vehicles

MDPI

Article

An Adaptive Spatiotemporal Graph Convolutional Method for Highway Traffic Flow Prediction Based on Multi-Period Modalities

Guozheng Li [1,2], Baijing Wu [1], Ke Gao [1] and Guanghui Yan [1,*]

[1] School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China; 13210066@stu.lzjtu.edu.cn (G.L.); 13240094@stu.lzjtu.edu.cn (B.W.); 13250125@stu.lzjtu.edu.cn (K.G.)
[2] Information Engineering College, Gansu University of Traditional Chinese Medicine, Lanzhou 730013, China
* Correspondence: ghyan@mail.lzjtu.cn; Tel.: +86-17393160728

Abstract

To address the limited prediction accuracy caused by neglecting the inherent periodicity of spatiotemporal traffic flows during spatial feature extraction, this study develops an adaptive spatiotemporal graph convolutional method for highway traffic flow prediction. Firstly, an adaptive temporal graph generation layer with multiple time periods is constructed to dynamically generate traffic flow temporal graphs with rich representations, enabling accurate characterization of spatiotemporal traffic patterns. Secondly, a lightweight Transformer architecture is introduced to design an efficient feature extraction module, which refines both global and local spatiotemporal variations as well as their interactions. Finally, a multi-head self-attention module integrating different temporal scales is designed to capture the intrinsic correlations and dynamic dependencies across multi-scale traffic data, thereby enhancing prediction accuracy and generalization capability. Extensive experiments on two publicly available datasets, PEMSBAY and PEMSM, demonstrate the effectiveness of the proposed method. Compared with the baseline approaches, the proposed model achieves average reductions of 14% in MAE, 19% in MAPE, and 15% in RMSE. These results indicate that the proposed framework improves forecasting accuracy and provides a reliable methodological foundation for intelligent transportation systems.

Keywords: traffic flow prediction; spatiotemporal graph convolution; time series graph; adaptive; attention

1. Introduction

With the rapid development of smart cities and the continuous growth of transportation demand, traffic flow prediction plays a crucial role in intelligent transportation systems. Accurate traffic flow forecasting provides traffic authorities with scientifically grounded support for management and decision-making, helping to alleviate traffic congestion, improve road utilization efficiency, and ensure the normal mobility of residents [1]. However, traffic flow data exhibit complex spatiotemporal characteristics. Their variations are influenced by multiple factors, including road network structure, traffic signal control, temporal patterns, and weather conditions, resulting in highly dynamic and uncertain behavior. Moreover, traffic flow data are inherently high-dimensional and multi-variable in nature [2]. Therefore, effectively extracting spatiotemporal features from traffic data, modeling intricate temporal and spatial dependencies, and designing an end-to-end high-precision

Check for updates

Received: 4 March 2026
Revised: 10 April 2026
Accepted: 29 April 2026
Published: 31 May 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

forecasting framework remain important research challenges with significant theoretical value and practical implications.

Traffic flow prediction refers to the process of analyzing historical traffic flow data collected over a certain period to estimate and infer traffic volumes in future time intervals. In this context, deep learning-based traffic flow prediction methods have gradually become the mainstream approaches in this field due to their strong capability to handle large-scale data and automatically extract features, as well as their effectiveness in capturing complex temporal patterns in time series data. Among them, recurrent neural networks (RNNs) [3], long short-term memory networks (LSTMs) [4], gated recurrent units (GRUs), and their variants [5] learn complex and dynamic traffic flow characteristics by accumulating information from different historical time steps and have been widely applied to traffic flow prediction tasks. To improve the accuracy of traffic flow forecasting, Ma et al. [6] proposed the Spatio-Temporal Fusion Graph Convolutional Network (STFGCN), which captures spatial dependencies through node-specific graph convolution and an adaptive adjacency matrix. In addition, a continuous-time learning module and a Transformer-based global module were employed to model local and global temporal dependencies, respectively. Feng et al. [7] introduced the Low–High-Frequency Network (LHFNet), which decomposes traffic signals into different frequency components. Specifically, a low-frequency encoder is designed to extract periodic patterns, while a high-frequency encoder captures short-term fluctuations. A gating mechanism is further adopted to fuse the frequency-aware representations for improved forecasting performance. Liang et al. [8] proposed Dual Spatial and Memory-Enhanced Single Spatial Feature Extraction and Static Architectures Framework (DSM-STWave), which integrates a dual-spatial mechanism combining global sparse attention with local k-nearest neighbors to efficiently capture both long- and short-range spatio-temporal dependencies. In addition, an adaptive module was introduced to satisfy the accuracy requirements of online prediction. Experiments conducted on four real-world datasets demonstrated the method's effectiveness in terms of both real-time performance and predictive accuracy. However, these methods remain prone to gradient vanishing or explosion during back propagation, making it difficult to effectively preserve historical state information beyond 10 steps. As a result, when confronted with dynamic variations at different temporal scales, such as morning and evening peak hours, holidays, and unexpected congestion, the model cannot accurately extract the temporal patterns most relevant to the current prediction.

To better integrate the dual temporal and spatial characteristics of traffic flow data, graph convolutional networks (GCNs) [9] have been introduced to model spatiotemporal relationships within traffic networks. In this framework, traffic sensors or road segments are abstracted as nodes in a graph, and their interactions are represented through edges. By constructing a predefined Laplacian matrix based on node distances or adjacency relationships, spatial dependencies among nodes are explicitly modeled, enabling effective capture of the spatial correlations inherent in traffic flow dynamics. To address the limitations of existing graph convolutional approaches, such as insufficient shared pattern learning, reliance on static relational assumptions, and limited capability in capturing complex traffic flow dynamics, Zhang et al. [10] proposed a dynamic graph convolution and temporal representation learning framework, termed DGCN-TRL (Dynamic Graph Convolutional Networks with Temporal Representation Learning). The framework employs temporal graph convolution blocks to capture flexible temporal dependencies, while a dynamic graph constructor is introduced to explore evolving spatiotemporal relationships. In addition, a masked sequence Transformer is adopted in a pre-training manner to learn contextual temporal representations. Zhao et al. [11] introduced a temporal graph convolutional network (T-GCN) for traffic prediction, which effectively captures both spatial and temporal dependencies in

traffic flow data by incorporating gated mechanisms inspired by GRU structures, achieving high-accuracy traffic forecasting in urban road networks. However, conventional graph convolutional networks rely on fixed Laplacian matrices, which limits their ability to capture the dynamic evolution of spatiotemporal dependencies in traffic flow data. To address this limitation, dynamic spatiotemporal graph convolutional networks have been developed to adaptively model evolving spatial relationships. By dynamically updating or analyzing the Laplacian matrix, these approaches integrate the spatial modeling capability of graph convolution with the temporal forecasting capacity of dynamic sequence modeling. This framework enables a more comprehensive exploration of spatiotemporal characteristics in traffic flow data and facilitates the modeling of complex and evolving dependency structures, thereby improving forecasting accuracy. Cao et al. [12] proposed a Spatiotemporal Multiscale Graph Convolutional Network (SMGCN) to capture multi-scale spatiotemporal correlations in traffic flow data. The model first generates coarse-grained graphs through graph clustering and extracts spatiotemporal dependencies at multiple scales. A cross-scale fusion mechanism is then employed to facilitate information interaction between coarse- and fine-grained graphs. In addition, an adaptive dynamic graph convolution module is introduced to jointly explore static and dynamic spatial features. Yang et al. [13] addressed the limitation of classical GCNs that only consider adjacent road connections by proposing a multi-graph learning-based model, TPP-GCN, for predicting traffic propagation flows in urban road networks. Liu et al. [14] introduced a spatiotemporal graph convolutional framework for fine-grained traffic state prediction, explicitly considering the physical evolution mechanisms underlying traffic dynamics. By integrating GRU-based temporal modeling with a Transformer encoder architecture, the model effectively captures short-term fluctuations and long-range temporal dependencies. The reported results indicate improved reliability in long-horizon forecasting tasks, contributing to more informed traffic management and congestion control strategies.

In summary, graph convolutional network-based methods have achieved notable progress in traffic flow prediction. However, most existing approaches predominantly focus on either spatial correlations or regional traffic flow dynamics, while failing to jointly account for the combined effects of these two aspects on traffic flow evolution. In addition, external influencing factors, such as weather conditions, have not been sufficiently incorporated into current modeling frameworks [15]. From an algorithmic perspective, conventional GCN models mainly consider interactions among predefined neighboring nodes during temporal information aggregation, while overlooking long-range node dependencies within the graph structure. As a result, feature representations of distant nodes are inadequately captured, which ultimately degrades prediction performance [16]. Therefore, when applying graph convolutional networks to traffic flow prediction, it is essential to comprehensively consider these factors and adaptively design and optimize models according to specific scenarios and requirements in order to achieve higher prediction accuracy.

To address the limitation that existing methods fail to incorporate the periodic characteristics of spatiotemporal data during spatial feature extraction, which leads to suboptimal prediction accuracy, this study develops an adaptive spatiotemporal graph convolutional-based highway traffic flow prediction method, termed ASM-STGCN, with ASTGCN as the baseline. The proposed method integrates adaptive temporal diagram generation, lightweight Transformer-based feature refinement, and multi-head self-attention mechanisms within the STGCN framework to enhance periodic dependency modeling and prediction performance. The main contributions of this study are summarized as follows:

1. An adaptive spatiotemporal graph convolutional framework for highway traffic flow forecasting, termed ASM-STGCN, is developed to explicitly incorporate the periodic characteristics of spatiotemporal traffic data into spatial feature learning.

By modeling traffic periodicity during feature extraction, the proposed framework mitigates performance degradation observed in existing methods that overlook such temporal regularities;

2. A multi-period adaptive temporal graph generation layer is designed to model periodic spatiotemporal dependencies. A multi-period Adaptive Timing Diagram (ATD) generation layer is introduced to dynamically construct adaptive temporal graphs at different time scales. By explicitly modeling spatiotemporal node correlations under multiple temporal periods, the proposed layer enhances the representation and extraction of periodic temporal dependencies in traffic flow data;
3. A lightweight Transformer-based feature refinement module is proposed to enhance deep feature propagation. A Simplifying Transformer (SIRT) module is developed and integrated after graph convolution to efficiently refine subtle variation patterns within graph-convolved features. Through a lightweight self-attention mechanism, SIRT improves feature interaction and propagation in deeper network layers while maintaining computational efficiency.
4. A multi-scale multi-head self-attention (MHA) fusion module is designed to integrate feature representations across different temporal resolutions. By capturing intrinsic correlations and dynamic variations among multi-scale patterns, the module enhances prediction accuracy and generalization capability.

2. ASM-STGCN Traffic Flow Prediction Method

2.1. Mathematical Formulation of Traffic Flow Prediction

The objective of highway traffic flow forecasting is to predict future traffic conditions based on historical observations using dynamic spatiotemporal graph convolutional modeling. A fundamental step in this framework is the formal definition and representation of the traffic network as a graph structure. In this representation, nodes correspond to traffic sensors or road segments, each associated with a unique identifier and attributes such as geographical location and traffic flow measurements. Edges describe the relationships between nodes, which may represent physical connectivity (e.g., adjacent road segments) or logical associations (e.g., traffic flow correlations). The weights of these edges are typically determined based on factors such as geographical proximity or statistical similarity in traffic patterns. Based on this graph representation, graph convolution operations are employed to capture spatial relationships among nodes in the traffic network. By leveraging the adjacency matrix and node feature matrix, spatial representations of nodes are computed to characterize the mutual influences and dependency structures within the network. Meanwhile, temporal dependencies in traffic flow data are modeled using convolutional neural networks. Specifically, sliding windows are applied to generate sequential inputs for temporal convolution modules, enabling the extraction of both local temporal patterns and global evolution trends in traffic flow time series [17].

In this study, the traffic flow observations collected by all nodes in traffic network G at time step t are defined as:

$$X_t = \{x_1^t, x_2^t, \ldots, x_N^t\} \in \mathbb{R}^N \tag{1}$$

where x_i^t denotes the traffic flow observations at time step t, and N represents the number of nodes in the traffic network. The historical traffic flow data over L consecutive time intervals are denoted as $X = \{X_1, X_2, \ldots, X_L\}$, while the predicted traffic flow outputs over the future l time intervals are represented as $Y = \{X_{l+1}, X_{l+2}, \ldots, X_{l+l}\}$. Based on these definitions, the overall process of highway traffic flow prediction using spatiotemporal graph convolution can be formulated as follows:

$$\left\{X_G^{t-L+1}, X_G^{t-L+2}, \ldots, X_G^t\right\} = f\left(\left\{X_G^{t+1}, X_G^{t+2}, \ldots, X_G^{t+L}\right\}\right) \tag{2}$$

where f denotes the modeling and computational process of the spatiotemporal graph convolutional neural network. The overall modeling strategy is to utilize historical traffic flow data over the past L time intervals to predict traffic flow values over the subsequent l future time intervals.

2.2. ASM-STGCN

The overall architecture of the proposed ASM-STGCN framework is illustrated in Figure 1. The model first takes traffic flow time-series data as input and defines the corresponding graph nodes representing sensors or road segments. The ATD then dynamically constructs temporal graphs at different time scales based on intrinsic correlations among traffic nodes. The generated temporal graphs are subsequently fed into spatiotemporal blocks (ST blocks) with residual connections to extract spatial and temporal features. Through residual learning, the ST blocks facilitate the modeling of increasingly complex spatiotemporal dependencies. Each ST block consists of a spatial attention layer (Satt), a temporal attention layer (Tatt), a graph convolution layer (GCN), and a convolutional layer. By jointly modeling spatial and temporal interactions at multiple scales, the ST block provides comprehensive spatiotemporal feature representations. The extracted features are further refined by the SIRT module, which integrates global spatial patterns, local temporal variations, and their interactions. Through attention matrix initialization and regulation, SIRT promotes stable feature propagation in deeper layers and mitigates potential rank collapse during training. After multi-scale feature extraction from different temporal resolutions, the MHA module performs adaptive fusion across scales. This multi-head self-attention mechanism enables the model to learn intrinsic correlations and dynamic interactions among multi-scale representations, thereby enhancing generalization capability. Finally, the network outputs the predicted traffic flow values after end-to-end training.

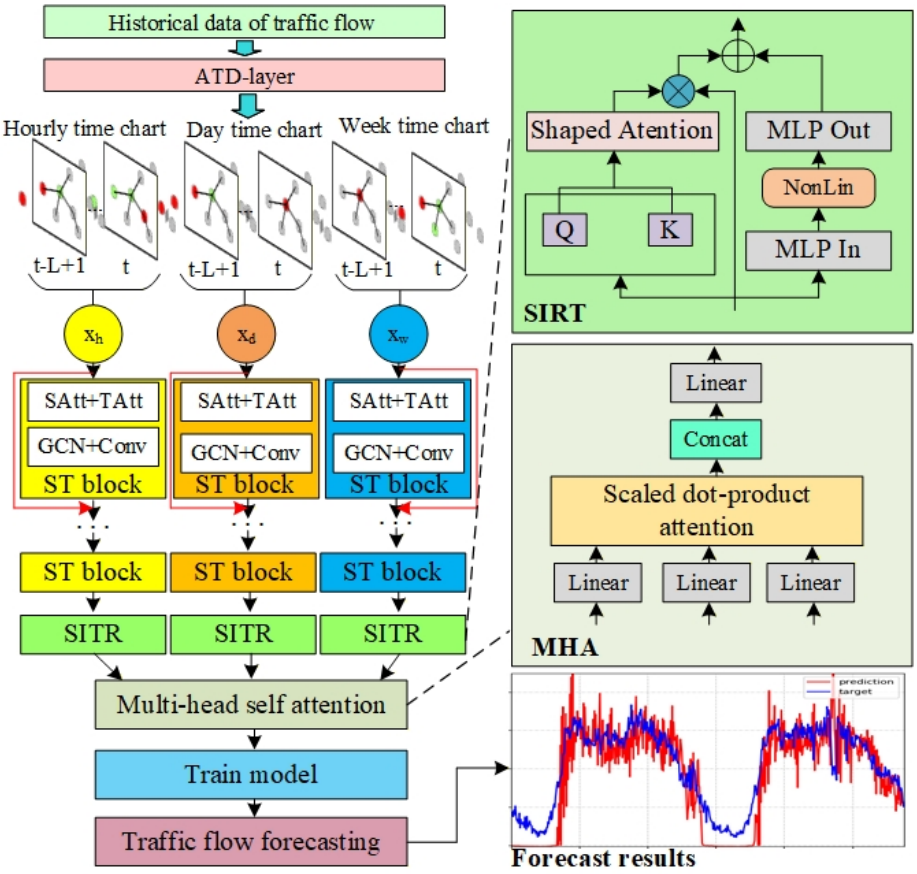

Figure 1. Overall framework of the ASM-STGCN model.

2.3. ATD Layer

Traffic networks constitute large-scale and highly complex systems, in which interactions among internal nodes—such as intersections, critical road segments, and transportation hubs—are intricate and heterogeneous. During traffic flow prediction, conventional graph structures often struggle to comprehensively capture these complex dependencies. To address this issue, graph learning layers have been introduced to adaptively learn adjacency matrices, enabling the discovery of latent relationships embedded in traffic time-series data and enhancing feature representation capability. In existing traffic temporal graph construction methods, node relationships and similarities are commonly established using distance-based metrics, such as Euclidean distance or dot-product similarity between nodes [18]. However, these approaches typically incur high computational complexity and fail to jointly consider spatial structures and temporal periodicity during feature extraction. As a result, spatiotemporal characteristics are insufficiently represented, which limits model efficiency and prediction accuracy [19].

To overcome the limitations of graph generation in traffic flow prediction scenarios, this study proposes a multi-period adaptive temporal graph generation layer (ATD). By explicitly modeling spatiotemporal correlations among traffic nodes, the ATD layer constructs adaptive temporal graphs at different time scales based on hourly, daily, and weekly traffic data, thereby improving the model's ability to capture and extract periodic temporal dependencies. In constructing accurate traffic temporal graphs, an association matrix is employed to characterize node connectivity. This matrix not only preserves local neighborhood information within the traffic network but also incorporates global high-order correlation features [20]. Consequently, the resulting traffic temporal graphs exhibit clear structures and rich information content, enabling a more accurate and comprehensive representation of real-world traffic network dynamics.

As illustrated in Figure 2, in the traffic flow prediction network G, traffic entities at different time steps—including nodes, edges, and their interrelationships—are represented as $G = (j, e, z)$, where j denotes the set of nodes, e represents the set of edges, and z corresponds to the feature representations associated with node interactions. The ATD layer first applies the tanh activation function to the traffic flow data to compute the hidden relationship matrix between adjacent nodes. Next, the hidden relationship matrices of neighboring nodes are subtracted to obtain a differential matrix. Finally, ReLU and tanh are employed to calculate the cross-correlation matrix between two adjacent nodes, thereby constructing a temporal graph among neighboring traffic flow data nodes. The ATD layer can adaptively learn the graph adjacency matrix to capture the latent relationships among time-series data. In multivariate time-series forecasting, changes in the condition of one node may lead to variations in another node. Accordingly, the temporal graph is constructed as follows:

$$M_1 = \tanh(\alpha \cdot j_1 \cdot \Theta_1) \tag{3}$$

$$M_2 = \tanh(\alpha \cdot j_2 \cdot \Theta_2) \tag{4}$$

$$\Omega = \mathrm{ReLU}\big(\tanh(\alpha \cdot (M_1 M_2^T - M_2 M_1^T))\big) \tag{5}$$

where α is a trainable parameter used to control the scale of the nonlinear transformation, j denotes the nodes, $j \in \mathbb{R}^{N \times d}$. N is the number of nodes, d represents the feature dimension, Θ denotes the cross-correlation matrix between two adjacent nodes, $\Theta \in \mathbb{R}^{d \times d}$. $M \in \mathbb{R}^{N \times d}$ is the hidden relationship matrix corresponding to the nodes. ReLU and tanh denote two different activation functions, respectively. Specifically, tanh is used to preserve the nonlinear characteristics in the graph structure modeling process, whereas ReLU ensures the non-negativity of adjacent nodes. Ω is the adaptive adjacency matrix, whose values are non-negative and characterize the directed or symmetric dependencies among nodes.

The ATD layer dynamically adjusts the weights between connected nodes according to the temporal input, thereby enabling the model to learn stable and interpretable node relationships over the periodic patterns of the training dataset.

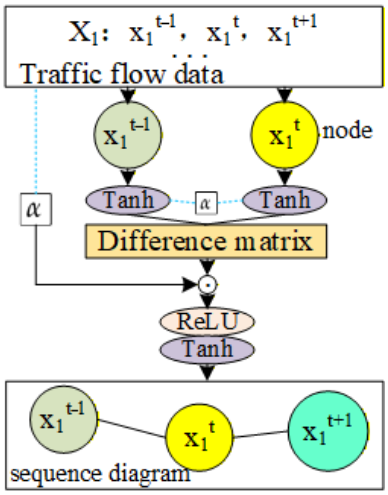

Figure 2. Adaptive temporal graph generation layer (ATD).

2.4. SIRT Module

Transformers have demonstrated strong capability in extracting spatiotemporal features from traffic flow time-series data by modeling long-range dependencies, enabling effective capture of interactions among traffic flows across different time intervals and spatial locations. Through the use of multi-head attention mechanisms and stacked layers, Transformer architectures support dynamic and hierarchical feature learning, allowing the integration of global spatial patterns, local temporal variations, and their interactions. However, the self-attention employed in standard Transformer models incurs quadratic computational complexity with respect to sequence length, which significantly limits computational efficiency and degrades both training and inference performance in large-scale traffic flow prediction tasks. Motivated by these limitations, this study introduces a lightweight Transformer-based design to achieve more efficient feature extraction and computation while preserving the capability to model essential spatiotemporal dependencies.

As illustrated in Figure 3, the SIRT architecture removes the original skip-connection structure of the standard Transformer block [21] and adopts a Value-SkipInit strategy to compute the self-attention matrix. This design effectively alleviates signal degradation during feature propagation and prevents rank collapse of feature representations during model training. SIRT first divides the input features into three branches. In the first branch, matrices Q and K are generated and shaped attention is employed for efficient and lightweight modeling of the long-range dependencies of traffic flow features. The resulting representation is then fused with the second branch through element-wise multiplication with a skip-connected feature map. Finally, the third branch passes through an MLP to extract global dependency features at different time steps, which are then added to the fused representations of the first and second branches. In this way, SIRT achieves efficient and stable long-term traffic flow prediction with extremely low parameter overhead. Given an input feature sequence $x \in \mathbb{R}^{E \times d}$ with E tokens and feature dimension d, the output feature representation is denoted as x_{out}. In a standard Transformer block, the entire computation process can be formulated as follows:

$$x_{\text{out}} = \gamma_{FF}\,\hat{X} + \beta_{FF} M_{\text{LP}}(N_{\text{orm}}(\hat{X})) \tag{6}$$

$$\hat{X} = \gamma_{SA} x + \beta_{SA} M_{\text{HA}}(N_{\text{orm}}(x)) \tag{7}$$

where $\hat{X}$ denotes the output feature generated by the multi-head attention mechanism in the Transformer block, γ_{SA}, γ_{FF}, β_{SA}, and β_{FF} are learnable scalar weights. N_{orm} denotes the normalization layer, M_{LP} denotes the multilayer perceptron, and M_{HA} denotes the multi-head attention mechanism in the Transformer block. The input to M_{HA} is formulated as follows:

$$M_{\text{HA}}(x) = C_{\text{oncat}}\left(\sum_{i=1}^{d} A(x)_i x_i W_i^V\right) W^P \tag{8}$$

$$A(x) = S_{\text{oftmax}}\left(\frac{1}{\sqrt{d}}\, x \cdot w^Q \cdot w^K \cdot w^{K^T} \cdot x^T + M\right) \tag{9}$$

Here, w^Q, w^K, and w^V are the learnable weights of the query Q, key K, and value V, respectively. C_{oncat} denotes the feature concatenation operation. $M \in \mathbb{R}^{E\times E}$ is a mask matrix, which is dynamically adjusted according to the prediction scheme to regulate the region processed by the activation function, with its values ranging from 0 to $-\infty$. In addition, $w^P \in \mathbb{R}^{d\times d}$ is the trainable projection matrix of the multi-head attention mechanism. S_{oftmax} denotes the Softmax activation function.

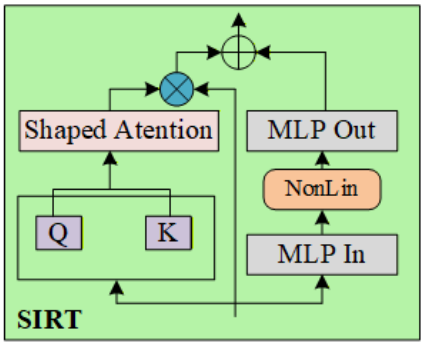

Figure 3. Architecture of the SIRT module.

The SIRT architecture removes the residual (skip-connection) structure of the original Transformer block and modifies the self-attention computation to obtain $A(x)$ as follows:

$$A(x) \leftarrow (\rho_1 \cdot F_{\text{self}} + \rho_2 \cdot A(x) - \rho_3 \cdot C) \tag{10}$$

where ρ_1, ρ_2, and ρ_3 are trainable parameters and C is a constant used to compute the center matrix. The values of ρ_1 and ρ_2 are initialized to 0 and 1, respectively. $F_{\text{self}} \in \mathbb{R}^{E\times E}$ denotes the self-attention matrix. The advantage of this module lies in initializing the self-attention matrix, which enables the model to capture more fine-grained variation trends among traffic flow input tokens, thereby achieving better feature propagation in deeper network layers.

2.5. MHA

In the original ASTGCN framework, traffic flow data at hourly, daily, and weekly scales are fused using a concatenation operation. However, such a fusion strategy may introduce feature redundancy across different temporal scales. Since Concat directly stacks traffic flow features from different time resolutions, the model may struggle to effectively capture intrinsic correlations and dynamic interactions among multi-scale temporal patterns, thereby limiting generalization performance. To better integrate features across different temporal scales and improve prediction accuracy, a multi-head self-attention (MHA) mechanism is introduced to enhance multi-scale feature fusion. This is illustrated in Figure 4. First, feature matrices at three different scales are taken as inputs and passed through linear layers, where linear transformations project the input vectors into query, key,

and value matrices. These representations capture information from the input traffic flow sequences across multiple temporal scales, thereby enhancing the model's representation capacity and parallel processing capability. Subsequently, scaled dot-product attention is applied to compute the correlation scores between different positions in the multi-scale input sequences, and the input sequences are weighted and aggregated according to these scores to generate new feature representations. The resulting features are then concatenated, after which a final linear layer is used to integrate the features and map the internal representations to the output of the traffic flow prediction task by adjusting the weights and biases accordingly [22]. The overall computation of the MHA module can be formulated as follows:

$$M_{\mathrm{HA}}(F_H, F_D, F_W) = S_{\mathrm{oftmax}}\left(\frac{F_H \cdot F_D^T}{\sqrt{d}}\right) \cdot F_W \tag{11}$$

here, F_H, F_D, and F_W denote the input feature maps corresponding to the hourly, daily, and weekly temporal scales, respectively. T represents the matrix transpose operation, and S_{oftmax} denotes the Softmax activation function.

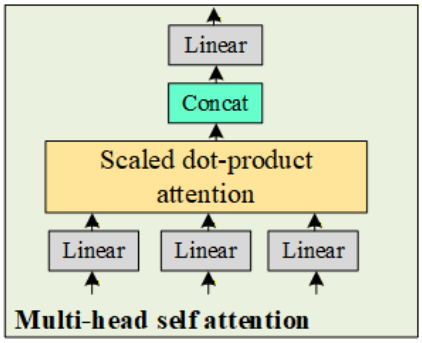

Figure 4. MHA Structure.

3. Experiments and Results Analysis

3.1. Dataset Description and Experimental Settings

To validate the effectiveness and robustness of the proposed method, experiments were conducted on two publicly available real-world traffic datasets: PEMSBAY [23] and PEMSM [24]. The PEMSBAY dataset contains traffic measurements collected from 1 January 2017 to 30 June 2017. It includes large-scale time-series data of traffic flow, speed, and occupancy recorded by 325 traffic sensors. The dataset provides sensor-related information such as sensor IDs and geographical locations. The traffic measurements are stored in time-series format, with each sensor reporting traffic flow, average speed, and occupancy at fixed time intervals. The PEMS-M dataset covers traffic data collected in District 7 of California from 1 May 2012 to 30 June 2012. The dataset includes traffic speed data recorded by 228 sensors and also contains time-series observations of traffic flow, speed, and occupancy. Traffic data were collected at regular intervals (e.g., every 5 min) by distributed detectors over consecutive days. For each time step, three features are recorded: traffic flow, average speed, and average occupancy. Occupancy reflects the degree of road utilization and can be interpreted in terms of temporal and spatial occupancy.

For model training and evaluation, the entire dataset was divided into training, validation, and test sets with a ratio of 7:1:2. Specifically, the first 70% of the data were used for training, the subsequent 10% for validation, and the remaining 20% for testing. During preprocessing, Z-score normalization was applied to the input data to preserve the original distribution characteristics while eliminating scale differences among features.

All experiments were implemented using the PyTorch 1.12.1 framework with Python 3.8 and CUDA 12.1. The hardware environment consisted of a 13th Gen Intel(R) Core(TM)

i7-13900KF CPU and an NVIDIA RTX 3060 GPU for single-GPU training. The input sequence length was set to 12 historical time steps, and the prediction horizons were set to 3, 6, and 12 time steps, corresponding to forecasting traffic conditions for the next 15, 30, and 60 min based on 60 min of historical observations. The Adam optimizer was adopted with an initial learning rate of 0.001, batch size of 64, and 200 training epochs. For fair comparison, all baseline models were trained under the same experimental settings, including optimizer, learning rate, batch size, and number of epochs.

3.2. Evaluation Metrics

To provide a comprehensive evaluation of the proposed model, performance is assessed from three perspectives: deviation from ground-truth values, relative prediction error, and overall stability of the forecasting model. Accordingly, three widely adopted metrics in traffic flow prediction are employed, namely Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). MAE measures the average magnitude of absolute differences between predicted values and corresponding ground-truth values. By avoiding cancellation between positive and negative errors, MAE provides a direct and intuitive assessment of prediction accuracy. In traffic flow forecasting, lower MAE values indicate that predicted results are closer to actual observations. MAPE expresses prediction error in percentage form and reflects the average relative deviation between predicted and actual values. It provides a normalized evaluation of forecasting accuracy, facilitating comparison across different scales. Smaller MAPE values indicate higher prediction precision. RMSE is defined as the square root of the mean of squared differences between predicted and ground-truth values. Due to its sensitivity to large deviations, RMSE reflects the dispersion of residuals and provides insight into the stability and overall performance of the prediction model. In traffic flow forecasting, lower RMSE values indicate better predictive performance and reduced variability in estimation errors. The evaluation metrics are calculated as follows:

$$\mathrm{MAE} = \frac{1}{n} \cdot \sum_{u=1}^{U} |y_i - \hat{y}_i| \tag{12}$$

$$\mathrm{MAPE} = \sum_{u=1}^{U} \left(\frac{|y_i - \hat{y}_i|}{y_i} \right) \cdot \frac{100\%}{n} \tag{13}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \cdot \sum_{u=1}^{U} (y_i - \hat{y}_i)^2} \tag{14}$$

here, y_i denotes the ground-truth value, $\hat{y}_i$ represents the predicted value, and u corresponds to the total number of samples.

3.3. Ablation Experiment of Improved Strategy

To evaluate the effectiveness and individual contribution of each proposed optimization strategy, an ablation study was conducted under identical experimental settings, including the same datasets, training configurations, and hyperparameters. The impact of the three key components—ATD layer, SIRT, and MHA—was separately examined for the 12-step prediction task (corresponding to traffic flow forecasting from 5 to 60 min). The experimental results are summarized in Table 1.

Table 1. Ablation study results of the proposed model.

Dataset	Model	MAE	MAPE	RMSE
PEMSBAY	ASTGCN	2.103	5.300	4.770
	ASTGCN + ATD	2.121	5.016	4.520
	ASTGCN + ATD + SIRT	2.068	4.993	4.505
	ASTGCN + ATD + SIRT + MHA	**1.904**	**4.591**	**4.416**
PEMSM	ASTGCN	3.572	9.480	6.587
	ASTGCN + ATD	3.308	9.244	6.229
	ASTGCN + ATD + SIRT	3.215	8.018	5.904
	ASTGCN + ATD + SIRT + MHA	**3.102**	**7.821**	**5.546**

Table 1 presents the performance variations of the model on the 60 min forecasting task over the PEMSBAY and PEMSM datasets after the progressive introduction of the ATD, SIRT, and MHA modules. Using ASTGCN as the baseline, when the ATD module was incorporated alone on the PEMSBAY dataset, the MAE slightly increased from 2.103 to 2.121, while the MAPE decreased from 5.300 to 5.016 and the RMSE decreased from 4.770 to 4.520. These results indicate that ATD significantly reduces the MAPE and RMSE values. This improvement can be attributed to the fact that ATD captures the underlying patterns of periodic traffic data more accurately, thereby yielding more stable predictions. After introducing the SIRT module, the MAE, MAPE, and RMSE further decreased to 2.068, 4.993, and 4.505, respectively, indicating that SIRT enhances the ability of graph convolution to exploit correlations in the feature space through a lightweight attention mechanism. Finally, in the ASM-STGCN model integrating MHA, the MAE, MAPE, and RMSE reached 1.904, 4.591, and 4.416, respectively, representing reductions of 9.5%, 13.4%, and 7.4% compared with the baseline ASTGCN. On the PEMSM dataset, the baseline ASTGCN achieved MAE, MAPE, and RMSE values of 3.572, 9.480, and 6.587, respectively. After incorporating ATD, these metrics decreased to 3.308, 9.244, and 6.229, respectively. With the addition of SIRT, they were further reduced to 3.215, 8.018, and 5.904. The final ASM-STGCN model achieved MAE, MAPE, and RMSE values of 3.102, 7.821, and 5.546, corresponding to reductions of 13.2%, 17.5%, and 15.8%, respectively, relative to the baseline model. These ablation results strongly demonstrate the effectiveness of each module. Specifically, ATD improves long-term forecasting performance by generating a multi-period adaptive temporal graph, which enables more accurate modeling of long-range periodic dependencies in traffic flow. SIRT further refines spatial feature extraction and effectively captures both higher-order spatial correlations among multiple nodes and local interaction features between adjacent nodes. In addition, MHA integrates feature representations at different temporal resolutions through a multi-scale multi-head attention fusion mechanism, thereby enabling more precise modeling of the dynamic variations in traffic flow.

The ablation results under the 3-step (15 min) and 6-step (30 min) forecasting horizons are illustrated in Figure 5. The results indicate that each proposed improvement component contributes positively to traffic flow prediction performance across different forecasting horizons. Meanwhile, prediction errors gradually increase as the forecasting horizon extends, reflecting the accumulation of uncertainty in longer-term forecasting tasks. For short-term prediction (15 min), traffic state variations are primarily influenced by local spatiotemporal dependencies. Under this setting, the ATD layer dynamically constructs adaptive temporal graphs to capture flow evolution patterns between adjacent time intervals, effectively modeling temporal continuity and smoothness of traffic states. Consequently, all three evaluation metrics (MAE, MAPE, and RMSE) show noticeable reductions compared with the baseline. When the forecasting horizon extends to 30 min, uncertainty in traffic flow increases and the spatial influence propagates over a wider range.

In this scenario, the SIRT module enhances spatial interaction modeling by capturing not only correlations between directly adjacent nodes but also higher-order and non-local spatial dependencies across regions. This mechanism alleviates error accumulation caused by spatial information attenuation and improves medium-term prediction performance. Furthermore, the MHA module performs parallel extraction of multi-scale temporal features and adaptively fuses them through attention-based weighting. This enables the model to simultaneously respond to short-term fluctuations and long-term evolutionary patterns in traffic flow. As a result, the growth of prediction error remains relatively moderate as the forecasting horizon increases. Traffic flow forecasting is inherently a spatiotemporally coupled dynamic process. Temporal dynamics exhibit memory effects and periodic patterns, while spatial interactions involve both localized correlations and global diffusion characteristics. The proposed ATD, SIRT, and MHA modules respectively address temporal dynamic modeling, spatial structural learning, and multi-scale feature fusion. By integrating these complementary mechanisms, the proposed framework better approximates the evolutionary characteristics of real-world traffic systems and achieves stable and consistent predictive performance across short-, medium-, and long-term forecasting horizons, providing a solid structural foundation for further long-range prediction tasks.

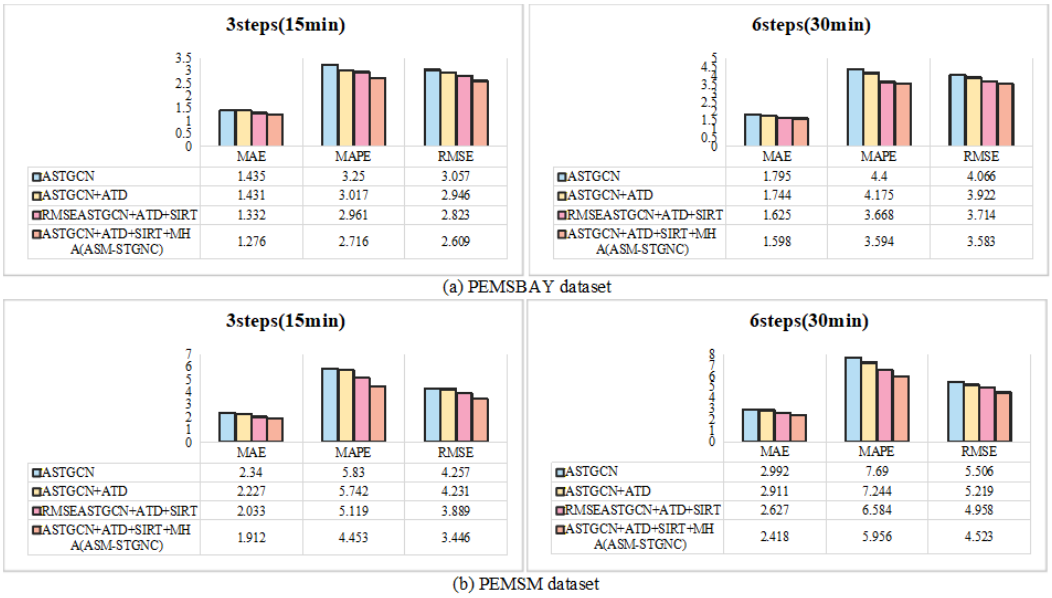

Figure 5. Ablation comparison results under 15 min and 30 min forecasting horizons.

3.4. Comparative Analysis

To further evaluate the generalization ability and prediction accuracy of the proposed ASM-STGCN framework, comparative experiments were conducted against nine representative traffic forecasting models under the same dataset and experimental settings to ensure fairness. The selected baseline methods include both traditional statistical approaches and state-of-the-art deep learning-based spatiotemporal models, namely HA [25] (Historical Average), LSTNet [26], DCRNN [27], STGCN [28], Graph WaveNet [29], GMAN [30], ASTGCN [31], MTGNN [32], AGCRN [33], and SAGCN-SST [34]. Quantitative evaluation and visualization analyses were performed to provide a comprehensive comparison. The experimental results are summarized in Table 2 and illustrated in Figure 6.

Table 2. Performance metrics for each model on different datasets.

Dataset	Model	3 Steps (15 min)			6 Steps (30 min)			12 Steps (60 min)		
		MAE	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**
PEMSM	HA	3.917	9.900	7.077	3.920	9.920	7.083	3.925	9.950	7.095
	LSTNet	2.423	5.730	4.308	5.132	12.220	8.951	6.624	16.720	10.881
	STGCN	2.124	5.020	4.051	2.783	6.960	5.532	3.374	8.740	6.695
	DCRNN	2.213	5.330	4.143	2.907	7.410	5.679	3.670	9.810	7.138
	GWN	2.130	5.000	3.992	2.715	6.750	5.332	3.266	8.470	6.431
	ASTGCN	2.340	5.830	4.257	2.992	7.690	5.506	3.572	9.480	6.587
	GMAN	2.877	7.250	5.711	3.084	7.770	6.171	3.988	10.020	7.897
	MTGNN	2.120	5.020	4.032	2.687	6.700	5.373	3.204	8.240	6.496
	AGCRN	2.167	5.190	4.073	2.769	6.890	5.479	3.358	8.550	6.733
	SAGCN-SST	2.011	4.774	3.685	2.576	6.348	4.837	3.154	7.987	5.846
	ASM-STGCN	**1.912**	**4.453**	**3.446**	**2.418**	**5.956**	**4.523**	**3.102**	**7.821**	**5.546**
PEMSBAY	HA	3.333	8.100	6.687	3.333	8.100	6.686	3.332	8.100	6.685
	LSTNet	1.643	3.470	3.224	2.383	5.040	4.375	2.974	6.860	5.515
	STGCN	1.327	2.790	2.827	1.698	3.810	3.887	2.055	5.020	4.748
	DCRNN	1.377	2.960	2.867	1.726	3.970	3.905	2.091	4.990	4.798
	GWN	1.322	2.780	2.759	1.660	3.750	3.737	1.991	4.750	4.562
	ASTGCN	1.435	3.250	3.057	1.795	4.400	4.066	2.103	5.300	4.770
	GMAN	1.802	4.470	4.219	1.794	4.400	4.143	2.186	5.290	5.034
	MTGNN	1.334	2.840	2.849	1.658	3.770	3.800	1.950	4.670	4.491
	AGCRN	1.354	2.940	2.856	1.670	3.840	3.818	1.964	4.690	4.570
	SAGCN-SST	1.322	2.794	2.776	1.765	4.025	4.152	2.293	4.906	4.745
	ASM-STGCN	**1.276**	**2.716**	**2.609**	**1.598**	**3.594**	**3.583**	**1.904**	**4.591**	**4.416**

Table 2 shows that the HA method exhibits poor predictive performance on nonlinear traffic flow data, indicating that it is not well suited for nonlinear dynamic scenarios. On the PEMSM dataset, SAGCN-SST achieves the best performance among all baseline models. Taking this model as the reference, ASM-STGCN reduces MAE, MAPE, and RMSE by 0.099, 0.312, and 0.239, respectively, at the 15 min forecasting horizon, corresponding to relative improvements of 4.9%, 6.5%, and 6.5%. At the 30 min horizon, the three metrics are further reduced by 0.158, 0.392, and 0.314, yielding relative improvements of 6.1%, 6.2%, and 6.5%, respectively. At the 60 min horizon, ASM-STGCN still outperforms the reference model, with reductions of 0.052, 0.166, and 0.300 in MAE, MAPE, and RMSE, corresponding to relative improvements of 1.6%, 2.1%, and 5.1%, respectively. On the PEMSBAY dataset, at the 15 min forecasting horizon, ASM-STGCN reduces MAE, MAPE, and RMSE by 0.046, 0.078, and 0.167, respectively, compared with the strongest baseline, SAGCN-SST, corresponding to relative improvements of 3.5%, 2.8%, and 6.0%. At the 30 min horizon, compared with MTGNN, which achieves the best overall performance among the baselines, the three metrics are reduced by 0.060, 0.176, and 0.217, corresponding to relative improvements of 3.6%, 4.7%, and 5.7%, respectively. At the 60 min horizon, again compared with MTGNN, ASM-STGCN achieves reductions of 0.046, 0.079, and 0.075 in MAE, MAPE, and RMSE, corresponding to relative improvements of 2.4%, 1.7%, and 1.7%, respectively. These results demonstrate that ASM-STGCN achieves superior predictive accuracy at most forecasting horizons, thereby fully verifying the effectiveness of the synergistic integration of the three core modules, namely ATD, SIRT, and MHA. Specifically, ATD models the inherent periodic dependencies in traffic flow data by constructing a multi-period adaptive temporal graph. SIRT enhances gradient propagation and feature stability in deep networks through a lightweight Shaped Attention mechanism and a three-branch skip-connection design. MHA, through a cross-scale multi-head attention fusion mechanism, captures dynamic correlations and intrinsic patterns across different temporal resolutions. Nevertheless, Table 2 also reveals that the magnitude of performance improvement gradually decreases

as the forecasting horizon increases. In particular, for the 60 min long-term traffic flow forecasting task, the gains achieved by all methods are relatively limited, suggesting that there remains room for further improvement in modeling ultra-long-term dependencies. Overall, ASM-STGCN demonstrates relatively stable predictive performance across the 15 min, 30 min, and 60 min forecasting tasks, providing a high-accuracy and high-efficiency solution for real-time traffic flow prediction in intelligent transportation systems.

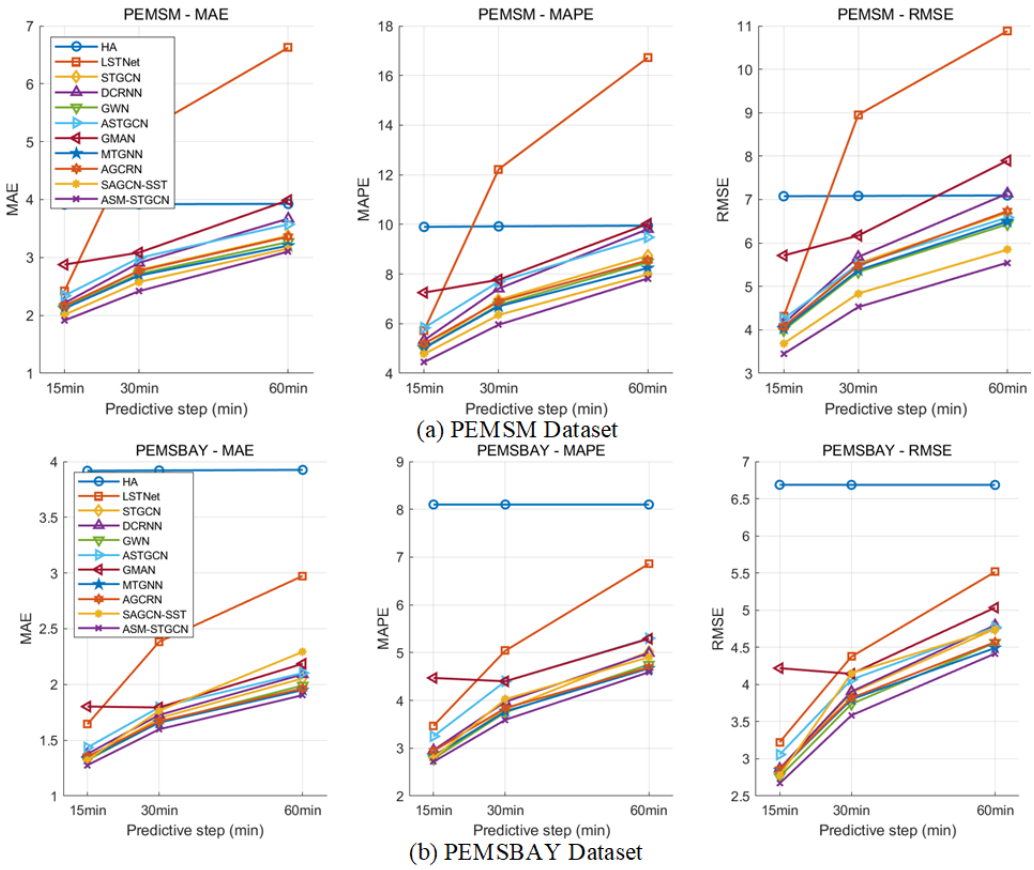

Figure 6. Performance comparison curves of different models under the same evaluation metric.

The comparison curves of different algorithms under the same evaluation metric are illustrated in Figure 6. The results indicate that the proposed ASM-STGCN consistently outperforms the statistical baseline HA and the non-graph-based deep learning model LSTNet in the traffic flow forecasting task. Moreover, the proposed method remains competitive among state-of-the-art graph-based spatiotemporal forecasting models. The HA model, as a classical statistical forecasting method, is primarily suitable for linear and stationary prediction scenarios. However, real-world traffic systems are influenced by dynamic road network conditions, unexpected events, and multi-factor coupling effects, exhibiting pronounced nonlinearity and temporal variability. Consequently, HA presents inherent limitations when applied to complex traffic forecasting tasks. LSTNet combines convolutional and recurrent neural networks to extract local temporal patterns. Nevertheless, its decoupled temporal modeling architecture does not explicitly capture spatial propagation mechanisms in traffic networks. Since LSTNet performs time-series prediction at individual nodes without modeling inter-node interactions, prediction errors tend to accumulate as the forecasting horizon increases. Graph-based forecasting models explicitly incorporate spatial topological constraints. Methods such as STGCN and DCRNN rely on predefined static graph structures, which effectively capture basic spatial correlations but may struggle to adapt to dynamically evolving traffic states. In contrast, adaptive graph

learning models such as MTGNN, AGCRN, and the proposed ASM-STGCN dynamically infer node correlations, enabling more accurate modeling of traffic influence propagation. Specifically, ASM-STGCN enhances temporal dynamic dependency modeling through the ATD module, captures higher-order spatial correlations via the SIRT module, and integrates multi-granularity temporal features through the MHA mechanism. These complementary strategies allow the model to better approximate the evolutionary patterns of complex traffic systems, resulting in more accurate and robust forecasting performance. From the quantitative comparisons in Table 2 and the trend analysis in Figure 6, several observations can be summarized. First, statistical forecasting methods such as HA exhibit clear limitations when modeling nonlinear traffic dynamics. Second, non-graph-based models like LSTNet perform reasonably well in short-term prediction but experience performance degradation as the prediction horizon increases. Third, graph-based models generally outperform non-graph-based approaches, and adaptive or dynamically learned graph structures (e.g., MTGNN, AGCRN, and ASM-STGCN) further improve prediction accuracy compared with models relying on predefined static graphs (e.g., DCRNN and STGCN).

3.5. Visualization Analysis of Prediction Results

To qualitatively assess the real-world forecasting performance of the proposed model, a representative day from the PEMSBAY dataset was selected for visualization analysis. To clearly illustrate the variation trends of prediction results, four models from different methodological categories were compared: the non-graph-based model LSTNet, the fixed-graph-based model STGCN, the adaptive graph learning model MTGNN, and the proposed ASM-STGCN framework. The visualization results are presented in Figure 7.

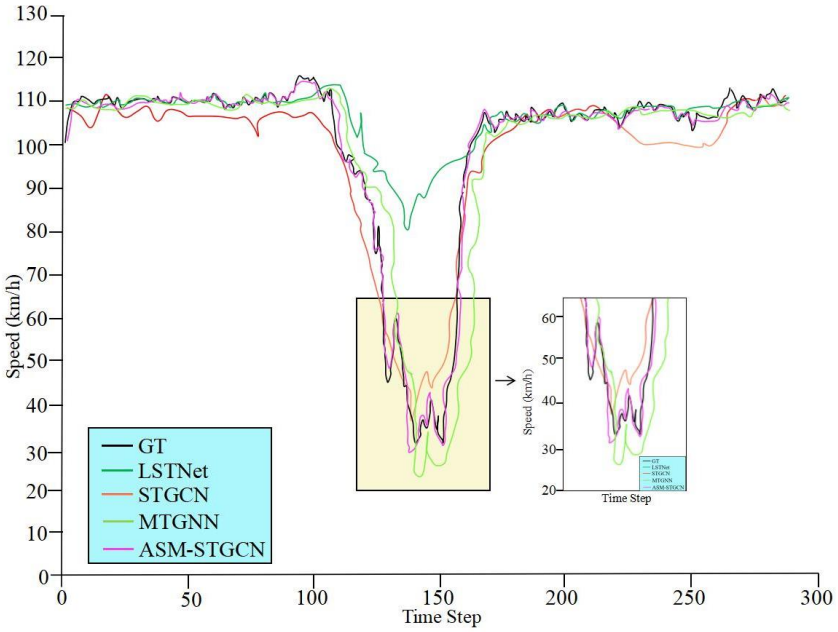

Figure 7. Visualization comparison of prediction results.

The visualization results in Figure 7 indicate that all compared models are capable of capturing the overall temporal evolution pattern of traffic flow, demonstrating sensitivity to the general trend of traffic dynamics. However, noticeable differences can be observed in prediction accuracy and curve alignment with ground-truth values. In particular, graph-based models—including STGCN, MTGNN, and the proposed ASM-STGCN—consistently produce predictions that more closely follow the true traffic flow variations compared with the non-graph-based model LSTNet. This observation suggests that incorporating graph structures enables the model to more effectively capture spatial correlations and inter-node

dependencies within the traffic network. By explicitly modeling spatial topology and propagation mechanisms, graph-based approaches achieve more comprehensive spatiotemporal representation learning, thereby improving overall forecasting performance.

3.6. Model Parameter Analysis

To assess the performance of ASM-STGCN in terms of model size and computational efficiency, Figure 8 presents a PM bubble chart, where the horizontal axis denotes parameter size (MB) and the vertical axis denotes the MAE on the 12-step (60 min) forecasting task over the PEMSBAY dataset. The bubble size is proportional to the number of model parameters. ASM-STGCN achieves an MAE of 1.904 with a parameter size of 1.32 MB. Compared with the baseline ASTGCN (1.18 MB, MAE 2.103), the proposed model increases the parameter size by only 0.14 MB (11.9%) while reducing the MAE by 9.5%. By contrast, although LSTNet is more lightweight (0.5 MB), its MAE reaches 2.95, suggesting that it has limited capability in capturing complex spatiotemporal dependencies. Moreover, DCRNN and MTGNN, both with parameter sizes of approximately 1.0 MB, still underperform ASM-STGCN in prediction accuracy. This result indicates that ATD effectively models periodic traffic dependencies through multi-period adaptive adjacency matrices at hourly, daily, and weekly scales with only a small parameter increase. SIRT further improves efficiency and stability by combining lightweight Shaped Attention with a three-branch skip-connection design, reducing the complexity of self-attention from quadratic to linear order while enhancing global nonlinear feature interactions through the MLP branch. In addition, MHA captures traffic fluctuations across multiple temporal resolutions by means of cross-scale multi-head attention fusion. Therefore, ASM-STGCN achieves the best overall predictive accuracy among the compared models with only a modest parameter overhead, demonstrating a favorable balance between efficiency and accuracy for deployment in resource-constrained traffic forecasting scenarios.

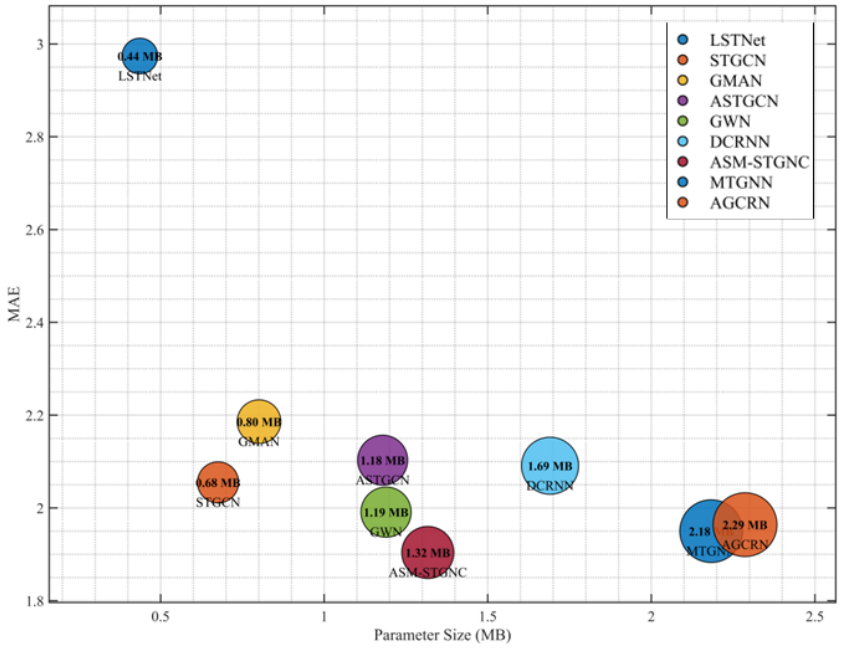

Figure 8. PM bubble chart for parameter size analysis.

4. Conclusions

In this study, to address the limitation of existing approaches that insufficiently account for periodic characteristics in spatiotemporal data during spatial feature extraction, an adaptive spatiotemporal graph convolutional framework termed ASM-STGCN was

developed based on the ASTGCN model. The core component of the proposed framework is a multi-period adaptive temporal graph generation layer (ATD), which explicitly models the influence of different temporal periodicities on the forecasting structure. By dynamically constructing temporal graphs at multiple time scales and capturing interdependencies among spatiotemporal nodes, the ATD module enhances the model's capability to represent temporal dynamics and periodic correlations. Furthermore, a lightweight feature refinement module, SIRT, was introduced following the graph convolution operation. By initializing and regulating the self-attention matrix, SIRT effectively captures subtle variations in graph-convolved features and facilitates stable feature propagation in deeper network layers. In addition, a multi-scale multi-head self-attention (MHA) fusion mechanism was designed to integrate feature representations across different temporal resolutions. This mechanism enables the model to learn intrinsic correlations and dynamic interactions among multi-scale traffic patterns, thereby improving generalization performance. Experiments were conducted on two public datasets, PEMSBAY and PEMSM. The ablation results verified the positive contributions of the proposed ATD, SIRT, and MHA modules to traffic flow prediction. Compared with the baseline method, the proposed model achieved average improvements of 14% in MAE, 19% in MAPE, and 15% in RMSE across the two datasets. In addition, the comparative experiments further demonstrated the effectiveness of the ASM-STGCN model. Future work will focus on developing more effective adaptive temporal graph generation strategies, further optimizing spatial feature extraction mechanisms, and exploring the integration of efficient sequence modeling architectures such as Mamba-like models to enhance long-range dependency modeling. In addition, the proposed framework may be extended to other graph-based spatiotemporal prediction tasks, including pedestrian mobility forecasting and air quality prediction.

Author Contributions: Conceptualization, G.L., B.W., K.G. and G.Y.; methodology, G.L. and B.W.; software, G.Y.; validation, G.L., B.W. and G.Y.; formal analysis, G.L.; investigation, B.W. and K.G.; resources, G.L.; data curation, G.L., B.W., K.G. and G.Y.; writing—original draft preparation, G.L. and B.W.; writing—review and editing, K.G. and G.Y.; visualization, G.Y.; supervision, G.L.; project administration, G.L. and B.W.; funding acquisition, G.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This document is supported by National Natural Science Foundation of China under Grant 624661032, Project Fund of Gansu Water Resources Research Institute under Grant LZJT524289, and the Gansu Provincial Department of Education "Innovation Star" Project for Outstanding Postgraduates under Grant 2026CXZX-633.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets are available at https://github.com/liyaguang/DCRNN accessed on 1 November 2025. The raw code supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: The authors thank the editor and reviewers for their valuable suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shin, Y.; Yoon, Y. PGCN: Progressive graph convolutional networks for spatial–temporal traffic forecasting. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 7633–7644. [CrossRef]
2. Carianni, A.; Gemma, A. Overview of traffic flow forecasting techniques. *IEEE Open J. Intell. Transp. Syst.* **2025**, *6*, 848–882. [CrossRef]
3. Lee, S.; Park, C. Temporal graph learning recurrent neural network for traffic forecasting. *arXiv* **2024**, arXiv:2406.02726. [CrossRef]

4. Naheliya, B.; Redhu, P.; Kumar, K. MFOA-Bi-LSTM: An optimized bidirectional long short-term memory model for short-term traffic flow prediction. *Phys. A* **2024**, *634*, 129448. [CrossRef]
5. Chauhan, N.S.; Kumar, N.; Eskandarian, A. A novel confined attention mechanism driven Bi-GRU model for traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 9181–9191. [CrossRef]
6. Ma, Y.; Lou, H.; Yan, M.; Sun, F.; Li, G. Spatio-temporal fusion graph convolutional network for traffic flow forecasting. *Inf. Fusion* **2024**, *104*, 102196. [CrossRef]
7. Feng, Q.; Li, B.; Liu, X.; Gao, X.; Wan, K. Low-high frequency network for spatial–temporal traffic flow forecasting. *Eng. Appl. Artif. Intell.* **2025**, *158*, 111304. [CrossRef]
8. Liang, Y.; Liu, S.; Bai, Y.; Gong, Y.; Zhu, T. DSM-STWave: Enhancing traffic flow prediction for both offline and online scenarios. *Neurocomputing* **2025**, *661*, 131836. [CrossRef]
9. Wang, H.; Zhang, R.; Cheng, X.; Yang, L. Hierarchical traffic flow prediction based on spatial-temporal graph convolutional network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16137–16147. [CrossRef]
10. Zhang, A. Dynamic graph convolutional networks with temporal representation learning for traffic flow prediction. *Sci. Rep.* **2025**, *15*, 17270. [CrossRef]
11. Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; Li, H. T-GCN: A temporal graph convolutional network for traffic prediction. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 3848–3858. [CrossRef]
12. Cao, S.; Wu, L.; Zhang, R.; Wu, D.; Cui, J.; Chang, Y. A spatiotemporal multiscale graph convolutional network for traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 8705–8718. [CrossRef]
13. Yang, H.; Li, Z.; Qi, Y. Predicting traffic propagation flow in urban road network with multi-graph convolutional network. *Complex Intell. Syst.* **2024**, *10*, 23–35. [CrossRef]
14. Liu, Z.; Ding, F.; Dai, Y.; Li, L.; Chen, T.; Tan, H. Spatial–temporal graph convolution network model with traffic fundamental diagram information informed for network traffic flow prediction. *Expert Syst. Appl.* **2024**, *249*, 123543. [CrossRef]
15. Huang, X.; Wang, H.; Li, Y.; Huang, L.; Zhao, H. Reservation-based traffic signal control for mixed traffic flow at intersections. *Phys. A* **2024**, *633*, 129426. [CrossRef]
16. Yang, H.; Zhao, X.; Luan, S.; Chai, S. A traffic dynamic operation risk assessment method using driving behaviors and traffic flow data: An empirical analysis. *Expert Syst. Appl.* **2024**, *249*, 123619. [CrossRef]
17. Zhai, C.; Li, K.; Zhang, R.; Peng, T.; Zong, C. Phase diagram in multi-phase heterogeneous traffic flow model integrating the perceptual range difference under human-driven and connected vehicles environment. *Chaos Solitons Fractals* **2024**, *182*, 114791. [CrossRef]
18. Li, Z.L.; Zhang, G.W.; Yu, J.; Xu, L.Y. Dynamic graph structure learning for multivariate time series forecasting. *Pattern Recognit.* **2023**, *138*, 109423. [CrossRef]
19. Bresson, R.; Nikolentzos, G.; Panagopoulos, G.; Chatzianastasis, M.; Pang, J.; Vazirgiannis, M. KAGNNs: Kolmogorov–Arnold networks meet graph learning. *arXiv* **2024**, arXiv:2406.18380.
20. Li, H.; Yang, S.; Song, Y.; Luo, Y.; Li, J.; Zhou, T. Spatial dynamic graph convolutional network for traffic flow forecasting. *Appl. Intell.* **2023**, *53*, 14986–14998. [CrossRef]
21. He, B.; Hofmann, T. Simplifying transformer blocks. *arXiv* **2023**, arXiv:2311.01906.
22. Jia, C.; He, H.; Zhou, J.; Li, K.; Li, J.; Wei, Z. A performance degradation prediction model for PEMFC based on bidirectional long short-term memory and multi-head self-attention mechanism. *Int. J. Hydrogen Energy* **2024**, *60*, 133–146. [CrossRef]
23. Chen, W.; Chen, L.; Xie, Y.; Cao, W.; Gao, Y.; Feng, X. Multi-range attentive bicomponent graph convolutional network for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA*; AAAI Press: Palo Alto, CA, USA, 2020; pp. 3529–3536.
24. Kanji, G.K. Performance measurement system. *Total Qual. Manag.* **2002**, *13*, 715–728. [CrossRef]
25. Kong, X.; Zhou, W.; Shen, G.; Zhang, W.; Liu, N.; Yang, Y. Dynamic graph convolutional recurrent imputation network for spatiotemporal traffic missing data. *Knowl.-Based Syst.* **2023**, *261*, 110188. [CrossRef]
26. Huang, C.; Yang, M. Memory long and short term time series network for ultra-short-term photovoltaic power forecasting. *Energy* **2023**, *279*, 127961. [CrossRef]
27. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv* **2017**, arXiv:1707.01926.
28. Yan, S.; Xiong, Y.; Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA*; AAAI Press: Palo Alto, CA, USA, 2018; pp. 7444–7452.
29. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Graph WaveNet for deep spatial–temporal graph modeling. In *International Joint Conference on Artificial Intelligence 2019, Macao, China*; AAAI Press: Palo Alto, CA, USA, 2019; pp. 1907–1913.
30. Zheng, C.; Fan, X.; Wang, C.; Qi, J. GMAN: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA*; AAAI Press: Palo Alto, CA, USA, 2020; pp. 1234–1241.

31. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention based spatial–temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA*; AAAI Press: Palo Alto, CA, USA, 2019; pp. 922–929.
32. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; ACM: New York, NY, USA, 2020; pp. 753–763.
33. Bai, L.; Yao, L.; Li, C.; Wang, X.; Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17804–17815.
34. Zheng, G.; Chai, W.K.; Katos, V. A dynamic spatial–temporal deep learning framework for traffic speed prediction on large-scale road networks. *Expert Syst. Appl.* **2022**, *195*, 116585. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.