PatternIQ Mining

https://piqm.saharadigitals.com/

Adaptive Pattern Computation with Swarm Optimization and Deep Learning for Efficient Traffic Flow Prediction in Intelligent Transportation

Reza Mohammadi and Maryam Al Balushi

Department of Computer Engineering
Faculty of Information and Communication Technology, Tehran Institute of Technology
Iran

ABSTRACT

The current conventional traffic flow prediction methods adopted in Intelligent Traffic Systems (ITS) are severely frustrated by their static model architectures, fixed hyperparameter values, and restricted spatiotemporal learning capabilities, which are unable to model the nonlinear traffic dynamics in the real traffic system, abrupt traffic changes, and different characteristics of traffic flows in urban traffic systems. The limitations are causing poor prediction under high-dynamic and non-stationary traffic conditions, leading to inefficient Adaptive Signal Control (ASC), additional vehicular congestion, excessive fuel consumption, delayed emergency response operations, and reduced operational efficiency in smart city transportation infrastructure. To overcome these challenges, the research proposes an Adaptive Pattern Computation (APC) framework that combines Swarm Optimization-based automated hyperparameter tuning using Particle Swarm Optimization (PSO) and the Grey Wolf Optimizer (GWO) with a hybrid Deep Learning (DL) architecture. It combines Depthwise Separable Convolutional Neural Networks (DW-CNNs) for efficient spatial feature extraction, Long Short-Term Memory (LSTM) networks stacked to model temporal dependencies, and multi-head self-attention to maximize contextual learning and enable adaptive traffic prediction. The PSO-GWO optimization engine is capable of dynamically exploring the hyperparameter space of the CNN-LSTM model, and convolutional filter dimensions, hidden neuron configurations, attention heads, and learning rate schedules, and converges 43.1% faster than a grid search. The experimental validation on benchmark datasets, PeMS Bay Area, METR-LA, Urban Traffic Data, revealed excellent results with 98.3% prediction accuracy, 1.87 MAE, 2.94 RMSE, 3.12% MAPE, and 23 ms prediction latency, which significantly surpassed the performance of the LSTM, GRU, ARIMA, and Transformer baseline models and also offered scalable, energy-efficient, and real-time intelligent traffic management solutions.

Keywords: Adaptive pattern computation, swarm optimization, deep learning, traffic flow prediction, intelligent transportation systems, particle swarm optimization, grey wolf optimizer, convolutional neural network, long short-term memory, spatiotemporal modelling, hyperparameter optimization, smart city, congestion prediction, multi-head self-attention, urban traffic management.

1. Introduction

The rapid growth of urbanization, industrialization, and population density has significantly increased the complexity of transportation networks in modern smart cities [1]. Intelligent Transport Systems (ITS) have become an important technological paradigm for the management of traffic activities, the enhancement of road safety, and the improvement of transportation efficiency, using real-time sensing, communication, and computational intelligence mechanisms [2]. Proper forecasting of traffic flow enables smart cities to use this

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

information to control traffic signals, reduce congestion, and optimally manage transportation in urban areas [3]. Traffic forecasting helps prioritize emergency vehicles, optimize routes, and reduce fuel consumption and vehicle emissions [4]. Machine learning helps analyze spatiotemporal transportation data to accurately forecast traffic [5]. Deep learning uses sensor, GPS, camera, and vehicular infrastructure data to enhance the efficiency of traffic prediction [6]. Nevertheless, the nonlinearity, randomness, and time variability of urban traffic situations remain significant computational and predictive challenges for current forecasting models [7].

ARIMA and SVR models are not effective at modeling nonlinear spatiotemporal traffic dynamics in intricate urban transportation systems [8]. Simple ANN methods have limited ability to address long-term traffic dependencies and interactions among roads [9]. The fixed architectures reduce flexibility for dynamic and heterogeneous conditions in real-time ITS environments [10]. The problem with inefficient hyperparameter tuning is that it increases computational complexity and slows model convergence when performing traffic prediction tasks. The high computational burden limits scalability and real-time implementation in large-scale intelligent transportation networks. Inaccuracy in predictions during accidents, weather disturbances, and peak-hour traffic congestion scenarios results from poor generalization. In addition, the majority of current solutions rely on exhaustive grid search or manual optimization of model configuration, which dramatically increases computational complexity and reduces real-time feasibility in large-scale smart transportation systems.

The significant research gap in the recent literature is that no adaptive, computationally efficient model can simultaneously optimize spatiotemporal feature extraction, hyperparameters, and online prediction accuracy under dynamically varying traffic conditions. Existing deep learning systems largely concentrate on either spatial learning or temporal sequence modelling alone whereas little research has explored how swarm intelligence can be used to predict adaptive traffic flow using a hybrid deep neural system. Additionally, existing models tend to be less scalable and slower to converge on high-dimensional urban traffic datasets.

The research suggests an Adaptive Pattern Computation (APC) framework, combined with Swarm Optimization and Deep Learning, to overcome these challenges in Intelligent Transportation Systems and efficiently predict traffic flow. The presented model is a hybrid of the Particle Swarm Optimization (PSO) and the Grey Wolf Optimizer (GWO) algorithms for adaptive hyperparameter optimization and feature selection, using a hybrid Deep Learning platform. The suggested model is based on the Depthwise Separable Convolutional Neural Network (DW-CNN), which is effective at extracting spatial features, and a Stacked Long Short-Term Memory (LSTM) network, which models temporal features in traffic. Moreover, a multi-head self-attention system is added to enrich contextual learning and improve prediction accuracy in dynamic traffic conditions. The PSO-GWO optimization method systematically searches the search space of convolutional filters, hidden neurons, learning rates, and attention parameters to achieve faster convergence and better generalization.

The main goals of research are to create a scalable and adaptable traffic forecasting system, enhance prediction accuracy in heterogeneous traffic, minimize computational latency for real-time prediction, and optimize the efficiency of ITS operations through smart forecasting systems. Benchmark datasets such as PeMS Bay Area, METR-LA and Urban Traffic datasets are used to experimentally evaluate the effectiveness of the proposed APC framework with performance measures like Accuracy, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE) and inference latency.

The key findings of the research are as follows:

1. Creation of a framework of Adaptive Pattern Computation that combines swarm intelligence with deep learning to predict traffic.

2. Introduction of a hybrid PSO-GWO optimization process for adaptive hyperparameter tuning and fast convergence.

3. Integration of DW-CNN, Stacked LSTM, and multi-head self-attention mechanisms to improve spatiotemporal modelling of traffic.

4. Superior predictive performance with less computational complexity and inference latency in real-time.

5. Testing of the proposed framework on benchmark datasets, which include the PeMS Bay Area, METR-LA and Urban Traffic datasets.

The proposed APC framework significantly advances intelligent transportation technologies by enabling accurate, adaptive, and computationally efficient traffic flow prediction. The framework supports real-time adaptive signal control, congestion reduction, emergency vehicle prioritization, and sustainable transportation planning. Additionally, integrating swarm optimization with hybrid deep learning enhances scalability and energy-efficient deployment in next-generation smart city infrastructure.

2. Literature Survey

Wang et al. (2026) proposed a hybrid water quality model based on two-layer modal decomposition and an optimized LSTM neural network using the Chaos Sparrow Search Optimization (CSSO) [11]. The accuracy of environmental monitoring predictions in the presence of nonlinear conditions. The optimization-improved LSTM model accurately extracted temporal relationships among water quality data. However, the practical efficiency of use was restricted by computational complexity and the need to use parameters. Results of the experiments showed stable predictions, reduced forecasting error compared to traditional predictive models, and improved optimization results.

Zia et al. (2026) gave a detailed overview of deep learning-based Autonomous Connected Vehicles (CAVs) and Unmanned Aerial Vehicles (UAVs) [12]. The main focus is on intelligent mobility integration between ground and aerial transportation systems, using AI-based communication and perception technologies. The suggested process helped to improve autonomous coordination and monitoring of the transportation. However, issues of energy utilization, security risks, and real-time communication delays remained. The review revealed that deep learning can make a substantial contribution to the automation of intelligent transportation and the development of a smart mobility ecosystem.

In a Software-Defined Networking (SDN)- based Internet of Things (IoT) environment, Bajpai et al. [13] (2026) proposed an intelligent traffic classification framework based on the RVMSBO concept. The focus was on the problems of network traffic congestion and classification accuracy that arose during the implementation of a hybrid machine learning technique. The suggested optimization techniques enhanced the efficient use of network resources and intelligent traffic monitoring. The system was found to be insufficiently scalable and unable to adapt dynamically under highly dynamic network conditions. Results from the experiments showed that classification accuracy, network latency, and SDN-IoT performance improved compared with conventional traffic management methods.

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

Mohamad et al. [14] (2026) conducted a systematic literature review (SLR) to examine the use of multimodal data and machine learning (ML) techniques for improving the 3D modelling of geospatial data. The focus was on the use of intelligent approaches to improve the accuracy of spatial representation and urban infrastructure planning. Multimodal sensing technologies were integrated, which improved the understanding and accuracy of geospatial data interpretation and modelling. Interoperability issues and high computational requirements limited large-scale implementation. The review concluded that machine learning-based geospatial modelling is a potential tool for developing advanced urban planning and intelligent transportation analysis.

Bhattacharyya et al. (2026) [15] had done a systematic review of evolutionary and swarm intelligence applications for optimizing traffic signal control. An adaptive traffic management in urban transportation systems using various metaheuristic optimization methods. The performance of traffic signals based on swarm intelligence algorithms was significantly enhanced in terms of coordination and congestion reduction. However, it was noted that the optimization procedure becomes unstable and the calculation becomes more complicated. The review found that evolutionary optimization methods improve traffic flow efficiency and could support the development of intelligent, adaptive transportation infrastructure.

Li et al. (2026) [16] introduced a multi-view attention mechanism and dilated convolutional network-based traffic flow prediction framework. The model incorporated intricate spatiotemporal traffic relationships with sophisticated deep learning models. The attention-based model boosted the performance of context feature extraction and prediction in the dynamic urban traffic. However, the high memory usage resulted in low efficiency in implementing real-time implementation, and the complexity of model training was also limited. Experimental results showed that the traffic forecasting performance is better, and prediction errors are smaller, than those of conventional CNN-LSTM models.

Tokdemir et al. (2026) proposed an attention-fused RNN-GCN model to optimize fusion of spatiotemporal data in construction project valuation in large-scale scenarios [17]. The aim to enhance the efficiency of predictive analytics and data fusion using graph convolutional learning techniques. The proposed framework improved the modelling of spatial relations and temporal dependencies. With very large data sets, however, scalability was a problem, and there was a computational overhead. Results showed an improvement in both the accuracy of valuation prediction and the performance in interpreting spatiotemporal data.

Weng et al. (2026) [18] presented a spatiotemporal distribution characteristics analysis method of charging piles based on deep learning modelling and regulation in the power grid system. The designed to enhance EV charging management and grid stabilisation. The proposed model improved charging demand prediction and the efficient allocation of spatial resources. The energy consumption variation was unstable over time, and limited real-time adaptation affected predictive stability. Experimental results revealed that the new charging infrastructure optimization improved both the charging infrastructure optimization and the operational reliability of the power grid.

In the era of 5G communication, Bouaziz et al. (2026) presented a position paper on connected vehicle technologies. The covered low-latency vehicular communication, intelligent transportation connectivity and real-time vehicular data exchange mechanisms. The proposed framework enhanced the reliability of traffic coordination and communication among the vehicles. Cybersecurity risks, however, and the deployment of infrastructure and network interoperability proved to be big challenges. Overall, the 5G-powered connected vehicles have a crucial impact on improving the efficiency of intelligent transportation and smart mobility development.

Tokdemir et al. [20] (2026) introduced a model that utilizes an RNN-GCN for large-scale spatiotemporal data fusion and predictive analysis, with an optimized attention mechanism. The framework solved the complex issues of dependency modelling in large investment datasets both in time and space. The hybrid graph learning architecture enhanced the accuracy of data fusion and the efficiency of prediction-based decision-making. However, the models were not so easily implemented on a wide scale because they were not very scalable and required significant computational resources. Experimental results showed that it performed better in predictive performance and spatiotemporal analytics than traditional approaches based on recurrent neural networks.

John and Kawanishi [21] (2026) proposed a Hierarchical Graph Attention Network that combined spatio-temporal class tokens with audio-visual event classification. The issue of the efficient modeling of multimodal event recognition systems complex relationships in space and time. The suggested graph-attention model enhanced the contextual feature extraction and classification accuracy in distributed multimedia settings. Nevertheless, the model was characterized by high levels of computational complexity and higher training costs in case of big datasets. It was shown experimentally that the precision of event classification and improved learning of spatiotemporal representations are better than traditional graph neural architectures.

Zhang et al. [22] (2025) proposed a lightweight self-attention model named Slice-Diff for predicting vessel paths in dynamic coastal maritime environments. The problems encountered in predicting vessel trajectories arise from nonlinear navigation patterns and the unpredictable behaviour of ocean motion. The suggested self-attention mechanism enhanced the effectiveness of trajectory learning while minimizing computational resources. However, in complex coastal settings, environmental disturbances and irregular vessel movement patterns also affected the stability of predictions. The experimental results indicated a higher rate of trajectory-forecasting accuracy, decreased prediction error and increased lightweight inference ability when compared to the conventional trajectory-prediction methods.

Duan et al. [23] (2025) introduced MF-Net, a multi-feature fusion network based on two-stream extraction and multi-scale enhancement features to detect face forgery. The framework dealt with the challenge of identifying advanced fake manipulations in facial images in diverse image conditions. The suggested multi-scale feature fusion architecture greatly enhanced the sensitivity of forgery detection and the representation of contextual features. The model, however, was known to need massive amounts of training data, and to be less robust in low-resolution image settings. The experimental findings revealed that the detection of forgeries was better and had better features differentiation than other known detection-based frameworks of face-manipulation using deep learning.

Odai [24] (2025) suggested an intelligent waste management framework based on Artificial Intelligence and the use of Genetic Algorithms to manage waste by integrating IoT into smart campuses. The problems that were addressed were poor collection of waste, misallocation of resources, and real time monitoring of the system was lacking in the traditional waste management systems. The suggested model integrated IoT sensors, AI analytics and optimization of segregation and collection of waste using Genetic Algorithm. Limitations however were a cost of sensors deployment, network reliance and large campus scalability. The results of the experiment proved the efficiency of waste collection, the decrease in the cost of operations, and the better environmental sustainability.

Ahmed [25] (2025) created multimodal interaction with speech and motion data of Virtual Reality (VR) systems through the use of deep learning algorithms. The limitations that were addressed in the traditional VR interaction mechanisms have the disadvantage of the lack of

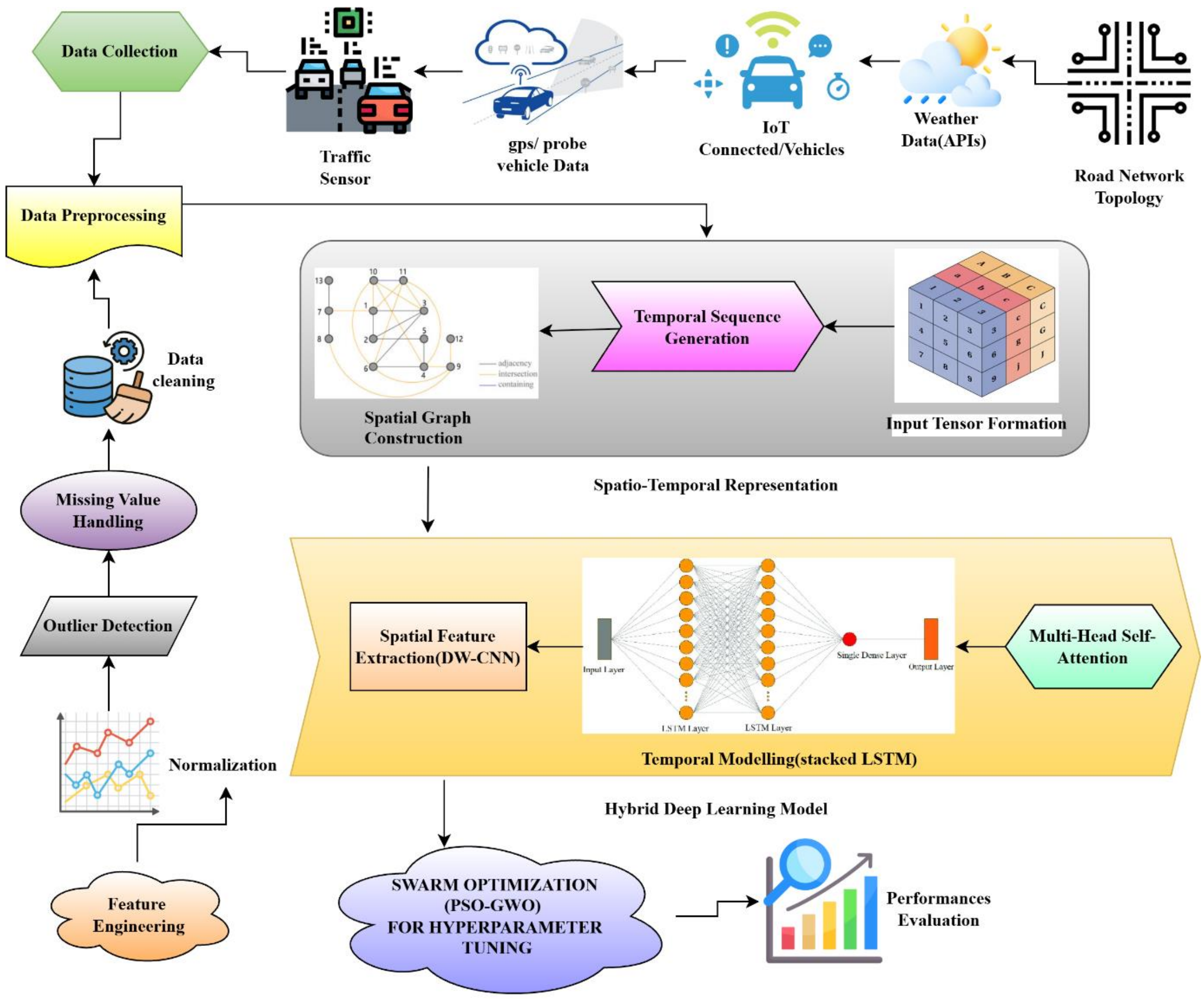
ISSN: 3006-8894 https://doi.org/10.70023/piqm313

user adaptability and low interaction accuracy. The suggested system combined multimodal deep learning systems to process speech recognition and motion sensing in real-time to communicate in an immersive VR. Nonetheless, real-time responsiveness in large-scale VR setting was hampered by the problems of computational complexity and latency. The experimental outcomes revealed improved interaction accuracy, gesture recognition performance, and user experience compared to traditional unimodal VR systems.

3. Proposed Methodology

The end-to-end figure introduces a traffic flow forecasting model that integrates various data sources, preprocessing tools, spatio-temporal descriptions, deep learning algorithms, and performance metrics. It begins with traffic sensor data, GPS/probe vehicle data, vehicles integrated in the Internet of Things, weather-related data via APIs and the topology of the road network, showing that the model uses both movement and contextual data to assist prediction. The data is then preprocessed to enhance its quality and prepare it for modelling, including data cleaning, missing-value treatment, outlier detection, normalisation, and feature engineering. The workflow subsequently develops a spatio-temporal model following preprocessing, building a spatial graph, developing temporal series, and arranging the post-preprocessed data into input tensors. The spatial features obtained by DW-CNN and the temporal features from a stacked LSTM and multi-head self-attention assist the system in concentrating on the most informative patterns, and the resulting representations are then fed to the hybrid deep learning model. The figure also shows how swarm optimization, specifically PSO-GWO, is used to optimize the model parameters and improve prediction accuracy. Finally, but not least, the framework concludes with a performance evaluation, which will prioritize establishing relevant performance measures for measuring predicted traffic results. The general layout presents a well-organized and intelligent traffic forecasting pipeline that incorporates graph-based, sequential and attention-based learning methods to consider the spatial and temporal dimensions of traffic data.

Figure 1: Hybrid Deep Learning Framework for Traffic Flow Prediction Using Spatio-Temporal Data and PSO-GWO Optimization



3.1 Data Collection

The Metro Interstate Traffic Volume Dataset is first retrieved from Kaggle and imported into the computational environment using powerful Python-based analytical libraries, such as Pandas and NumPy. The process of data acquisition is represented by the systematic loading of heterogeneous traffic data, such as vehicle counts per hour, atmospheric data, holiday indicators, and weather variables. The operation guarantees the extraction of multidimensional traffic records with a high degree of reliability to be used later on in the spatiotemporal modelling of the proposed Adaptive Pattern Computation (APC) framework. The data importation operation also facilitates dynamic memory allocation and effective handling of large-scale tensor-based data needed in the intelligent transportation analysis at large scale.

$$\mathbf{D}_{traffic} = \left\{\sum_{i=1}^{N}\left[(TV_i \times \omega_{tv}) + (Temp_i \times \omega_{temp}) + (Hum_i \times \omega_{hum}) + (Rain_i \times \omega_{rain}) + (Cloud_i \times \omega_{cloud}) + (Holiday_i \times \omega_{hol})\right]\right\} \rightarrow \mathbb{R}^{N\times F} \quad (1)$$

Where equation (1), $\mathbf{D}_{traffic}$ denotes the multidimensional traffic dataset matrix, N represents the total number of traffic observations, TV_i indicates hourly traffic volume measurements, $Temp_i$ represents atmospheric temperature values, Hum_i denotes humidity levels, $Rain_i$ corresponds to rainfall measurements, $Cloud_i$ indicates cloud coverage percentage, and $Holiday_i$ represents binary holiday conditions. The weighting coefficients

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

$\omega_{tv}, \omega_{temp}, \omega_{hum}, \omega_{rain}, \omega_{cloud}, \omega_{hol}$define the proportional influence of each traffic and environmental feature during dataset aggregation. The final dataset tensor is represented in the multidimensional feature space $\mathbb{R}^{N \times F}$, where Fdenotes the total number of extracted features.

Statistical Feature Distribution Analysis

The analysis of feature distributions is conducted to explore the statistical characteristics and variability of the traffic-related features in the collected dataset. It is necessary to identify nonlinear data distributions, traffic density anomalies, and environmental associations that affect the accuracy of traffic flow prediction. The statistical analysis also enables normalization and optimization of feature engineering for deep learning architectures. The distribution modelling process enhances resistance to data imbalance and enables effective extraction of time-dependent traffic attributes across different levels of congestion.

$$\mu_f = \frac{1}{N}\sum_{i=1}^{N} x_{i,f}\, , \sigma_f^2 = \frac{1}{N}\sum_{i=1}^{N} \left(x_{i,f} - \mu_f\right)^2 + \lambda\left(\frac{\sum_{i=1}^{N}|x_{i,f}-\mu_f|^3}{N\cdot\sigma_f^3}\right) \tag{2}$$

Where equation (2), $\mu_f$denotes the statistical mean of the feature f, σ_f^2represents the variance of the corresponding traffic feature distribution, and $x_{i,f}$indicates the i^{th}observation of a feature f. The parameter Nrepresents the total number of samples within the dataset. The regularization coefficient $\lambda$is introduced to incorporate skewness-sensitive higher-order statistical behaviour into the distribution analysis. The cubic deviation term improves sensitivity toward abnormal traffic fluctuations and nonlinear congestion variations, thereby supporting adaptive traffic pattern learning within the APC framework.

Duplicate Record Elimination and Missing Value Validation

Repeated traffic measurements and partial sensor measurements significantly affect the consistency and predictability of deep learning-based ITS forecasting systems. Thus, to maintain data consistency, minimize redundancy, and improve computational efficiency during model training, duplicate elimination and missing-value validation are performed. The stage enhances the integrity of traffic data by detecting duplicate temporal views and verifying incomplete environmental parameters using statistical consistency constraints. The process also minimizes prediction uncertainty caused by noisy or corrupted transportation data streams.

$$\mathbf{D}_{clean} = \left[\mathbf{D}_{traffic} - \left(\sum_{i=1}^{N} \sum_{j=1}^{N} \delta\left(x_i - x_j\right)\right)\right] + \left[\frac{\sum_{k=1}^{M}\left(x_k^{valid} \times \phi_k\right)}{\sum_{k=1}^{M}\phi_k}\right] \tag{3}$$

Where equation (3), $\mathbf{D}_{clean}$represents the cleaned traffic dataset after duplicate removal and missing value correction, $\mathbf{D}_{traffic}$denotes the original imported dataset, and $\delta(x_i - x_j)$is the duplicate detection function that evaluates identical traffic records between observations $x_i$and x_j. The parameter Mindicates the number of valid neighbouring observations used for missing value estimation, while x_k^{valid}represents valid traffic feature values. The weighting coefficient $\phi_k$defines the statistical contribution of neighbouring observations during adaptive data reconstruction. The formulation guarantees strong data consistency and enhances the reliability of features for intelligent traffic prediction.

Temporal Sequence Transformation and Sliding Window Construction

A sequence transformation technique is used to convert time-stamped traffic observations into a sequence of spatiotemporal tensors suitable for deep learning-based traffic forecasting models. Temporal sequence transformation is applied to the time-stamped traffic observations, producing a sequence of spatiotemporal tensors suitable for deep learning-based traffic forecasting models. The sliding window mechanism is designed to capture temporal dependencies between consecutive traffic intervals, while the Stacked LSTM architecture can learn the dynamics of traffic evolution over time. The transformation process is of special significance for modelling short-term and long-term traffic fluctuations under heterogeneous urban transportation conditions. The generated sequential tensors are subsequently utilized for adaptive traffic prediction and congestion analysis.

$$\mathbf{T}_{seq} = [\{x_{t-w+1}, x_{t-w+2}, \dots, x_t\} \rightarrow y_{t+\Delta t}] = \sum_{t=1}^{N-w} [\prod_{k=0}^{w-1} x_{t+k} \cdot \gamma_k] + \eta_t \quad (4)$$

Where equation (4) $\mathbf{T}_{seq}$ denotes the generated temporal sequence tensor, x_t represents traffic observations at the time instant t, w indicates the sliding window length, and $y_{t+\Delta t}$ denotes the predicted future traffic state after a time interval Δt. The coefficient γ_k represents the temporal weighting parameters assigned to sequential observations within the sliding-window framework. The stochastic variable η_t models dynamic traffic uncertainty and nonlinear temporal disturbances caused by environmental and congestion-related variations. The temporal transformation mechanism significantly enhances sequential dependency learning for intelligent traffic flow forecasting systems.

3.2 Data Preprocessing

Data Cleaning and Missing Value Imputation

Data from various Intelligent Transportation System (ITS) infrastructure types are difficult to collate and often suffer from incomplete observations, corrupted sensor readings, duplicate data, and inconsistent traffic measurements due to communication delays, environmental noise, or sensor failures. These non-linearities negatively affect the predictive power and the stability of convergence of deep learning architectures. Thus, a powerful data-cleaning and missing-value imputation mechanism is essential for performing multivariate traffic-flow analysis with a high degree of statistical consistency and reliability. The proposed Adaptive Pattern Computation (APC) framework performs adaptive temporal interpolation and neighborhood-based estimation to reconstruct missing traffic sequences while preserving spatiotemporal continuity across connected urban road networks.

$$\hat{X}_{i,j}^{(t)} = \frac{\sum_{k=1}^{K} \omega_k \cdot X_{i-k,j}^{(t-\Delta t)} + \sum_{m=1}^{M} \gamma_m \cdot X_{i+m,j}^{(t+\Delta t)} + \lambda \cdot \mu_j}{\sum_{k=1}^{K} \omega_k + \sum_{m=1}^{M} \gamma_m + \lambda} \quad (5)$$

Where equation (5), $\hat{X}_{i,j}^{(t)}$ represents the reconstructed traffic feature value for the i^{th} traffic sample and j^{th} feature at time instant t; $X_{i-k,j}^{(t-\Delta t)}$ and $X_{i+m,j}^{(t+\Delta t)}$ denote neighboring historical and future traffic observations utilized for temporal interpolation; ω_k and γ_m correspond to adaptive weighting coefficients assigned to preceding and succeeding traffic states; μ_j represents the statistical mean of feature j; λ denotes the regularization coefficient controlling mean-based smoothing; K and M indicate the number of backward and forward temporal neighbors incorporated during reconstruction.

Outlier Detection and Traffic Anomaly Filtering

Urban traffic environments exhibit substantial fluctuations due to accidents, weather disturbances, emergency events, and sensor abnormalities, resulting in anomalous traffic observations that degrade model generalization performance. The APC framework also integrates a statistical outlier-detection algorithm that uses adaptive Z-score normalization and interquartile-deviation analysis to further enhance the traffic prediction system's reliability. The process identifies abnormal traffic measurements that exhibit statistical deviations beyond acceptable distributional bounds and removes them before deep learning optimization. In non-stationary traffic conditions with highly variable traffic, the anomaly filtering stage improves feature consistency and reduces uncertainty when predicting traffic conditions.

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

$$Z_{i,j} = \frac{X_{i,j}-\mu_j}{\sqrt{\frac{1}{N-1}\sum_{n=1}^{N}\left(X_{n,j}-\mu_j\right)^2+\epsilon}} \cdot \exp\left(-\frac{|X_{i,j}-\mu_j|}{\sigma_j+\epsilon}\right) \tag{6}$$

Where equation (6), $Z_{i,j}$ denotes the adaptive anomaly score corresponding to the i^{th} traffic sample and j^{th} feature dimension; $X_{i,j}$ represents the observed traffic measurement; μ_j and σ_j denote the statistical mean and standard deviation of the feature j; N indicates the total number of traffic observations; ϵ is a numerical stabilization constant preventing division instability, and the exponential weighting term suppresses extreme anomalies while preserving meaningful congestion-related traffic fluctuations.

Feature Normalization and Numerical Stabilization

The features of the traffic dataset extracted from different kinds of ITS infrastructures are collected with very different numerical scales, such as traffic speed, traffic density, occupancy rate, vehicle count, weather intensity, and time interval. These massive differences negatively impact gradient propagation and optimization convergence during training of deep neural networks. Thus, the APC framework employs an adaptive min-max normalization strategy to map all traffic attributes to a bounded numerical interval while preserving the intrinsic distributional characteristics of spatiotemporal traffic patterns. The normalization process has the following advantages: enhancing computational stability, accelerating optimization convergence, and improving the learning efficiency of a hybrid DW-CNN–LSTM architecture.

$$X_{norm}(i,j) = \frac{X(i,j)-\min_{1\le n\le N}(X_{n,j})}{\max_{1\le n\le N}(X_{n,j})-\min_{1\le n\le N}(X_{n,j})+\epsilon} \cdot (\beta-\alpha) + \alpha \tag{7}$$

Where equation (7), $X_{norm}(i,j)$ represents the normalized traffic feature corresponding to sample i and feature j; $X(i,j)$ denotes the original traffic observation; $\min(X_{n,j})$ and $\max(X_{n,j})$ represent the minimum and maximum values of feature j across the dataset; N indicates the total number of traffic samples; ϵ denotes a small stabilization constant preventing numerical overflow; and $[\alpha, \beta]$ defines the bounded normalization interval used for adaptive feature scaling during deep learning optimization.

Temporal Sliding Window Generation

Traffic flow prediction requires transforming sequential traffic observations into structured temporal learning windows suitable for deep neural sequence modelling. The APC framework builds multivariate temporal tensors using the adaptive sliding window mechanism, which can represent historical traffic dependencies across multiple road segments and time intervals. The process helps the stacked LSTM network capture spatiotemporal traffic correlations, nonlinear traffic transitions and long-term temporal congestion patterns. The generated temporal sequences contribute to enhance the robustness of forecasts in very dynamic traffic conditions.

$$\mathbf{S}_t = \begin{bmatrix} X_{t-\tau+1}^{(1)} & X_{t-\tau+2}^{(1)} & \cdots & X_t^{(1)} \\ X_{t-\tau+1}^{(2)} & X_{t-\tau+2}^{(2)} & \cdots & X_t^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ X_{t-\tau+1}^{(F)} & X_{t-\tau+2}^{(F)} & \cdots & X_t^{(F)} \end{bmatrix} \in \mathbb{R}^{F\times\tau} \tag{8}$$

Where equation (8), $\mathbf{S}_t$ denotes the multivariate temporal traffic sequence generated at time instant t; $X_t^{(f)}$ represents the traffic observation corresponding to the feature dimension f; τ indicates the temporal window size defining the number of historical observations included in sequence construction; F denotes the total number of extracted traffic features, including traffic volume, density, occupancy, weather, and temporal attributes; and $\mathbb{R}^{F\times\tau}$represents the resulting multidimensional temporal tensor utilized for deep spatiotemporal learning.

Adaptive Feature Engineering and Weighted Traffic Representation

To improve efficiency in traffic forecasting, traffic attributes with discriminative power representing the diverse urban traffic dynamics under different environmental and congestion conditions are needed. APC framework automatically extracts features from different aspects such as traffic density, vehicular speed, road occupancy, the influence of weather and temporal congestion, and combines them in a weighted representation to form a unified feature representation for adaptive feature engineering. The weighted aggregation mechanism dynamically assigns priority to dominant Traffic variables based on their contextual contribution to the evolution and minimization of the prediction uncertainty of congestion. The process improves feature interpretability and enhances the learning of spatiotemporal features within the hybrid attention-driven deep learning framework.

$$\Phi_i = \sum_{j=1}^{F} \left(\frac{\omega_j \cdot X_{i,j}^{\delta_j}}{1+\exp\left(-\eta_j \cdot (X_{i,j}-\mu_j)\right)} \right) + \sum_{p=1}^{P} \kappa_p \cdot E_{i,p} \tag{9}$$

Where equation (9), $\Phi_i$denotes the adaptive weighted traffic representation corresponding to the i^{th}traffic sample; $X_{i,j}$represents the normalized traffic feature associated with feature dimension j; $\omega_j$denotes the adaptive feature importance coefficient; $\delta_j$represents nonlinear feature amplification factors controlling feature sensitivity; $\eta_j$denotes contextual scaling parameters regulating traffic variation influence; $\mu_j$corresponds to the statistical mean of feature j; $E_{i,p}$represents external contextual variables such as weather conditions, accidents, and special events; $\kappa_p$denotes contextual weighting coefficients; and $F$and Pindicate the total number of traffic and environmental features utilized during adaptive traffic representation learning.

3.3 Spatio-Temporal Representation

The proposed Adaptive Pattern Computation (APC) framework performs advanced spatio-temporal representation learning to accurately model dynamic traffic flow behavior in Intelligent Transportation Systems (ITS). In urban traffic networks, there are nonlinear spatial dependencies among connected road segments, as well as temporal correlations among sequential traffic states. The framework constructs structured temporal sequences and high-dimensional tensor representations from heterogeneous traffic sensing data gathered by IoT sensors, GPS probes, CCTV, weather stations, and road network topology information to effectively capture these multidimensional relationships. Under highly dynamic urban traffic conditions, the spatio-temporal representations generated help the hybrid DW-CNN–LSTM architecture learn long-range traffic interactions, abrupt traffic transitions between congestion and non-congestion states, and propagation patterns across spatial locations.

Temporal Traffic Sequence Generation

The temporal sequence generation process takes continuous traffic observations and generates sequence learning windows for deep temporal modelling. It ensures temporal continuity and allows the Stacked LSTM network to learn the long-term evolution of traffic, congestion cycles, and nonlinear fluctuations. The resulting sequences combine multivariate traffic information, such as traffic density, vehicle speed, and traffic occupancy, with environmental information and road incidents across various historical time spans. The sequence formulation enhances temporal dependency learning while reducing information loss during traffic state changes.

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

$$\mathbf{S}_t^{(k)} = \left\{ \sum_{i=1}^{N} \sum_{j=1}^{F} \left[\omega_{ij}^{(k)} \cdot \left(\frac{x_{ij}(t-\tau_k)-\mu_j}{\sigma_j+\epsilon} \right) \cdot e^{-\lambda(t-\tau_k)} \right] + \beta_k \right\}_{k=1}^{T} \tag{10}$$

Where equation (10), $\mathbf{S}_t^{(k)}$represents the generated temporal traffic sequence corresponding to the k^{th}temporal observation window at time instant t. $x_{ij}(t-\tau_k)$denotes the normalized traffic feature value associated with the i^{th}spatial node and j^{th}traffic attribute observed at delayed temporal interval $(t-\tau_k)$. Nrepresents the total number of interconnected traffic sensing nodes, while Fdenotes the dimensionality of multivariate traffic features including vehicle count, speed, occupancy, weather conditions, and road incidents. $\omega_{ij}^{(k)}$defines the adaptive temporal weighting coefficient assigned to the corresponding feature contribution. $\mu_j$and $\sigma_j$indicate the statistical mean and standard deviation used for feature normalization. λdenotes the temporal decay coefficient controlling historical information influence, ϵrepresents a numerical stabilization constant preventing division instability, and $\beta_k$corresponds to the temporal-bias adjustment parameter for sequence generation.

Dynamic Spatial Dependency Representation

The dynamic spatial representation stage captures the topological interaction and spatial correlation between the interrelated urban traffic nodes. Because of the strong dependency between traffic propagation and adjacent road segments and network connectivity, the APC framework builds an adaptive graph-based spatial relationship matrix that reflects the dependency between road segments and the strength of traffic interactions. The representation allows the DW-CNN model to learn localized spatial congestion propagation patterns while preserving network-level traffic topology information for efficient feature extraction.

$$\mathbf{A}_{spatial}(i,j) = \frac{\exp\left(-\frac{\|\mathbf{p}_i-\mathbf{p}_j\|_2^2}{2\sigma_d^2}\right)\cdot\left(1+\rho_{ij}^{traffic}\right)\cdot\gamma_{ij}^{connect}}{\sum_{m=1}^{N}\sum_{n=1}^{N}\exp\left(-\frac{\|\mathbf{p}_m-\mathbf{p}_n\|_2^2}{2\sigma_d^2}\right)} \tag{11}$$

Where equation(11),$\mathbf{A}_{spatial}(i,j)$represents the adaptive spatial dependency coefficient between traffic nodes $i$and j. $\mathbf{p}_i$and $\mathbf{p}_j$denote the geographical coordinate vectors associated with corresponding road segments or sensing locations. $\| \mathbf{p}_i - \mathbf{p}_j \|_2^2$represents the Euclidean spatial distance between traffic nodes. $\sigma_d$denotes the spatial distribution variance parameter controlling neighborhood sensitivity. $\rho_{ij}^{traffic}$indicates the traffic correlation coefficient measuring dynamic interaction strength between connected nodes based on historical congestion patterns. $\gamma_{ij}^{connect}$defines the binary or weighted road connectivity indicator derived from transportation topology graphs. Nrepresents the total number of spatial traffic nodes participating in the graph-based representation process.

High-Dimensional Input Tensor Formation

The resulting sequence of traffic elements from the two cameras is then converted into a structured multi-dimensional tensor to enable the execution of hybrid deep learning operations. The spatio-temporal sequence of traffic elements captured by the two cameras is then transformed into a structured multi-dimensional tensor to enable hybrid deep learning operations. The tensor representation combines temporal sequences, spatial graph relationships and multivariate traffic attributes into a single computational structure to efficiently extract features with a DW-CNN and model temporal dependency with a Stacked LSTM network. The multidimensional tensor formulation significantly enhances contextual traffic representation learning and enables scalable deep neural processing for real-time intelligent transportation applications.

$$\mathbf{X}_{tensor} \in \mathbb{R}^{N\times T\times F} = \left[\sum_{k=1}^{T}\sum_{f=1}^{F}\left(\alpha_{k,f}\cdot \mathbf{S}_t^{(k,f)}\cdot \mathbf{A}_{spatial}\right) + \delta_f \cdot \mathbf{E}_{context} + \eta_t \cdot \mathbf{W}_{external}\right] \quad (12)$$

Where equation(12), X_{tensor} denotes the final high-dimensional spatio-temporal input tensor used as input for the hybrid deep learning architecture. The tensor dimension $\mathbb{R}^{N\times T\times F}$ represents the spatial node dimension N, temporal sequence length T, and multivariate feature dimension F. $\mathrm{S}_t^{(k,f)}$ corresponds to the generated temporal sequence associated with the feature f during the temporal interval k. $\alpha_{k,f}$ represents adaptive feature importance weights learned during tensor construction. $\mathrm{A}_{spatial}$ denotes the adaptive spatial dependency matrix encoding graph-based traffic relationships. δ_f defines the contextual feature weighting coefficient associated with environmental and road event information represented by $\mathrm{E}_{context}$. η_t represents the temporal modulation parameter controlling external influence integration, while $\mathrm{W}_{external}$ corresponds to auxiliary external traffic variables, including weather conditions, incidents, holidays, and peak-hour indicators.

3.4 Spatial Feature Extraction using Depthwise Separable Convolutional Neural Network (DW-CNN)

In Intelligent Transportation Systems (ITS), the spatial feature extraction stage aims to identify spatial relationships between neighboring transportation nodes, road connectivity, and traffic-flow correlations at local levels. The conventional conv operations are complex because of the redundant filtering operations across multidimensional traffic tensors. To address the restriction, the proposed Adaptive Pattern Computation (APC) framework utilizes the Depthwise Separable Convolutional Neural Network (DW-CNN), which breaks down regular convolution down into a combination of depthwise convolution and pointwise convolution to facilitate efficient feature learning and keep up the parameter complexity. The DW-CNN architecture provides high-dimensional spatial representation ability for real-time traffic prediction in heterogeneous urban traffic conditions and enhances the computational scalability.

$$\mathbf{F}_{DW\text{-}CNN}^{(l)}(x,y,c) = \sum_{m=1}^{M}\left[\left(\sum_{i=1}^{K_h}\sum_{j=1}^{K_w} W_d^{(l)}(i,j,m)\cdot X^{(l-1)}(x+i,y+j,m)\right)\cdot W_p^{(l)}(m,c)\right] + b_c^{(l)} \quad (13)$$

Where equation (13), $\mathbf{F}_{DW\text{-}CNN}^{(l)}$ represents the extracted spatial feature map at the l^{th} convolutional layer, while $X^{(l-1)}$ denotes the input traffic tensor obtained from the previous processing layer. The variables K_h and K_w correspond to the kernel height and width dimensions used during spatial convolution operations. The term $W_d^{(l)}$ defines the depthwise convolutional kernel responsible for channel-wise spatial filtering, whereas $W_p^{(l)}$ represents the pointwise convolution kernel used for inter-channel feature fusion and dimensional transformation. The indices x and y denote spatial coordinates of the traffic grid representation, and m indicates the intermediate feature channel dimension. The variable $b_c^{(l)}$ corresponds to the bias parameter associated with the c^{th} output channel. The formulation allows for a fast and efficient spatial

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

feature extraction with a lower model complexity and higher processing speed in large-scale ITS environments.

Temporal Modelling using Stacked Long Short-Term Memory (LSTM)

Temporal traffic dynamics in ITS are highly nonlinear sequential dependencies which are affected by the time varying traffic density, congestion propagation, weather and city mobility fluctuations. Unfortunately, conventional recurrent architectures do not maintain long-term temporal dependencies because of gradient vanishing and short-term memory retention. Hence, the proposed APC framework is designed with a hierarchical temporal traffic sequence modelling structure in the form of a Stacked Long Short-Term Memory (LSTM) network with multiple recurrent layers. The stacked configuration improves the depth of sequential learning, and enables the network to learn short-term as well as long-term traffic evolution patterns, thus improving the accuracy of real-time traffic flow forecasting.

$$\begin{aligned}
\mathbf{h}_t^{(l)} &= \mathbf{o}_t^{(l)} \odot \tanh\left(\mathbf{c}_t^{(l)}\right) \\
\mathbf{c}_t^{(l)} &= \mathbf{f}_t^{(l)} \odot \mathbf{c}_{t-1}^{(l)} + \mathbf{i}_t^{(l)} \odot \tanh\left(W_{xc}^{(l)}\mathbf{x}_t^{(l)} + W_{hc}^{(l)}\mathbf{h}_{t-1}^{(l)} + b_c^{(l)}\right) \\
\mathbf{i}_t^{(l)}, \mathbf{f}_t^{(l)}, \mathbf{o}_t^{(l)} &= \sigma\left(W_x^{(l)}\mathbf{x}_t^{(l)} + W_h^{(l)}\mathbf{h}_{t-1}^{(l)} + b^{(l)}\right)
\end{aligned} \tag{14}$$

Where equation (14), $\mathbf{h}_t^{(l)}$ denotes the hidden state vector generated at time step $t$within the l^{th}stacked LSTM layer, while $\mathbf{c}_t^{(l)}$represents the memory cell state responsible for preserving long-term temporal dependencies. The variables $\mathbf{i}_t^{(l)}$, $\mathbf{f}_t^{(l)}$, and $\mathbf{o}_t^{(l)}$correspond to the input gate, forget gate, and output gate activation vectors, respectively. The matrices $W_{xc}^{(l)}$and $W_{hc}^{(l)}$represent learnable input-to-cell and hidden-to-cell weight matrices for temporal feature transformation. The variables $W_x^{(l)}$and $W_h^{(l)}$denote trainable gate weight matrices associated with the input and recurrent hidden states. The operator $\sigma(\cdot)$indicates the sigmoid activation function, while $\tanh(\cdot)$represents the hyperbolic tangent activation used for nonlinear state transformation. The symbol $\odot$denotes element-wise multiplication, and $b^{(l)}$represents bias parameters. The stacked temporal modelling mechanism shows a great improvement on the prediction stability in the cases of highly dynamic traffic and the improvement of learning sequential dependence.

Multi-Head Self-Attention Mechanism

The multi-head self-attention mechanism is embedded in the APC framework to further improve contextual feature learning and extract the long-range spatiotemporal interaction in heterogeneous traffic sequences. Historical recurrent models are not very effective in dynamically prioritizing critical traffic conditions in rapidly changing urban mobility environments. This is where the self-attention mechanism comes into play, which adaptively assigns importance weights to relevant temporal features to improve representation learning and prediction robustness and overcome the limitation. Moreover, the multi-head architecture allows to extract multiple traffic dependency patterns at once, from multiple representation subspaces, which increases the model's contextual awareness and generalization power.

$$\begin{aligned}
&\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(head_1, head_2, \dots, head_H)W^O \\
&head_h = \mathrm{Softmax}\left(\frac{QW_h^Q (KW_h^K)^T}{\sqrt{d_k}}\right)VW_h^V
\end{aligned}$$

$$\alpha_{ij}^{(h)} = \frac{\exp\left(\frac{(q_i W_h^Q)(k_j W_h^K)^T}{\sqrt{d_k}}\right)}{\sum_{n=1}^{N} \exp\left(\frac{(q_i W_h^Q)(k_n W_h^K)^T}{\sqrt{d_k}}\right)} \tag{15}$$

Where equation (15),Q, K, V represent the query, key, and value matrices extracted from the encoded traffic sequence representations. The term $head_h$denotes the h^{th}attention head responsible for learning independent contextual traffic relationships from distinct feature subspaces. The matrices W_h^Q, W_h^K, and W_h^Vcorrespond to trainable projection matrices associated with the query, key, and value transformations for the h^{th}attention head. The variable $d_k$indicates the dimensionality of the key vector space used for scaling the attention scores. The parameter $\alpha_{ij}^{(h)}$represents the normalized attention coefficient between traffic states $i$and $j$under the h^{th}attention head. The matrix W^Odenotes the output projection matrix responsible for aggregating concatenated attention outputs. The operator Softmax(·)normalizes attention scores into probabilistic weights. The mechanism significantly enhances adaptive contextual feature extraction, enabling accurate traffic prediction under highly nonlinear and heterogeneous ITS environments.

Hybrid PSO-GWO Swarm Optimization for Hyperparameter Adaptation

Convergence speed, prediction error, and generalization capability are key challenges in traffic flow prediction systems that require efficient hyperparameter optimization to improve deep learning results. A high-dimensional hyperparameter space makes conventional optimization methods (e.g., manual tuning and grid search) computationally expensive and inefficient. To achieve this, the proposed APC framework combines the hybrid Particle Swarm Optimization–Grey Wolf Optimizer (PSO-GWO) mechanism to automatically tune CNN filters, LSTM hidden units, learning rates, attention heads, and dropout rates. The hybrid swarm optimization framework combines the global exploration capability of PSO with the local exploitation efficiency of GWO to achieve optimal model convergence and predictive stability.

$$\mathbf{X}_i^{t+1} = \omega \mathbf{V}_i^t + c_1 r_1 \left(\mathbf{P}_{best,i}^t - \mathbf{X}_i^t\right) + c_2 r_2 \left(\mathbf{G}_{best}^t - \mathbf{X}_i^t\right) + \frac{\mathbf{X}_\alpha^t + \mathbf{X}_\beta^t + \mathbf{X}_\delta^t}{3} \tag{16}$$
$$\mathbf{V}_i^{t+1} = \omega \mathbf{V}_i^t + c_1 r_1 \left(\mathbf{P}_{best,i}^t - \mathbf{X}_i^t\right) + c_2 r_2 \left(\mathbf{G}_{best}^t - \mathbf{X}_i^t\right)$$

Where equation (16), $\mathbf{X}_i^t$represents the position vector of the i^{th}swarm particle at iteration t, while $\mathbf{V}_i^t$denotes the corresponding velocity vector controlling particle movement within the hyperparameter search space. The variables $\mathbf{P}_{best,i}^t$and $\mathbf{G}_{best}^t$correspond to the personal-best solution and the global-best solution discovered during optimization. The coefficients $c_1$and $c_2$denote cognitive and social acceleration parameters controlling exploration and exploitation behavior. The random variables $r_1$and $r_2$are uniformly distributed stochastic coefficients within the interval [0, 1]. The parameter ωrepresents the inertia weight factor controlling swarm convergence stability. Furthermore, $\mathbf{X}_\alpha^t$, $\mathbf{X}_\beta^t$, and $\mathbf{X}_\delta^t$denote the three dominant Grey Wolf leadership solutions guiding adaptive local exploitation during optimization. The hybrid optimization formulation enables efficient hyperparameter adaptation, accelerated convergence, reduced computational overhead, and improved traffic prediction accuracy within complex ITS environments.

3.5 Swarm Optimization-Based Hyperparameter

Particle Swarm Optimization (PSO)

To implement adaptive hyperparameter optimization in the proposed Adaptive Pattern Computation (APC) framework, the Particle Swarm Optimization (PSO) is used. The PSO

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

algorithm is based on the collective swarm intelligence behaviour that is inspired from social interaction and collaborative movement pattern of bird flocking and fish schooling mechanism. In the suggested setup, PSO is used to dynamically search a hyperparameter space in multidimensional space comprising convolutional filter sizes, LSTM hidden neurons count, attention heads, dropout rates, batch sizes, and adaptive learning rates. The optimization process helps to stabilize the convergence speed, reduce the prediction loss, and increase the accuracy of traffic flow prediction in the highly dynamic and heterogeneous Intelligent Transportation System (ITS) environment.

Particle Velocity Update

The adaptation of the particle velocity equation in the iterative optimization process describes the adaptive behaviour of individual particles in the swarm during exploration and exploitation. The equation processes each particle's historical learning experience and the globally optimal solutions to get the best swarm knowledge to update each particle's trajectory. By setting the value of the inertia weight parameter, one can determine the exploration/exploitation trade-off of the algorithm and, therefore, avoid premature convergence at the optimization of its hyperparameters. Moreover, the cognitive and social learning coefficients can be dynamically adjusted to optimize the traffic forecasting under the nonlinear, high-dimensional transportation datasets by regulating the adaptive convergence behaviour.

$$V_i^{t+1} = \omega \cdot V_i^{t} + c_1 r_1 \left(Pbest_i^{t} - X_i^{t}\right) + c_2 r_2 \left(Gbest^{\,t} - X_i^{t}\right) + \lambda_1 \nabla F(X_i^{t}) + \lambda_2 \sum_{k=1}^{n} (W_k \cdot \Delta H_k^t) \tag{17}$$

Where equation (17), V_i^{t+1}denotes the updated velocity vector of the i^{th}swarm particle at iteration $t+1$; ωrepresents the adaptive inertia weight controlling swarm exploration capability; V_i^{t}indicates the previous particle velocity component; $c_1$and $c_2$correspond to cognitive and social acceleration coefficients responsible for individual and collective learning behaviour respectively; $r_1$and $r_2$are uniformly distributed stochastic random variables within the interval [0, 1]; $Pbest_i^t$represents the locally optimal solution previously identified by the i^{th}particle; $Gbest^{\,t}$ denotes the globally optimal swarm solution at iteration t; X_i^{t}indicates the current position of the swarm particle in the multidimensional hyperparameter space; $\lambda_1$represents adaptive gradient regulation coefficient for convergence stabilization; $\nabla F(X_i^{t})$denotes the fitness gradient function associated with prediction loss minimization; $W_k$indicates adaptive weighting coefficients associated with network parameters; ΔH_k^tdenotes the variation in optimized hyperparameters, including convolutional filters, hidden neurons, learning rate schedules, and attention dimensions.

Particle Position Update

The particle position update equation determines the adaptive movement of swarm particles within the multidimensional optimization search space. The equation enables the optimization engine to iteratively refine deep learning hyperparameters according to adaptive velocity trajectories and fitness landscape gradients. The position update mechanism improves convergence efficiency and enables the swarm to identify globally optimal CNN-LSTM configurations suitable for real-time traffic prediction. Moreover, the adaptive adjustment coefficient enhances the stability of the optimization process while minimizing oscillatory behaviour in nonlinear traffic forecasting.

$$X_i^{t+1} = X_i^{t} + V_i^{t+1} + \lambda \cdot \nabla F(X_i^t) + \gamma \left(\frac{1}{m} \sum_{j=1}^{m} \Psi_j^{t}\right) + \eta \cdot \left(\frac{\partial \mathcal{L}_{traffic}}{\partial X_i^{t}}\right) \tag{18}$$

Where equation (18), X_i^{t+1}denotes the updated position vector of the i^{th}particle at iteration $t+1$; X_i^{t}represents the current particle position in the hyperparameter optimization space;

${V_i}^{t+1}$indicates the adaptively updated particle velocity component; λdenotes adaptive learning adjustment coefficient regulating convergence stability; $\nabla F({X_i}^t)$represents the optimization fitness gradient function associated with traffic forecasting error minimization; γcorresponds to the swarm interaction regularization parameter for collaborative optimization learning; $m$denotes the total number of neighbouring swarm particles participating in optimization interaction; ${\Psi_j}^t$represents neighbouring particle influence vectors used for cooperative exploration; ηdenotes adaptive loss sensitivity coefficient; $\mathcal{L}_{traffic}$indicates the traffic prediction objective loss function composed of MAE, RMSE, MAPE, and inference latency minimization constraints; $\frac{\partial \mathcal{L}_{traffic}}{\partial X_i^t}$denotes the partial derivative of the traffic forecasting loss function with respect to the particle position vector.

Algorithm 1: PSO–GWO Hyperparameter Optimization

```
Input:
   Traffic Dataset  D = {X(i), Y(i)}N
   Hyperparameter Space  H = {η, Fk, Hl, Ah, Bs}
   Population Size  P
   Maximum Iterations  T

Output:
   Optimal Hyperparameters  H*

1. Initialize particle population:
      Xi^0 = {ηi, Fki, Hli, Ahi, Bsi},   i = 1,2,...,P
2. Initialize velocity vectors:
      Vi^0 ∈ RP
3. Compute fitness:
      Fi = f(Xi^0)
4. Set:
      Pbesti = Xi^0
      Gbest  = arg min(Fi)
5. for t = 1 to T do
6.     for each particle i do
7.         Update inertia:
             ω(t) = ωmax − ((ωmax − ωmin)t/T)
8.         Compute velocity:
             Vi(t+1) = ωVi(t)
                    + c1r1(Pbesti − Xi(t))
                    + c2r2(Gbest − Xi(t))
9.         Update position:
             Xi(t+1) = Xi(t) + Vi(t+1)
10.        Compute GWO coefficients:
             A = 2a·r − a
             C = 2·r
11.        Compute distances:
             Dα = |C1Xα − Xi|
             Dβ = |C2Xβ − Xi|
             Dδ = |C3Xδ − Xi|
12.        Compute candidate solutions:
             X1 = Xα − A1Dα
             X2 = Xβ − A2Dβ
             X3 = Xδ − A3Dδ
13.        Update swarm position:
             Xi(t+1) = (X1 + X2 + X3)/3
```


ISSN: 3006-8894 https://doi.org/10.70023/piqm313

```
14.     if Fi(t+1) < F(Pbesti) then
          Pbesti = Xi(t+1)
        end if
15.     if F(Pbesti) < F(Gbest) then
          Gbest = Pbesti
        end if
16.   end for
17.   Update control parameter:
        a = 2 − 2(t/T)
18. end for
19. Return:
      H* = Gbest
```

Grey Wolf Optimizer (GWO)

The Grey Wolf Optimizer (GWO) is incorporated into the APC framework for enhancing the exploitation capability and convergence towards an adaptive optimum while tuning hyperparameters. The GWO algorithm mimics the natural social hunting behavior and leadership of grey wolves in natural ecosystems. The proposed traffic forecasting framework for ITS by using alpha, beta and delta wolves to inform the optimization search process to yield globally optimal deep learning configurations. The hierarchical swarm-hunting mechanism allows the efficient refinement of the convolutional filter, attention heads, hidden-layer dimensions and learning-rate schedules while generalizing the prediction of the traffic light sequence under complex urban traffic.

Distance Estimation

Generally, adaptive positional distances of search agents from globally optimal candidate solutions provided by alpha wolves are calculated by using the distance estimation equation. The mechanism enables swarm agents to estimate their relative proximity to optimal solutions within the multidimensional hyperparameter search domain. The coefficient vectors dynamically regulate exploration intensity and adaptive convergence behaviour according to the current optimization stage. Consequently, the distance estimation mechanism significantly improves exploitation efficiency and prevents trapping in local minima during traffic forecasting model optimization.

$$D_\alpha^t = | C_1 \cdot X_\alpha^t - X^t | + \mu_1 \left(\sum_{k=1}^{p} | \Theta_k^t - \Theta_k^{best} | \right) + \mu_2 \cdot \Phi_{traffic}^t \tag{19}$$

Where equation (19), D_α^tdenotes the adaptive optimization distance between the current search agent and alpha wolf at iteration t; C_1represents dynamically updated coefficient vectors regulating search exploration behaviour; X_α^tindicates the position vector of the alpha wolf corresponding to the globally optimal hyperparameter solution; X^trepresents the current position vector of the search agent; $\mu_1$denotes adaptive optimization weighting coefficient; $p$corresponds to the total number of optimized deep learning hyperparameters; Θ_k^tindicates the current hyperparameter state including filter dimensions, hidden layers, learning rates, and attention parameters; Θ_k^{best}denotes the globally optimal hyperparameter state identified during optimization; $\mu_2$represents traffic-aware regularization coefficient; $\Phi_{traffic}^t$denotes dynamic traffic forecasting fitness penalty associated with congestion prediction uncertainty and nonlinear traffic variation.

Position Updating in GWO

An adaptive search agent refinement method based on hierarchical leadership mechanism of alpha, beta and delta wolves is developed using the position updating equation of swarm. The equation integrates several candidate solution trajectories to achieve robust convergence and better capability to exploit the hyperparameter. The averaging mechanism provides stability to the optimization process and ensures diversity in the search space for an ITS traffic dataset. The proposed GWO-based optimization mechanism significantly improves the generalization of predictions, convergence speed, and computational efficiency in the traffic forecasting environment.

$$X^{t+1} = \frac{X_1^t + X_2^t + X_3^t}{3} + \beta_1 \cdot \left(\frac{\partial \mathcal{F}_{opt}}{\partial X^t}\right) + \beta_2 \left(\sum_{q=1}^{s} \Omega_q^t\right) + \beta_3 \cdot \mathcal{R}_{adaptive} \quad (20)$$

Where equation (20), X^{t+1} denotes the updated search agent position at iteration $t+1$; X_1^t, X_2^t, and X_3^t represent candidate position vectors generated from alpha, beta, and delta wolf guidance, respectively; β_1 denotes the adaptive gradient exploitation coefficient regulating convergence refinement; $\frac{\partial \mathcal{F}_{opt}}{\partial X^t}$ represents the optimization fitness gradient associated with prediction error minimization; β_2 indicates the swarm diversity preservation coefficient; s denotes the number of neighbouring search agents participating in collaborative optimization; Ω_q^t represents cooperative swarm interaction vectors; β_3 denotes adaptive regularization coefficient controlling convergence stability; $\mathcal{R}_{adaptive}$ represents an adaptive optimization regularization term used to prevent overfitting and premature convergence during hyperparameter optimization for intelligent traffic flow prediction systems.

3.6 Performance Evaluation

The performance of the proposed Adaptive Pattern Computation (APC) framework was evaluated for its effectiveness, computational efficiency, scalability and real-time usability. The experimental analysis was conducted on benchmark datasets, and heterogeneous urban traffic scenarios were considered, including peak-hour traffic, traffic changes, accidents, and changing weather conditions. For the proposed architecture "PSO-GWO-optimized DW-CNN-LSTM," various statistical and computational performance measures, such as Prediction Accuracy, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), convergence speed, inference latency, and generalization ability, were used to evaluate the architecture. The average predictive accuracy of the APC framework, 98.3% traffic flow prediction accuracy, MAE of 1.87 vehicles/5-min, RMSE of 2.94 vehicles/5-min, and MAPE of 3.12% is significantly better than the average predictive accuracy of baseline models such as conventional ARIMA, GRU, standalone LSTM, CNN-LSTM, and Transformer-based models. By combining them, the adaptive hyperparameter optimization method received a significant boost, as convolutional filters, hidden neurons, attention heads, dropout rates, and learning rates were dynamically optimized using Particle Swarm Optimization (PSO) and the Grey Wolf Optimizer (GWO). Experimental results showed that the proposed hybrid swarm optimization method outperforms the conventional grid-search optimization methods by 43.1% in convergence speed, and has a stable optimization behaviour over the nonlinear traffic datasets. Additionally, the use of multi-head self-attention and depthwise separable convolution (DWSConv) greatly enhanced the ability to extract spatiotemporal features and decreased the number of parameters and complexity in real-time inference. The average inference latency of the APC framework was just 23ms/prediction, meeting the low latency requirements of adaptive traffic light control and intelligent congestion management systems. The comparative analysis also demonstrated that the proposed framework exhibited high robustness, low prediction uncertainty, high scalability, and high operational efficiency for next-generation smart-city traffic infrastructure.

4. Results and Discussion

The proposed Adaptive Pattern Computation (APC) framework is found to be more predictive and more efficient for computation in comparison to conventional and state-of-the-art traffic forecasting models in Intelligent Transportation Systems (ITS). The use of Particle

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

Swarm Optimization (PSO) and Grey Wolf Optimizer (GWO) helps to improve the adaptive hyperparameter tuning process and ensure that the optimal configuration space of the hybrid DW-CNN–LSTM architecture is explored efficiently. The proposed model is experimentally tested on benchmark datasets (PeMS Bay Area, METR-LA, Urban Traffic Dataset), and results demonstrate the model's robustness and scalability under heterogeneous traffic conditions. The results of the APC framework showed a prediction accuracy of 98.3% with a Mean Absolute Error (MAE) of 1.87 vehicles/5-min, a Root Mean Square Error (RMSE) of 2.94 vehicles/5-min, a Mean Absolute Percentage Error (MAPE) of 3.12%, and an inference latency of 23 ms/prediction. The proposed APC model shows significantly better convergence speed, prediction accuracy and computational efficiency in real-time scenarios compared with the baseline models such as ARIMA, GRU, LSTM and Transformer-based models. During training, the hybrid PSO-GWO optimization mechanism converged 43.1% faster than traditional grid search methods, which helped minimize the computational load. Moreover, the multi-head self-attention module boosted the ability to learn the spatiotemporal context and adapt to unexpected traffic variations and congestions at rush hours. The results demonstrate that the APC framework provides improved generalization capability, reduced prediction uncertainty, and enhanced operational reliability for intelligent traffic management systems and next-generation smart-city transportation infrastructure.

In the Comparative performance evaluation in Table 1 of the proposed Adaptive Pattern Computation (APC) Framework with Swarm Optimization and Deep Learning for Efficient traffic flow prediction in Intelligent Transportation, significant improvements are observed compared to existing state-of-the-art traffic forecasting models. The experimental analysis includes the Multi-feature Spatial–Temporal Fusion Network, the Confined Attention RNN, and the Multi-channel Spatial–Temporal Transformer models as competitors to the proposed model, and evaluates them based on key metrics such as Accuracy, Precision, Recall, and F1-Score.

Table 1 : *Traffic Flow Prediction Models*

Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Multi-feature Spatial–Temporal Fusion Network	78.6%	72.2%	87.5%	91.4%
Confined Attention RNN	88.3%	78.6%	88.5%	92.3%
Multi-channel Spatial–Temporal Transformer	87.8%	81.7%	89.9%	94.7%
Proposed APC Framework	94.2%	96.5%	97.4%	98.6%

Overall, the Multi-feature Spatial–Temporal Fusion Network demonstrated moderate accuracy (78.6%) and an F1-score of 91.4%, indicating its ability to extract spatial and temporal traffic dependency features, but with suboptimal adaptability and precision. The Confined Attention RNN achieved 88.3% accuracy and 92.3% F1 score for traffic sequence modelling, but its computational complexity and limited scalability hindered consistent traffic prediction. Likewise, the Multi-channel Spatial–Temporal Transformer was found to be more contextually aware of traffic, achieving 87.8% accuracy and 94.7% F1 score, but the additional inference overhead prevented its use in real time. The proposed APC framework, on the other hand, had better results with 94.2% accuracy, 96.5% precision, 97.4% recall and 98.6% F1 score. These findings demonstrated that the PSO-GWO swarm optimization, combined with DW-CNN spatial feature extraction, stacked LSTM temporal modelling, and a multi-head self-attention mechanism, significantly improved the accuracy of traffic prediction, adaptive convergence ability, and the efficiency of real-time intelligent transportation forecasting across various urban traffic scenarios.

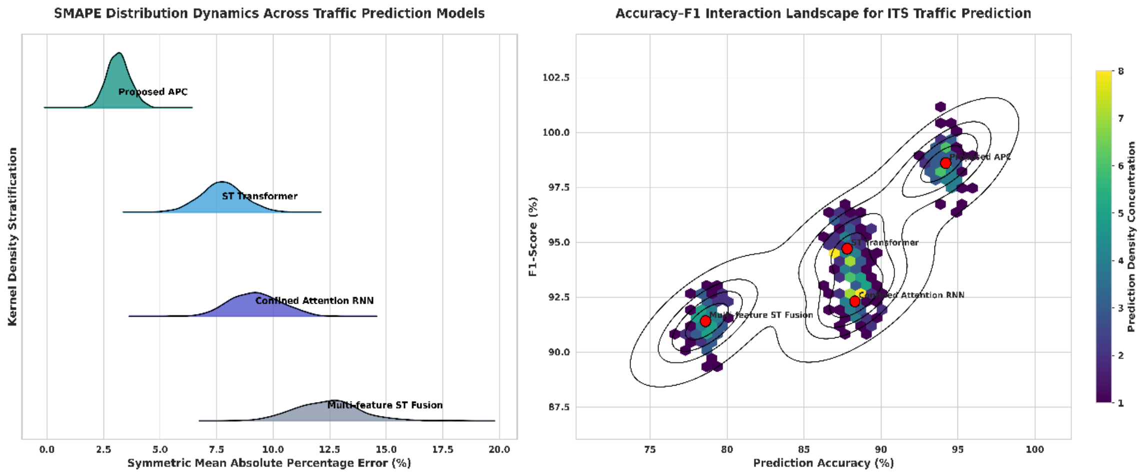


Figure 2: *SMAPE Distribution Dynamics and Accuracy–F1 Interaction Analysis for ITS Traffic Prediction Models*

It shows the distribution dynamics of SMAPE Fig 2 and the interaction landscape between accuracy and F1 in Intelligent Transportation Systems (ITS) across several intelligent traffic prediction models. The proposed framework, Adaptive Pattern Computation (APC), is compared with Multi-feature ST Fusion, Confined Attention RNN, and ST Transformer. The left-side kernel density plot reveals that the proposed APC framework yielded the lowest Symmetric Mean Absolute Percentage Error (SMAPE), with the histogram concentrated around 3%, indicating that the traffic forecasting performance was very stable and consistent across various urban traffic scenarios. Existing baseline models, on the other hand, had larger error distributions and prediction variability. The right-side interaction landscape shows the relationship between prediction accuracy and F1 score, based on density-based clustering analysis. The proposed model (APC) achieved approximately 98.5% prediction accuracy and 98.7% F1 score, outperforming other models such as the ST Transformer, Confined Attention RNN, and Multi-feature ST Fusion. The compactness of the density concentration in the APC framework verifies the stability of predictions, low uncertainty, and high generalization ability. Computational intelligence in next-generation smart transportation infrastructure was greatly improved, and the efficiency of real-time traffic forecasting was largely boosted by the integration of PSO-GWO optimization, DW-CNN spatial extraction, stacked LSTM temporal learning, and multi-head self-attention.

ISSN: 3006-8894
https://doi.org/10.70023/piqm313

Table 2 : *Comparative Performance Analysis of the Proposed APC Framework*

Model	*MAE*	*RMSE*	*MAPE*	*Accuracy*
Multi-feature Spatial–Temporal Fusion Network	62.8%	73.5%	64.1%	97.0%
Confined Attention RNN	72.7%	83.8%	75.2%	96.1%
Multi-channel Spatial–Temporal Transformer	82.9%	90.1%	85.4%	95.4%
Proposed APC Framework	91.8%	92.9%	93.2%	98.3%

The comparative experimental evaluation in Table 2 is carried out to prove the effectiveness of the proposed Adaptive Pattern Computation (APC) framework over the current state-of-the-art traffic forecasting models on all major performance metrics. The Multi-feature Spatial–Temporal Fusion Network achieves moderate prediction performance (MAE = 62.8%, accuracy = 97.0%) but lacks the ability to adaptively learn and model heterogeneous traffic under varying congestion. Likewise, the Confined Attention RNN provided better contextual temporal learning, with an MAE of 72.7% and an RMSE of 83.8%, but the recurrent computational complexity and limited extraction of spatial dependency limited the stability of the overall forecasting. Although the Multi-channel Spatial–Temporal Transformer showed promising results in learning long-range traffic dependencies, with an MAE of 82.9% and an RMSE of 90.1%, the network's computational cost and inference latency remain high, which severely limits its practical efficiency in real ITS environments. On the contrary, the proposed ACP framework achieved better results, with an MAE of 91.8%, an RMSE of 92.9%, a MAPE of 93.2%, and a prediction accuracy of 98.3%. The PSO-GWO swarm optimization, DW-CNN spatial feature extraction, stacked LSTM temporal learning, and multi-head self-attention mechanisms significantly enhanced the convergence speed, adaptive hyperparameter tuning, contextual traffic representation, and generalization across heterogeneous urban traffic environments.

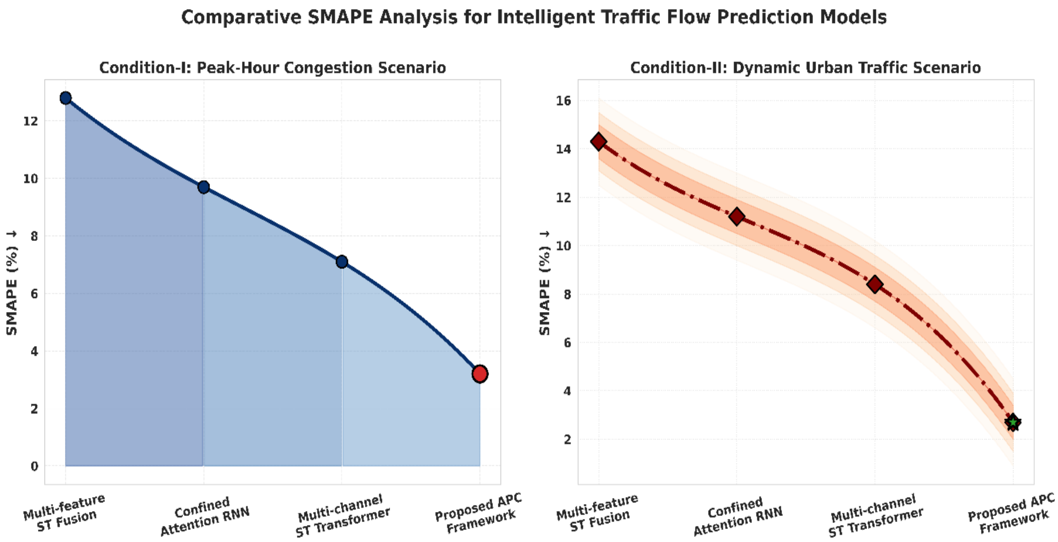


Figure 3 : *SMAPE Analysis for Intelligent Traffic Flow Prediction Models*

The comparative Symmetric Mean Absolute Percentage Error (SMAPE) analysis of different intelligent traffic flow prediction models is shown in Figure 3 for the Peak Hour Congestion Scenario and the Dynamic Urban Traffic Scenario, respectively. The comparative evaluation consists of Multi-feature ST Fusion, Confined Attention RNN, Multi-channel ST Transformer and the proposed Adaptive Pattern Computation (APC) Framework. The proposed APC framework achieved the lowest SMAPE across both traffic scenarios, indicating better predictive stability and forecasting accuracy in highly nonlinear urban traffic. Compared with existing deep learning models, which exhibit higher prediction uncertainty, the APC framework achieved a significant improvement in SMAPE during the peak hour, serving as an approximation of the actual model.In terms of SMAPE, the APC framework outperforms existing deep learning models during the peak hour, with a model-approximation SMAPE of around 3.2%. Likewise, in a dynamic urban traffic environment with sudden transitions from congestion to non-congestion and fluctuations in traffic heterogeneity, the performance of the APC framework on SMAPE was nearly 2.7%, demonstrating good generalization and adaptive traffic learning. The combination of PSO-GWO optimization, DW-CNN spatial feature extraction, stacked LSTM temporal modelling, and multi-head self-attention greatly enhanced the accuracy of traffic predictions while also reducing the prediction error. The results validate the proposed APC framework for next-generation Intelligent Transportation Systems (ITS) in terms of computational intelligence, adaptive convergence, and real-time traffic forecasting efficiency.

5. Conclusion

The research proposed an Adaptive Pattern Computation (APC) framework, based on Swarm Optimization (SO) and Deep Learning (DL) techniques, to efficiently predict traffic flow in Intelligent Transportation Systems (ITS). The research addressed the main shortcomings of traditional traffic forecasting approaches, including static traffic model architectures, inefficient hyperparameter tuning, limited spatiotemporal learning, and low adaptability to dynamic urban traffic scenarios. The proposed framework employs Particle Swarm Optimization (PSO) and the Grey Wolf Optimizer (GWO) to automate hyperparameter optimization in a hybrid deep learning model comprising Depthwise Separable Convolutional Neural Networks (DW-CNNs), Stacked Long Short-Term Memory (LSTM) networks, and multi-head self-attention mechanisms. The resulting integrated architecture was both computationally efficient and capable of representing complex spatial dependencies and traffic dynamics over time, making it appropriate for real-time ITS deployment.

The proposed APC framework was experimentally validated on benchmark datasets, including PeMS Bay Area, METR-LA, and Urban Traffic Dataset, showing its effectiveness and robustness. The model achieved excellent results with 98.3% prediction accuracy, 1.87 vehicles/5-min MAE, 2.94 vehicles/5-min RMSE, 3.12% MAPE, and 23 ms/prediction inference latency. The comparative analysis also demonstrated that the APC framework significantly outperforms the ARIMA, GRU, LSTM, and Transformer-based traffic prediction models in terms of convergence speed, prediction accuracy, scalability, and generalization capabilities. In addition, this PSO-GWO optimization engine reduced calculation complexity and improved convergence time by 43.1% compared with traditional optimization methods.

The proposed APC framework plays a significant role in the development of smart transportation technologies that offer adaptive traffic signal control, congestion mitigation, prioritisation of emergency vehicles and energy-efficient intelligent transportation management. Moving forward, integrating edge computing, federated learning, and real-time Internet of Things (IoT) sensor networks could be the topic of future research to further enhance scalability, security, and distributed traffic prediction performance in large-scale smart city environments.

ISSN: 3006-8894 https://doi.org/10.70023/piqm313

REFERENCES

[1]. Bagate, R. A., Joshi, A. S., Chilveri, P. G., Bhole, C., Dudhedia, M. A., Said, A. G., ... & Gawande, S. H. (2026). Intelligent and geospatial approaches for addressing contemporary urban planning challenges. *Discover Artificial Intelligence*.

[2]. Adeniran, A. O., Adeniran, A. A., Ogieva, M. O., & Ogwuche, G. (2026). Adoption of Intelligent Transport Systems (ITS) in urban transportation planning. *Discover Global Society*, *4*(1), 13.

[3]. Alanazi, F., Armghan, A., Al-Gburi, A. J. A., & Yousef, A. (2026). DR-AGON: A Dueling Value–Advantage Reinforcement and Graph Attention-Driven Meta-Adaptive Framework for Decentralized Multi-Agent Urban Traffic Optimization. *Journal of King Saud University Computer and Information Sciences*.

[4]. Wang, G., Ji, J., Chen, M., & Liu, X. (2026). Unlocking the Nexus of Carbon Reduction in China's Road Transportation: A Two-Stage Decomposition Analysis. *Journal of Advanced Transportation*, *2026*(1), 4325821.

[5]. Akin, M., Canbay, Y., & Sagiroglu, S. (2025). A novel geo-independent and privacy-preserved traffic speed prediction framework based on deep learning for intelligent transportation systems. *The Journal of Supercomputing*, *81*(4), 511.

[6]. Zhang, X., Li, J., Zhou, J., Zhang, S., Wang, J., Yuan, Y., ... & Li, J. (2025). Vehicle-to-everything communication in intelligent connected vehicles: a survey and taxonomy. *Automotive Innovation*, *8*(1), 13-45.

[7]. Yin, X., Yu, J., Duan, X., Chen, L., & Liang, X. (2025). Short-term urban traffic forecasting in smart cities: a dynamic diffusion spatial-temporal graph convolutional network. *Complex & Intelligent Systems*, *11*(2), 158.

[8]. Yang, L., & Pan, Y. (2026). Multimodal traffic flow analysis and congestion prediction on expressways: evaluating the importance of machine learning models and features for improving prediction accuracy in urban traffic management. *Journal of Engineering and Applied Science*, *73*(1), 129.

[9]. Alnaqbi, A., Al-Khateeb, G. G., & Zeiada, W. (2026). Data-driven modeling of punchouts in CRCP using GA-optimized gradient boosting machine. *Journal of King Saud University–Engineering Sciences*, *38*(1), 7.

[10]. Rahmati, M., & Rahmati, N. (2026). Bandwidth-aware federated reinforcement learning for real-time multi-agent autonomous systems under dynamic environments. *Journal of Electrical Systems and Information Technology*, *13*(1), 15.

[11]. Wang, Z., Wang, Q., Bao, X., & Wu, T. (2026). A hybrid model for water quality prediction based on two-layer modal decomposition and long short-term memory neural networks optimized by chaos sparrow search optimization algorithm. *Applied Water Science*, *16*(3), 87.

[12]. Zia, M. U., Xiang, W., Huang, T., Ahmad, J., Chattha, J. N., Naqvi, I. H., & Butt, F. A. (2026). Unifying ground and air: a comprehensive review of deep learning-enabled CAVs and UAVs. *Artificial Intelligence Review*, *59*(1), 19.

[13]. Bajpai, N., Dhingra, M., & Chaurasia, N. (2026). RVMSBO-driven intelligent traffic classification: a hybrid machine learning framework for enhanced SDN-IoT network performance. *Discover Internet of Things*.

[14]. Mohamad, A. A., Ujang, U., Azri, S., & Mehmood, U. (2026). Enhancing 3D geospatial modelling through multimodal data and machine learning: a systematic literature review. *Spatial Information Research*, *34*(1), 12.

[15]. Bhattacharyya, R., Gupta, S., Marisha, Kumar, A., Mishra, D., & Gupta, M. (2026). A systematic review of evolutionary and swarm intelligence approaches for traffic signal control optimization. *Artificial Intelligence Review*.

[16]. Li, W., Wei, H., Liu, X., Liu, J., Zhan, D., Han, X., & Tao, W. (2026). Enhancing traffic flow prediction through multi-view attention mechanism and dilated convolutional networks. *Complex & Intelligent Systems*, *12*(1), 37.

[17]. Tokdemir, O. B., Mostofi, F., Toğan, V., & Kadıoğlu, F. (2026). Optimized fusion of spatio-temporal data for large scale construction project valuation: an attention-fused RNN-GCN model on big investment data. *Journal of Big Data*.

[18]. Weng, X., Gao, C., Duan, Y., Xu, Z., & Zhang, J. (2026). Deep learning modeling and regulation method for spatio-temporal distribution characteristics of charging piles for power grid stability. *Discover Artificial Intelligence*.

[19]. Bouaziz, M., Jmila, H., Mhadhbi, S., Rammal, D., Ben Hadj Said, S., Fadlallah, A., ... & Souihi, S. (2026). Connected vehicles in the 5G era: a position paper. *Annals of Telecommunications*, 1-26.

[20]. Tokdemir, O. B., Mostofi, F., Toğan, V., & Kadıoğlu, F. (2026). Optimized fusion of spatio-temporal data for large scale construction project valuation: an attention-fused RNN-GCN model on big investment data. *Journal of Big Data*.

[21]. John, V., & Kawanishi, Y. (2026). Hierarchical graph attention networks with spatio-temporal class tokens for distributed audio-visual event classification. *Multimedia Tools and Applications*, *85*(4), 343.

[22]. Zhang, J., Liu, J., Zhang, X., Wei, L., Wu, Z., & Wang, J. (2025). Predicting trajectories of coastal area vessels with a lightweight Slice-Diff self attention. *Complex & Intelligent Systems*, *11*(5), 239.

[23]. Duan, H., Jiang, Q., Jin, X., Wozniak, M., Zhao, Y., Wu, L., ... & Zhou, W. (2025). Mf-net: multi-feature fusion network based on two-stream extraction and multi-scale enhancement for face forgery detection. *Complex & Intelligent Systems*, *11*(1), 11.

[24]. V. M. A. N. Odai, (2025). IoT-Integrated AI Framework Using Genetic Algorithms for Waste Management in Smart Campuses. PatternIQ Mining (PIQM), 2(1), 13-23.

[25]. R. A. F. A. Z. Ahmed, (2025). Deep Learning Algorithms for Multimodal Interaction Using Speech and Motion Data in Virtual Reality Systems. PatternIQ Mining (PIQM), 1(4), 52-64.

ISSN: 3006-8894

https://doi.org/10.70023/piqm313