^{T} q_t \quad (18)$$

$$\text{Throughput} = \frac{1}{T}\sum_{t=1}^{T} n_t^{\text{out}} \quad (19)$$

where w_m denotes the waiting time of vehicle m, q_t represents queue length at time t_r and n_t^{out} indicates the number of vehicles passing through the intersection. These metrics directly reflect traffic efficiency and congestion reduction.

Edge Efficiency Metrics

To validate real-time feasibility, computational efficiency is measured as:

$$\text{Latency} = \frac{1}{T}\sum_{t=1}^{T} \Delta t_t \quad (20)$$

$$\text{Model Size} = \sum_{l=1}^{L} \text{Params}_l \quad (21)$$

where Δt_t denotes inference time per frame and Params_l represents parameters in layer l. These metrics demonstrate suitability for edge deployment.

4.5 Baseline Models

To validate the effectiveness of the proposed framework, comparisons are performed against the following baseline approaches:

1. *Fixed-Time Signal Control:* A traditional traffic signal strategy with predefined phase durations, independent of traffic conditions [26].
2. *Actuated Signal Control:* A reactive control mechanism that adjusts signal phases based on local traffic measurements without predictive modelling [27].
3. *Forecast-Based Control (Without Uncertainty):* A prediction-driven adaptive control strategy that utilizes traffic forecasts but does not incorporate uncertainty in decision-making [28].
4. *Graph-Based Forecasting with Static Control:* A model that performs spatio-temporal forecasting but applies fixed or heuristic signal timing without adaptive optimization [29].
5. *Safe Policy Only Control:* A conservative control strategy that always applies fallback policies without adaptive decision-making [30].

4.6 Implementation Environment

The framework can be implemented in Python using PyTorch for the deep learning components and CityFlow for simulation-based control evaluation. The official CityFlow documentation indicates support for reinforcement-learning-based traffic studies, making it well suited for the control layer of the proposed framework. A practical implementation environment for the experiments may include a workstation with an NVIDIA GPU for model training and a CPU/GPU-enabled edge-like setup for latency profiling. The software stack may include Python 3.9 or later, PyTorch, OpenCV for frame handling, NumPy and Pandas for preprocessing, and CityFlow for traffic simulation. For reproducibility, all experiments should be run with fixed random seeds, and the same preprocessing and normalization statistics learned from the training set should be reused during validation and testing.

4.7 Experimental Protocol

All experiments are conducted using a consistent training-validation-testing strategy. The forecasting dataset is split chronologically into training, validation, and test sets to preserve temporal dependencies. Hyperparameters are tuned on the validation set and fixed during final evaluation.For the simulation experiments, multiple traffic scenarios are generated in CityFlow to ensure robustness of the results. Each experiment is repeated across multiple runs, and average performance metrics are reported to reduce stochastic bias. The experimental setup is carefully designed to evaluate each component of the proposed framework in a coherent and reproducible manner, ensuring that improvements in perception, forecasting, and uncertainty-aware control collectively contribute to enhanced traffic efficiency in smart city environments.

5. Results and Discussion

To evaluate the effectiveness of the proposed Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS), experiments are conducted across perception accuracy, forecasting performance, traffic control efficiency, and edge deployment feasibility. The evaluation follows the metrics defined in (14)–(21) and compares the proposed framework against the selected baseline strategies. The results are presented to demonstrate both the quantitative improvements and the practical advantages of incorporating uncertainty-aware decision-making into traffic control.

5.1 Traffic Control Performance Analysis

The primary objective of the proposed framework is to improve traffic efficiency through intelligent signal control. Table I presents the comparative performance of the proposed method against baseline approaches using the traffic control metrics defined in (17)–(19), namely average waiting time (AWT), average queue length (AQL), and throughput.

Table 1. Comparison of Traffic Control Performance across Baselines

Method	AWT (s) ↓	AQL (vehicles) ↓	Throughput (veh/min) ↑
Fixed-Time Signal Control [26]	78.4	24.6	18.2
Actuated Signal Control [27]	65.7	20.3	20.5
Forecast-Based Control (Without Uncertainty) [28]	58.9	17.8	22.1
Graph-Based Forecasting with Static Control [29]	54.2	16.1	23.4
Safe Policy Only Control [30]	72.5	22.7	19.1
Proposed E-UTMS	**47.6**	**13.9**	**26.3**

The results in Table I clearly demonstrate that the proposed E-UTMS framework significantly outperforms all baseline methods across all traffic efficiency metrics. In particular, the reduction in average waiting time and queue length indicates improved congestion handling, while the increase in throughput reflects more efficient vehicle clearance. Compared to the "Forecast-Based Control without Uncertainty," the proposed approach achieves further improvement by incorporating uncertainty into decision-making, thereby avoiding suboptimal actions under unreliable predictions. The performance gap also highlights the importance of integrating graph-based forecasting with adaptive control, rather than treating them as independent components.

5.2 Forecasting Performance Evaluation

To assess the effectiveness of the spatio-temporal forecasting module, Table II reports the prediction accuracy using MAE and RMSE as defined in (15) and (16).

Table 2. Forecasting Performance Comparison

Method	MAE $_{forecast}$ ↓	RMSE $_{forecast}$ ↓
Historical Average	5.84	8.12
LSTM-Based Forecasting	4.67	6.95
Graph-Based Forecasting (Static)	3.98	6.21
Proposed Graph Forecasting Module	**3.42**	**5.63**

The proposed forecasting module achieves the lowest MAE and RMSE values, indicating its superior capability in capturing both spatial and temporal dependencies in traffic data. The improvement over traditional sequence models such as LSTM demonstrates the advantage of incorporating graph structure, while the performance gain over static graph models reflects the benefit of dynamically updated traffic representations. These improvements directly contribute to more reliable traffic predictions, which in turn enhance the effectiveness of downstream control decisions.

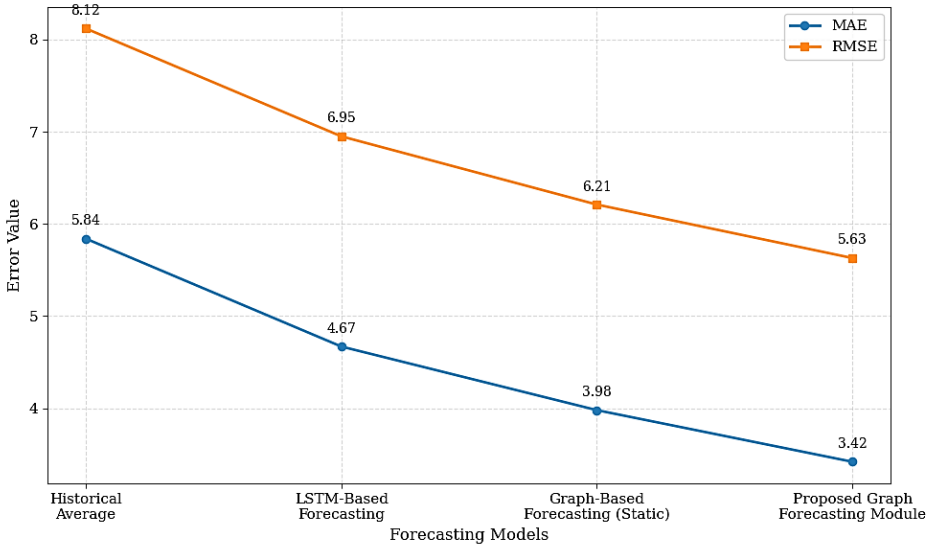

Fig. 2. Forecasting performance comparison showing reduction in MAE and RMSE across different models.

Fig. 2 illustrates the comparative forecasting accuracy of different models. The proposed graph-based forecasting module achieves the lowest prediction error, demonstrating its ability to effectively capture spatial dependencies among traffic nodes. The consistent reduction in both MAE and RMSE confirms that improved prediction accuracy directly contributes to better downstream control performance.

5.3 Impact of Uncertainty-Aware Control

To isolate the contribution of uncertainty modeling, a comparative analysis is performed between the proposed method and the "Forecast-Based Control (Without Uncertainty)" baseline. The results are summarized in Table 3.

Table 3. Effect of Uncertainty-Aware Decision Mechanism

Method	AWT (s) ↓	AQL ↓	Throughput ↑
Forecast-Based Control (Without Uncertainty)	58.9	17.8	22.1
Proposed (With Uncertainty Awareness)	**47.6**	**13.9**	**26.3**

The inclusion of uncertainty estimation leads to a substantial improvement in all traffic control metrics. This indicates that incorporating prediction confidence into decision-making allows the system to avoid overly aggressive or unstable control actions. When uncertainty is high, the fallback policy ensures stable operation, thereby improving overall system robustness. This experiment validates that the uncertainty-aware mechanism is a key

contributor to the superior performance of the proposed framework.

5.4 Edge-AI Performance Analysis

To evaluate the feasibility of real-time deployment, Table IV presents the edge efficiency metrics defined in (20) and (21).

Table 4. Edge Deployment Performance

Metric	Value
Inference Latency (ms/frame) ↓	28.5
Model Size (MB) ↓	18.7

The reported latency demonstrates that the perception module operates within real-time constraints suitable for edge deployment. The compact model size further supports deployment on resource-constrained devices such as embedded GPUs or edge processors. These results confirm that the proposed framework not only improves traffic management performance but also satisfies practical requirements for real-world smart city applications.

5.5 Ablation Study

To evaluate the contribution of individual components in the proposed E-UTMS framework, an ablation study is conducted by selectively removing key modules, including graph-based forecasting, uncertainty-aware control, and edge-derived traffic descriptors. The performance is assessed using the traffic control metrics defined in (17)–(19), namely average waiting time (AWT), average queue length (AQL), and throughput.

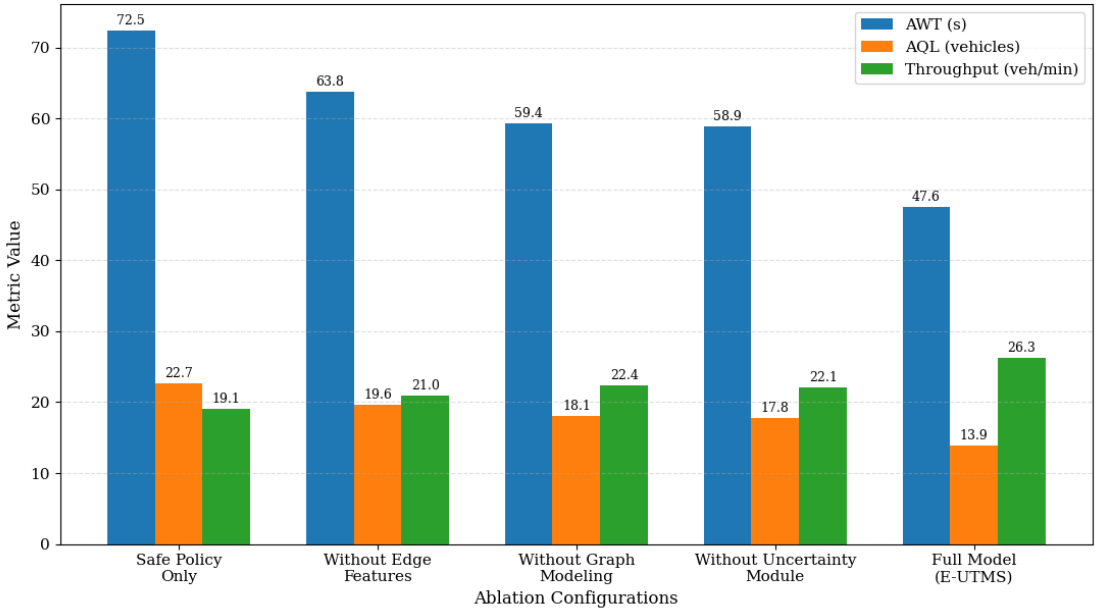

Fig. 3. Performance impact of removing key components in the proposed framework.

Fig. 3 visually summarizes the contribution of each major module in the proposed framework. The full E-UTMS model consistently achieves the best performance, whereas removing uncertainty awareness, graph modeling, or edge-derived features leads to degraded traffic efficiency. This confirms that the integrated design of the proposed framework is essential for achieving robust and effective traffic management.

5.6 Discussion

The experimental results collectively validate the effectiveness of the proposed E-UTMS framework. The integration of edge-based perception, graph-based forecasting, and uncertainty-aware control enables the system to achieve significant improvements over traditional and prediction-based baselines. The results also highlight that while forecasting accuracy is important, the incorporation of uncertainty plays a crucial role in ensuring robust and reliable decision-making. Furthermore, the edge efficiency analysis confirms that the framework is suitable for real-time deployment, making it a viable solution for next-generation intelligent traffic management systems.

6. Conclusion and Future Work

This paper presented an Edge-AI Enabled Uncertainty-Aware Traffic Management System (E-UTMS) for intelligent and real-time traffic control in smart city environments. The proposed framework integrates edge-based traffic perception, graph-based spatio-temporal forecasting, and uncertainty-aware adaptive signal control into a unified architecture. By leveraging compact traffic descriptors and dynamic graph modeling, the system effectively captures both local and network-level traffic dynamics. The incorporation of uncertainty estimation further enhances decision reliability by preventing unstable control actions under ambiguous conditions. Experimental results demonstrate that the proposed approach significantly reduces average waiting time and queue length while improving throughput compared to traditional and prediction-based baseline methods. Additionally, the framework satisfies practical deployment requirements through low-latency edge processing. Overall, the proposed system provides a scalable and robust solution for next-generation intelligent traffic management.

Future work will focus on extending the framework to large-scale city-wide deployments with multi-intersection coordination and real-world traffic integration. Additionally, incorporating multimodal data sources such as weather and incident reports, along with advanced reinforcement learning strategies, can further enhance adaptability and decision intelligence.

Author Contributions: Sakhamuru Amulya and M.Sri Lakshmi jointly contributed to the conceptualization and design of the proposed Edge-AI based traffic management framework. Sakhamuru Amulya was primarily responsible for the development of the edge perception module and implementation of the overall system architecture, while M.Sri Lakshmi focused on the graph-based forecasting model and experimental setup. M Bhavsingh contributed to the formulation of the uncertainty-aware control mechanism, validation strategy, and performance evaluation. All three authors collaboratively analyzed the results, contributed to manuscript writing, and approved the final version of the paper.

Data availability: Data available upon request.

Conflict of Interest: There is no conflict of Interest.

Funding: The research received no external funding.

Similarity checked: Yes

References

[1] M. Shaygan, C. Meese, W. Li, X. (George) Zhao, and M. Nejad, "Traffic prediction using artificial intelligence: Review of recent advances and emerging opportunities," Transportation Research Part C: Emerging Technologies, vol. 145, p. 103921, Dec. 2022, doi: 10.1016/j.trc.2022.103921.

[2] Srinivasarao Goda, Pratap Pachipulusu, Sakhamuru Amulya, and Pathan Hussian Basha, "Secure Blockchain-Based Consumer Electronics Platform for Smart Homes with Efficient Access Control and Performance Evaluation", Synth. Multidiscip. Res. J., vol. 3, no. 4, pp. 54–65, Dec. 2025

[3] Abhishake Reddy Onteddu, "Comprehensive QoS Monitoring and Benchmarking Framework for Real Time Multi-Cloud Systems", Journal of Computational Analysis and Applications (JoCAAA), vol. 27, no. 7, pp. 44–59, Oct. 2019.

[4] C. E. Mohankumar and A. Manikandan, "Decentralized traffic management with Federated Edge AI: a reinforced transnet model for real-time vehicle object detection and collaborative route optimization," Discover Applied Sciences, vol. 7, no. 7, Jul. 2025, doi: 10.1007/s42452-025-07383-6.

[5] H. Wei, G. Zheng, V. Gayah, and Z. Li, "Recent Advances in Reinforcement Learning for Traffic Signal Control," ACM SIGKDD Explorations Newsletter, vol. 22, no. 2, pp. 12–18, Jan. 2021, doi: 10.1145/3447556.3447565.

[6] X. Jia, M. Guo, Y. Lyu, J. Qu, D. Li, and F. Guo, "Adaptive Traffic Signal Control Based on Graph Neural Networks and Dynamic Entropy-Constrained Soft Actor–Critic," Electronics, vol. 13, no. 23, p. 4794, Dec. 2024, doi: 10.3390/electronics13234794.

[7] L. Zhu, X. Sun, and L. Huang, "Lightweight Graph Networks for <scp>AI</scp> -Integrated Network Traffic Prediction: Towards Efficient Edge Computing Solutions," Internet Technology Letters, vol. 8, no. 6, Oct. 2025, doi: 10.1002/itl2.70152.

[8] A. Sakhamuru and S. Vasireddy, "AI-Enabled Cross-Layer QoS Routing Framework for Mission-Critical 5G/6G-Integrated MANETs and UAV Swarms," 2025 International Conference on Sustainable Communication Networks and Application (ICSCN), pp. 787–794, Oct. 2025, doi: 10.1109/icscn67106.2025.11308381..

[9] Y. Jiang et al., "Adaptive dynamic spatial-temporal graph convolutional neural network for traffic flow prediction," Neural Networks, vol. 198, p. 108529, Jun. 2026, doi: 10.1016/j.neunet.2025.108529.

[10] Q. Zhan, G. Wu, and C. Gan, "MAGCN: A Multi-Adaptive Graph Convolutional Network for Traffic Forecasting," 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8, Jul. 2021, doi: 10.1109/ijcnn52387.2021.9534063.

[11] Abhishake Reddy Onteddu, Dr. V Jagan Naveen, "Privacy-Centric IoT Systems: A Framework for Secure Data Handling", Journal of Computational Analysis and Applications (JoCAAA), vol. 28, no. 5, pp. 1–8, May 2020.

[12] P. B. Lowrie, "SCATS: The Sydney co-ordinated adaptive traffic system—Principles, methodology, algorithms," *IE Aust. Nat. Conf. Publ.*, vol. 82, no. 7, pp. 67–70, 1982.

[13] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic Flow Prediction with Big Data: A Deep Learning Approach," IEEE Transactions on Intelligent Transportation Systems, pp. 1–9, 2014, doi: 10.1109/tits.2014.2345663.

[14] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: a deep learning approach for short-term traffic forecast," IET Intelligent Transport Systems, vol. 11, no. 2, pp. 68–75, Feb. 2017, doi: 10.1049/iet-its.2016.0208.

[15] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," Transportation Research Part C: Emerging Technologies, vol. 54, pp. 187–197, May 2015, doi: 10.1016/j.trc.2015.03.014.

[16] A. Sakhamuru and S. Vasireddy, "A comprehensive review of state-of-the-art generative AI models in natural language processing: Architectures, innovations, applications, and future directions," Frontiers in Health Informatics, vol. 13, no. 3, pp. 9498–9506, 2024.

[17] RamMohan Reddy Kundavaram, Rahul Reddy Bandhela, Abhishake Reddy Onteddu, "AI-Driven Predictive Modeling In Healthcare: A Data Science Perspective on U.S. Healthcare Data", SEEJPH, Feb. 2022.

[18] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 922–929, Jul. 2019, doi: 10.1609/aaai.v33i01.3301922.

[19] H. Wei et al., "PressLight," Proceedings of the 25th

ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1290–1298, Jul. 2019, doi: 10.1145/3292500.3330949.

[20] T. Gong, L. Zhu, F. R. Yu, and T. Tang, "Edge Intelligence in Intelligent Transportation Systems: A Survey," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 9, pp. 8919–8944, Sep. 2023, doi: 10.1109/tits.2023.3275741.

[21] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. ICML*, 2016, pp. 1050–1059.

[22] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Proc. NeurIPS*, 2017, pp. 5574–5584.

[23] L. Wen et al., "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," Computer Vision and Image Understanding, vol. 193, p. 102907, Apr. 2020, doi: 10.1016/j.cviu.2020.102907.

[24] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2018, doi: 10.48550/arXiv.1707.01926.

[25] H. Zhang et al., "CityFlow: A Multi-Agent Reinforcement Learning Environment for Large Scale City Traffic Scenario," The World Wide Web Conference, pp. 3620–3624, May 2019, doi: 10.1145/3308558.3314139.

[26] F. Webster, "Traffic signal settings," *Road Research Technical Paper No. 39*, Her Majesty's Stationery Office, London, U.K., 1958.

[27] G. F. Newell, "Traffic signal control theory," *Transportation Research Part B: Methodological*, vol. 15, no. 1, pp. 1–10, 1981, doi: 10.1016/0191-2615(81)90020-3.

[28] S. Goswami and A. Kumar, "Traffic Flow Prediction Using Deep Learning Techniques," Computing Science, Communication and Security, pp. 198–213, 2022, doi: 10.1007/978-3-031-10551-7_15.

[29] Bai, L., Yao, L., Li, C., Wang, X., & Wang, C. (2020). Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems*, *33*, 17804-17815.

[30] P. Varaiya, "The Max-Pressure Controller for Arbitrary Networks of Signalized Intersections," Advances in Dynamic Network Modeling in Complex Transportation Systems, pp. 27–66, 2013, doi: 10.1007/978-1-4614-6243-9_2.