International Journal of Digital Earth

ISSN: 1753-8947 (Print) 1753-8955 (Online) Journal homepage: www.tandfonline.com/journals/tjde20

CrossGraphNet: a cross-spatiotemporal graph-based method for traffic speed reconstruction using remote sensing vehicle detection

Yan Zhang, Mei-Po Kwan, Jiannan Cai, Jianying Wang & Peifeng Ma

To cite this article: Yan Zhang, Mei-Po Kwan, Jiannan Cai, Jianying Wang & Peifeng Ma (2025) CrossGraphNet: a cross-spatiotemporal graph-based method for traffic speed reconstruction using remote sensing vehicle detection, International Journal of Digital Earth, 18:1, 2528640, DOI: 10.1080/17538947.2025.2528640

To link to this article: https://doi.org/10.1080/17538947.2025.2528640

Published online: 15 Jul 2025.

INTERNATIONAL JOURNAL OF DIGITAL EARTH
2025, VOL. 18, NO. 1, 2528640
https://doi.org/10.1080/17538947.2025.2528640

CrossGraphNet: a cross-spatiotemporal graph-based method for traffic speed reconstruction using remote sensing vehicle detection

Yan Zhang [a,b], Mei-Po Kwan [a,b], Jiannan Cai[a,b], Jianying Wang[a,b] and Peifeng Ma[a,b]

[a]Institute of Space and Earth Information Science, The Chinese University of Hong Kong, Hong Kong SAR, People's Republic of China; [b]Department of Geography and Resource Management, The Chinese University of Hong Kong, Hong Kong SAR, People's Republic of China

ABSTRACT
Existing traffic flow modeling approaches typically rely on real-time observations, such as road sensors or GPS trajectories, which constrain their research scope and application scenarios. This study proposes a novel cross-spatiotemporal graph-based network method to rapidly reconstruct traffic flow speed based on remote sensing images. The method is designed to address the challenges of traffic modeling in the absence of ground observation data. Combining high-resolution remote sensing imagery, vehicle object detection, and graph modeling technology, our approach could handle the discontinuous spatiotemporal graph information. The method incorporates two key modules: a two-layer masked structure mechanism and a cross-spatiotemporal attention computation. This innovative design enables the model to continuously synthesize learning from discontinuous remote sensing images and sparse ground-based sensor data during pre-training, optimizing its parameters and improving prediction accuracy over time. Once pre-trained, the graph model can directly estimate street-level traffic flow speed based solely on remote sensing images. Our results demonstrate state-of-the-art performance (MSE=40.117, MAE=4.768, RMSE=6.334, RSE=0.228), outperforming previous graph-based and sequence-based models. This study showcases the potential of utilizing remote sensing techniques to reconstruct traffic speed in urbanizing regions. It can even be used in scenarios lacking sufficient ground stations and with discontinuous remote sensing data, and enables low-cost, large-scale, and multi-temporal traffic flow speed reconstruction.

ARTICLE HISTORY
Received 25 March 2025
Accepted 27 June 2025

KEYWORDS
Traffic flow reconstruction; GeoAI; vehicle detection; graph neural network; multimodal fusion; ground-based sensors

Highlights

- Reconstruct traffic speed using remote sensing imagery and graph neural networks.
- CrossGraphNet with two-layer masked and cross-spatiotemporal attention to handle discontinuous data.
- Combines with ground-based sensor data and uses only satellite imagery for estimation.

1. Introduction

Traffic flow modeling is fundamental to intelligent transportation systems, offering essential data support for urban planning, traffic management, and control (Boukerche, Tao, and Sun 2020; Deng et al. 2023). Current traffic flow modeling methods predominantly rely on real-time observational data from road sensors or vehicle-mounted GPS, using spatio-temporal models to reconstruct and predict traffic speeds (Kan et al. 2018; Li, Zhang, and Chen 2022; Yu et al. 2020). However, these conventional methods face limitations. Firstly, the coverage of real-time speed observation data is often restricted (Scheibenreif, Mommert, and Borth 2022), with many road areas lacking observation data, particularly in urbanized regions (Xu et al. 2022; Xu, Li, and Wen 2018; Yu, Lee, and Sohn 2020). Secondly, the collection of

CONTACT Mei-Po Kwan mpkwan@cuhk.edu.hk
This article has been corrected with minor changes. These changes do not impact the academic content of the article.

real-time data necessitates significant infrastructure investment, leading to high costs and challenges in large-scale deployment (Asensio et al. 2020).

Deep learning has revolutionized various domains of data analysis, with Convolutional Neural Networks (CNNs) achieving remarkable success in the processing of structured data, particularly in image analysis, time series processing, and geospatial applications (Ma et al. 2019). However, CNNs, designed primarily for grid-like data structures, face inherent limitations when handling non-Euclidean data such as road networks. This limitation led to the development of Graph Neural Networks (GNNs), which excel at capturing complex spatial relationships through sparse adjacency matrices (H. Chen et al. 2022; Yi et al. 2021; S. Zhao et al. 2024). The versatility of GNNs has led to their widespread adoption in various fields, from social network analysis (Z. Wu et al. 2020) to molecular structure prediction in biochemistry (Stokes et al. 2020), and increasingly in traffic modeling.

A particularly promising but challenging frontier lies in the integration of GNNs with widely-covered remote sensing imagery for transportation system modeling. Remote sensing data has shown potential in vehicle detection and traffic volume estimation (Wu, Guo et al. 2021; Wu, Zhu et al. 2021). Through vehicle object detection results based on high-resolution remote sensing imagery, vehicle density on roads can be estimated, providing important parameters for traffic modeling (Bouguettaya et al. 2021; Leitloff et al. 2014).

However, existing studies on remote sensing-based vehicle detection are limited mainly to static detection, such as being able to detect only traffic density without inferring traffic speed (Balid, Tafish, and Refai 2017; Gao et al. 2024; Macioszek and Kurek 2021). This is due to the relatively low temporal resolution of remote sensing images (Min et al. 2024; Y. Zhang, Chen et al. 2021), and discontinuous input makes it difficult to meet the requirements of traditional traffic models (Y. Chen et al. 2021). This disconnect means the critical gap in the application of remote sensing technology to intelligent transportation systems (Li and Zhu 2021).

Specifically, methods that use comprehensive CCTV coverage or sensor data can continuously record traffic speeds on road segments at high temporal resolutions, such as every minute or hour. In such cases, the speed at a previous time step (e.g. one minute ago) often directly influences subsequent states, making Spatial-Temporal sequential models the most natural and optimal choice. However, for remote sensing imagery, the revisit cycle is typically on the scale of several days rather than hours. It is important to note that while remote sensing imagery has a longer revisit cycle, the influence of traffic conditions from several days ago on conditions several days later remains uncertain.

To tackle the challenge of ground-based sensors lacking spatial coverage but offering high temporal resolution, and remote sensing images providing extensive spatial coverage but with low temporal resolution, this research aims to integrate vehicle detection data obtained from remote sensing images with sparse ground-based sensor data. The GNN model, commonly utilized in traffic modeling, is adopted to capture spatial relationships in road networks and reconstruct traffic flow speed at the street level. In this study, we introduce a novel method called Cross-GraphNet for traffic speed estimation, incorporating two crucial modules: a two-layer masking mechanism (Node Mask and Graph Mask) and a cross-temporal spatial attention structure. These components are designed to address missing data at both node and graph levels and effectively process discontinuous data, thereby enhancing the model's adaptability and resilience. This methodology requires minimal ground-based sensor data to handle intermittent remote sensing images and forecast traffic speeds. It transitions from depending on the immediate past state (a few minutes) to integrating information across multiple time intervals. Once pretrained, the model can accurately predict traffic speeds based solely on remote sensing images.

Our study is organized as follows: The second section reviews related work on the application of remote sensing data in traffic modeling. The third section introduces the model architecture of our CrossGraphNet graph neural network. The fourth section presents the experimental results, including model accuracy, comparison studies, and ablation experiments. Finally, the last section summarizes the study, discussing its advantages and limitations, as well as its potential for reconstructing traffic flow, particularly in the developing boundaries of cities.

2. Related work

2.1. Traffic speed estimation with sparse data

Traffic speed reconstruction research focuses primarily on reconstructing network-wide traffic conditions using limited observational data. A classical approach uses compressed sensing and spatiotemporal tensor decomposition, utilizing the low-rank properties of traffic flow to complete the matrix (Z. Chen et al. 2024; Jia et al. 2021). Spatiotemporal tensor decomposition models can effectively capture periodic patterns in traffic data (Wang, Cao, and Philip 2020). With the advancement of deep learning, graph-based methods have become mainstream (Huo et al. 2023; Zhang, Li, and Zhang 2024). For example, the classic spatio-temporal graph convolutional network (ST-GCN) model can reconstruct data by capturing spatio-temporal dependencies (Yu et al. 2021). Graph-based approaches combined with various sequence models such as ST-RNN, ST-LSTM, and CTLE effectively integrate temporal and spatial information (Cui et al. 2019). In particular, methods that incorporate Transformers and attention mechanisms excel in long sequence prediction (Zheng et al. 2020). Furthermore, incremental learning approaches using progressive learning strategies with graph models such as GraphSage and GAT enable feature extraction and state reconstruction on irregular road networks through multitemporal learning (Lv et al. 2020; Zhang, Liu, and Biljecki 2023). Multi-source data fusion is another important research direction, combining fixed sensor and floating vehicle data for hybrid modeling (Kashinath et al. 2021; T. Wang et al. 2021). However, these methods typically rely on high-frequency sensor data and perform poorly in scenarios with low sampling frequency and sparse sensor coverage.

2.2. Remote sensing for traffic research

Remote sensing technology opens up new possibilities for large-scale traffic monitoring. Recent years have witnessed significant progress in deep learning-based vehicle detection from remote sensing imagery. This progress includes the development of multi-scale object detection frameworks based on Faster R-CNN, specifically tailored for remote sensing scenarios, which notably enhance small vehicle detection performance (Yi, Su, and Chen 2021). The integration of spatial attention and similar methods effectively tackles challenges such as dense distribution of vehicles and vehicle orientation variations in remote sensing images (Lin et al. 2023; Ming et al. 2021; Yan et al. 2021), leading to the achievement of end-to-end vehicle detection (Cao et al. 2020; X. Wu et al. 2020). However, traditional traffic modeling methods typically necessitate short time intervals (e.g. 5 min) for sequence modeling (Abduljabbar et al. 2021; H. Zhang et al. 2022). The varying revisit cycles of remote sensing imagery (Li and Chen 2020) present a challenge in inferring dynamic parameters like speed from detected static features, including the non-continuous input features.

2.3. Challenges in remote sensing-based traffic speed modeling

The integration of remote sensing data with traffic modeling faces two key challenges. First, the scarcity of ground-truth data hinders model training and validation. While remote sensing provides extensive spatial coverage, it cannot directly measure vehicle speeds. Ground sensors, though accurate, are sparsely deployed, limiting their utility for large-scale model training (X. Wu et al. 2020). Second, the temporal resolution of remote sensing data is constrained by satellite revisit cycles and atmospheric conditions, creating a mismatch with conventional traffic modeling algorithms that require continuous time-series data (Y. Shi et al. 2024; Y. Zhang, Dong et al. 2020). Unlike ground sensors that provide minute or hourly data, remote sensing imagery has revisit cycles of several days, leading to spatial sparsity and temporal discontinuity that challenge conventional models (Y. Guo et al. 2022; Y. Zhang, Chen et al. 2021).

Current approaches to handling these challenges fall into two categories. Spatial-temporal graph-based methods, such as ST-GCN (S. Guo et al. 2019; Y. Wu, Tan et al. 2018; Yan, Xiong, and Lin 2018), require continuous temporal sequences, which are often unavailable in remote sensing data. Incremental learning methods (He, Chen, and Li 2011; Y. Wu, Chen et al. 2019) process time points sequentially but suffer from information loss between steps, preventing effective bidirectional information flow. These limitations highlight the need for new methodologies to better integrate remote sensing data into traffic modeling frameworks.

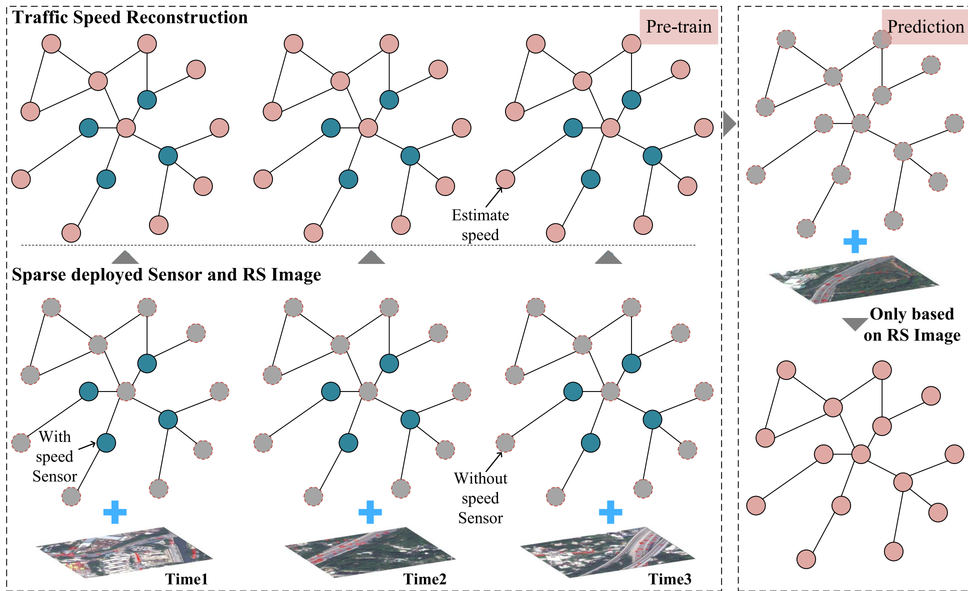

Figure 1. The illustrate of our research objective and questions.

3. Method

Our research focused on a real-world scenario, as illustrated in Figure 1. The study area is located in the urbanization region of Hong Kong, where only approximately 6% of the road segments are equipped with ground-based speed sensors. Our objective is to estimate and reconstruct the traffic speeds for all roads. Given the sparse coverage of ground-based sensors, we also want to incorporate widely covered remote sensing imagery for complementary purposes. Our goal is to predict street-level traffic speeds using only remote sensing imagery as input, following model pretraining.

Specifically, the complex urban road segments can be abstracted as a graph $G = (V, E)$, where $V = \{V_1, \ldots, V_n\}$ represents road segments, $E = \{E_{(0,1)}, \ldots, E_{(i,j)}\}$ represents connections between roads, and $H = \{H_1, \ldots, H_n\}$ represents the node feature matrix. In this study, 'traffic speed' refers to the spatially averaged speed of all vehicles traveling within a given road segment over a specific time interval. We want to estimate the average traffic speed (in km/h) for each road segment. A road's speed is influenced not only by its own capacity and vehicle density but also by the traffic conditions of neighboring roads. The introduction of GNNs can effectively address this problem. In GNN models, node (i.e. road) embeddings (where we treat road vehicle density as node embeddings) are built upon predefined GNN layers, aggregating information from a node's network neighbors through nonlinear transformations and message-passing architectures using different pooling schemes (e.g. mean/max/LSTM pooling). Multiple GNN layers can be used to capture short-range or long-range node dependencies.

To handle the proposed problem in the related work, we propose the CrossGraphNet Model (The sourcecode was shared at the Github repositories: github.com/yemanzhongting/CrossGraphNet). The model could process bi-directional contextual information, allow data missing at arbitrary time points, and predict missing graphs at any time point. It comprehensively considers the sequential and state information of the graph, and improves data utilization efficiency greatly. And the traditional sequence models often can only achieve unidirectional prediction. The cross-attention mechanism effectively handles such discontinuous data, autonomously learning and prioritizing crucial spatio-temporal information.

3.1. The double-layered mask structure

First, we want to solve the random missing data problem, which extends beyond simple temporal gaps to include spatial discontinuities in sensor coverage. The missing data issue refers not only to the lack of traffic data from sensors at specific time points but, more importantly, indicates the missing observational data for road segments that are not covered by sensors. Due to the strong randomness in traffic data collection, the

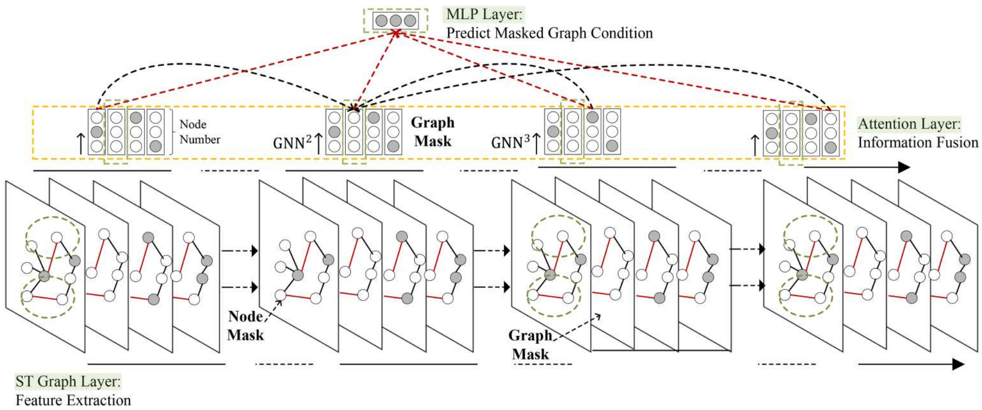

Figure 2. The CrossGraphNet model architecture. The model has a double-layered mask, including both node mask and graph mask.

road segments with missing data are inconsistent at different time points, necessitating a robust handling mechanism (Wang, Tong, and Nie 2023; T. Zhao et al. 2023). We propose a novel multitask masking approach that treats observations at different time points as independent tasks, enabling selective backpropagation based on available data while maintaining model integrity.

As shown in Figure 2, we designed a double-layered mask structure in the CrossGraphNet model. The first layer of mask is at the node (road segment) level: For the road segments that have ground sensor observation ground truth, we set their mask to True, meaning these segments will participate in the model training. For the road segments that do not have ground sensor observation ground truth, we set their mask to False, and these segments will not participate in the model training.

The second layer of the mask is at the entire spatio-temporal graph level: Since our spatio-temporal graph is not continuous (i.e. we do not consider the sequential features of the graph), but instead incorporate the features of each time step's spatio-temporal graph. For the spatio-temporal graphs to be inferred, we set their mask to False, meaning these graphs will not participate in the final model training. Through this double-layered mask mechanism, we can effectively handle missing data at both the node level and the graph level, improving the model's performance on discrete spatio-temporal graph data.

In mathematical terms, this double-layered mask structure can be formalized as follows: Given m time steps $\{t_1, t_2, \ldots, t_m\}$ with corresponding spatio-temporal graphs $\{G_{t_1}, G_{t_2}, \ldots, G_{t_m}\}$, each graph G_{t_i} represents the road network state at time step i. Due to the different spatio-temporal graphs having different missing states, we define a node-level validity mask $M^{node} \in \{\text{True, Fasle}\}^{m\times n}$. The node-level validity mask of G_{t_i} is $M^{node}_{t_i} \in \{\text{True, Fasle}\}^{1\times n}$:

$$M^{node}_{ij} = \begin{cases} \text{True}, & \text{if } (d_{ij} \neq 0) \wedge (d_{ij} \neq \text{null}) \wedge (s_{ij} \neq 0) \wedge (s_{ij} \neq \text{null}) \\ \text{Fasle}, & \text{otherwise} \end{cases} \tag{1}$$

where d_{ij} and s_{ij} denote the density and speed values of road node j in G_{t_i}, respectively. At the graph-level, we define the graph-level mask a graph-level validity mask $M^{graph} \in \{\text{True, Fasle}\}^{m\times 1}$:

$$M^{graph}_{t_i} = \begin{cases} \text{True}, & \text{Involved in training} \\ \text{Fasle}, & \text{otherwise} \end{cases} \tag{2}$$

where True indicates the spatio-temporal graph G_{t_i} is involved in training.

3.2. The model forward propagation computation process and cross-attention mechanism

In the model forward propagation, we only consider the spatio-temporal graphs with the train mask set to true, and perform two-layer graph convolution calculations for each spatio-temporal graph, followed by calculating the global mean pool value. When performing graph convolution calculations for each spatio-temporal graph, we only calculated the loss for the road segment nodes with real ground sensor data in the model's backpropagation. This is why we call it a double-layered mask structure.

For each individual time step, the graph convolutional network updates node features and integrates multi-spatial relationships (Zhang et al. 2022). The process can be described by the following set of equations:

$$Z^{(l)} = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(l)}W^{(l)} \tag{3}$$

$$H^{(l+1)} = \sigma(Z^{(l)}) \tag{4}$$

$$\tilde{A} = A + I \tag{5}$$

$$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij} \tag{6}$$

Here, $H^{(l)} \in \mathbb{R}^{N\times F_l}$ represents the matrix of node features at layer l, containing N nodes with F_l features each, while $W^{(l)} \in \mathbb{R}^{F_l\times F_{l+1}}$ denotes the trainable weight matrix for layer l. The adjacency matrix of the graph is given by $A \in \mathbb{R}^{N\times N}$, and $I \in \mathbb{R}^{N\times N}$ stands for the identity matrix. $\tilde{D} \in \mathbb{R}^{N\times N}$ represents the degree matrix of $\tilde{A}$, and $\sigma(\cdot)$ is a non-linear activation function, typically ReLU.

The normalization factor $\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$ is crucial for maintaining the scale of the features and preventing numerical instabilities during training. For a two-layer GCN, the full forward pass can be written as:

$$H^{(2)} = \sigma(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}\sigma(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(0)}W^{(0)})W^{(1)}) \tag{7}$$

In addition, we also want to address the non-continuous nature of the data. After completing the two-layer GCN computation, we can use the final node feature matrix $H^{(2)}$ as the initial embedding representation for this time step (that is, Embedding_T), which will be input into the subsequent calculation of the attention mechanism to achieve information fusion in multiple time steps (Cheng et al. 2017).

Assuming that we have 3 ($n = 3$) spatio-temporal graphs participating in the training, with the hidden layer embedding dimension as d, we will obtain a mixed embedding of size $3 \times d$ (Equation (8)). Since these three time steps are not continuous, we assume that they have the same impact on the target spatio-temporal graph (target spatio-temporal graph Mask set to False), and use a multi-head attention mechanism to sample this mixed embedding (Equation (9)), obtaining a new mixed embedding with attention weighting (as shown in Figure 3).

The computation formula of the mixed embedding is as follows:

$$\text{Mixed Embedding} = \left[\text{Embedding}T_1,\ \text{Embedding}T_2,\ \text{Embedding}_{T_3}\right] \tag{8}$$

$$\text{Attention Weighted Embedding} = \text{Cross Attention}(\text{Mixed Embedding}) \tag{9}$$

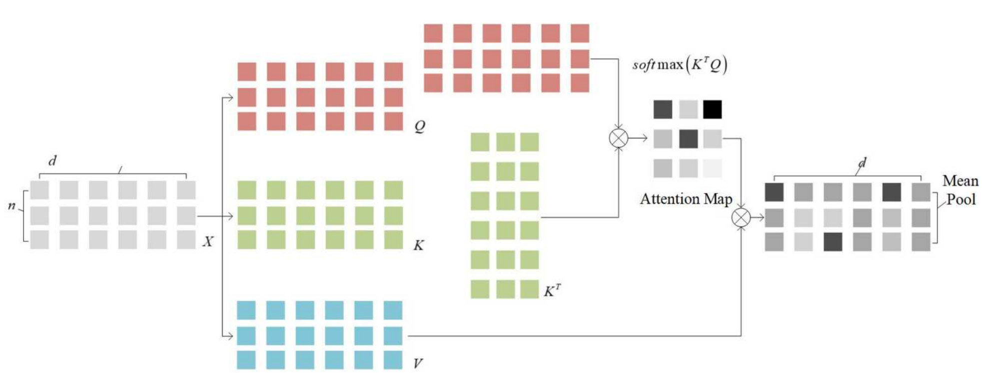

Figure 3. Illustration of the Spatio-Temporal Graph Feature Cross-Attention Mechanism (The input is the variable-length and non-contiguous spatio-temporal graph features, and the output is the mean pooling of the fuzed attention value embeddings. *n* represents the total number of spatio-temporal graph. *d* represents the dimension of the hidden layer in the graph convolution computation.)

The cross self-attention mechanism can be formulated as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{10}$$

Where Q, K, and V are all derived from the Mixed Embedding, d_k is the dimension of the key vectors, and the softmax operation is applied row-wise. In this work, d_k in the Figure 3 equals to the spatio-temporal graph node number n. Specifically:

$$\begin{aligned} Q &= \text{Mixed Embedding} \cdot W_Q \\ K &= \text{Mixed Embedding} \cdot W_K \\ V &= \text{Mixed Embedding} \cdot W_V \end{aligned} \tag{11}$$

Here, W_Q, W_K, and W_V are learnable weight matrices. This cross self-attention mechanism allows each time step to attend to all other time steps, enabling the model to capture complex temporal dependencies in the data and produce a weighted embedding that effectively integrates information across different time points.

Taking our training scenario with three spatio-temporal graphs ($n = 3$) as an example, the Mixed Embedding has a dimension of $3 \times d$. To comprehensively consider the impact of spatio-temporal graphs at different time steps on the target time step, we apply average pooling on the attention weighted embeddings to obtain the final embedding of size $1 \times d$ (Equation (12)):

$$\text{Final Embedding} = \text{Average Pool}(\text{Attention Weighted Embedding}) \tag{12}$$

This final embedding is then fed into the Multi-Layer Perceptron (MLP) layer to estimate the road node speed. Furthermore, we have provided the pseudocode for the model's forward propagation algorithm (Algorithm 1). In simple terms, we integrate the attention mechanism to sample the discrete spatio-temporal graph information and input the fuzed features to the MLP layer. The input and output dimensions of the MLP are identical, both equal to the number of nodes in the spatio-temporal graph.

3.3. The spatial smoothing module

To keep the spatial correlations between road segments after MLP layer, we introduce a Spatial Smoothing Module that applies adaptive Gaussian smoothing to the predictions. Let $P = (lat_1, lon_1), \ldots, (lat_n, lon_n)$ represent the coordinates of road segments, where $D \in \mathbb{R}^{n \times n}$ denotes the distance matrix, and D_{ij} represents the Euclidean distance between road segments i and j. The smoothing weights are calculated using a Gaussian kernel with learnable parameter σ:

$$S_{ij} = \frac{\exp\left(-D_{ij}^2/(2\sigma^2)\right)}{\sum_{k=1}^{n} \exp\left(-D_{ik}^2/(2\sigma^2)\right)} \tag{13}$$

For input features $X \in \mathbb{R}^{n \times d}$, the smoothed output $Y \in \mathbb{R}^{n \times d}$ is computed as:

$$Y = \alpha X + (1 - \alpha)(SX) \tag{14}$$

where $\alpha = \sigma(\sigma)$ ($\sigma(\cdot)$ is the sigmoid function) controls the smoothing intensity, and $S \in \mathbb{R}^{n \times n}$ is the smoothing weight matrix computed from Equation (13). This mechanism enables adaptive spatial smoothing while preserving local road segment characteristics.

Algorithm 1 CrossGraphNet Forward Pass with Spatial Smoothing

```
Require:
  data: List of graph data objects
  num features, hidden channels, num heads, num nodes
  node mask: Mask for valid nodes in each graph
  graph mask: Mask for selecting specific graphs
  coords: Road center coordinates (lat, lon)
  smooth sigma: Smoothing parameter
Ensure: Output estimations for selected graphs
  function SpatialSmoothing(x)
      weights ← exp(-EuclideanDistance(coords)^2 / (2 * sigma^2))
      weights ← Normalize(weights)
```

```
        smoothed ← MatrixMultiply(x, weights)
        return sigmoid(sigma) * x + (1 - sigmoid(sigma)) * smoothed
    end function
    function CrossAttention(num_features, hidden_channels, num_heads, num_nodes)
        Initialize GCN1(num_features → hidden_channels)
        Initialize GCN2(hidden_channels → num_nodes)
        Initialize LinearLayer(num_nodes → num_nodes)
        Initialize MultiheadAttention(num_nodes, num_heads)
    end function
    function Forward(data)
        embeddings ← []
        for each sub_data in data do do
            x, edge_index, batch ← sub_data
            x ← ReLU(GCN1(x, edge_index))
            x ← GCN2(x, edge_index)
            x ← GlobalMeanPool(x, batch)
            embeddings.append(x)
        end for
        embeddings ← Stack(embeddings)
        embeddings ← MultiheadAttention(embeddings, embeddings, embeddings)
        out ← Linear(Mean(embeddings, dim = 1))
        out ← SpatialSmoothing(out)
        return out
    end function
    function ApplyMasks(out, node_mask, graph_mask)
        out ← out[graph_mask]                                 ▷ Apply graph mask
        for each graph i in selected graphs do do
            out[i] ← out[i][node_mask[i]]                     ▷ Apply node mask
        end for
        return out
    end function
```

3.4. Bounded loss function and performance evaluation metrics

Moreover, in order to avoid extreme predictions in the model, and considering that ground station statistics indicate the majority of traffic speeds fall within the 30–110 km/h range, we constrain all of the model's predicted results to this interval. Then we modified the loss function and set additional penalty parameters. If the model's prediction value exceeds this range during the training process, a higher loss will be imposed to adjust the model's performance. The calculation formula of the loss function is as follows:

$$\text{Loss} = \begin{cases} \text{MSE}(y, \hat{y}) & \text{if } 30 \leq \hat{y} \leq 110 \\ \text{MSE}(y, \hat{y}) + \alpha \cdot (\max(0, \hat{y} - 110))^2 + \beta \cdot (\max(0, 30 - \hat{y}))^2 & \text{otherwise} \end{cases} \tag{15}$$

where y is the true value, $\hat{y}$ is the model's predicted value, and α and β are the pre-set penalty coefficients. When the predicted value exceeds the reasonable range of [30, 110] (reasonable Hong Kong road speed range, kilometer per hour), additional loss penalty will be applied.

To evaluate the model's performance, we used various metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Relative Squared Error (RSE).

MSE measures the average of the squared differences between the predicted values ($\hat{y}_i$) and the actual values (y_i), and is a widely used metric to evaluate the performance of regression models. MAE measures the average of the absolute differences between the predicted values and the actual values, and is less sensitive to outliers compared to MSE. RMSE is the square root of the MSE, and has the same unit as the target variable, which makes it easier to interpret the magnitude of the error. RSE is the ratio of the sum of squared errors (SSE) to the total sum of squares (TSS), which measures the proportion of the variance in the target variable that is not explained by the model, ranging from 0 to 1, with 0 indicating a perfect fit and 1 indicating that the model performance is no better than the mean of the target variable. These metrics provide a comprehensive evaluation of the model's performance in predicting road node travel speeds, with different perspectives on the magnitude and relative quality of the predictions.

The formulas for these metrics are as follows, used to evaluate the difference between the predicted road node travel speeds and the actual ground sensor recordings, i.e. the model's performance:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{16}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{17}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{18}$$

$$\mathrm{RSE} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{19}$$

where: n is the number of samples. $\bar{y}$ is the average of the true values. These metrics can comprehensively evaluate the model's performance in the road travel speed prediction task.

4. Result

The modeling framework of this study is presented in Figure 4. The proposed method comprises four main steps: remote sensing and ground sensor data processing, spatio-temporal graph modeling, cross-temporal spatial graph attention computation and training, and model application and prediction. To elaborate, the process proceeds as follows. Initially, vehicle detection is conducted on remote sensing images at four discontinuous time steps using the You Only Look Once (YOLO) model. The YOLOv8 model employed in this study underwent fine-tuning on a custom dataset annotated with vehicles detected in high-resolution satellite images from the Yuen Long District in Hong Kong. This fine-tuning ensures adaptation to local vehicle characteristics and road conditions, reducing the likelihood of misdetections, with a detection accuracy of not less than 80%. Subsequently, ground sensor data is collected at the time of remote sensing image capture, recording traffic flow speeds and setting training masks. Road segments with ground stations are labeled as True, while those without data are marked as False. A discrete spatio-temporal graph is then

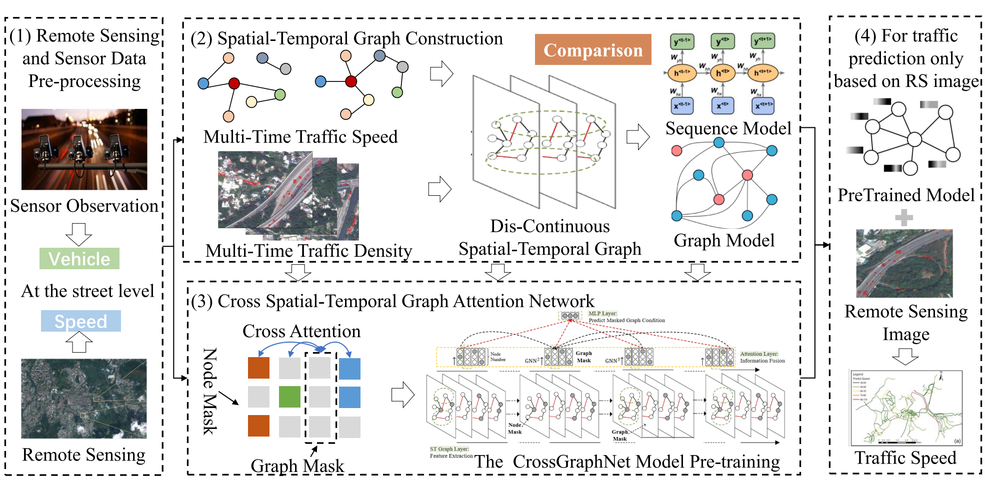

Figure 4. The modeling framework of this study. The proposed method consists of four main steps: remote sensing and ground sensor data processing; spatio-temporal graph modeling; cross-temporal spatial graph attention computation and training; model application and prediction. First, deep learning-based object detection are utilized to identify and locate vehicles within the satellite imagery. Next, the detected vehicles are spatially joint with their nearest road segments. Then, the vehicle-to-road mapping component enables the computation of vehicle density metrics for each road segment. Finally, the ST-Graph was built and pre-trained, which could predict the traffic conditions only relying on the remote sensing images.

constructed, and cross-attention mechanisms are utilized to capture correlations and complementary information between different graphs, integrating graph information from various time steps. The model engages in bidirectional learning on discontinuous data. Finally, upon completion of training, the model can estimate traffic speeds for all streets using only widely covered remote sensing images.

4.1. Remote sensing and ground sensor data processing

As shown in Figure 5, the study area is the Yuen Long District of Hong Kong Special Administrative Region, which is one of the urbanizing areas. The study area is equipped with 155 road sensors, of which 75 are responsible for detecting average road speeds, 75 are for monitoring road traffic volumes, and 5 are core stations that can detect speeds and count traffic volumes. In this paper, we primarily utilize the data from the 80 speed sensors, retrieving the speed data corresponding to the time of remote sensing image capture, which serves as training data for traffic modeling. It is worth noting that the 75 sensors do not always have available data at every timestamp, so custom masks need to be created for different spatio-temporal images to construct a complete training dataset.

As shown in Figure 6, we use two sensors as examples. At the time of remote sensing image acquisition on November 1, 2023, the road speeds monitored by these stations varied significantly. Sensor A recorded an average speed with a maximum of more than 100 km/h. In contrast, sensor B recorded lower average speeds. This difference reflects the distinct traffic conditions on roads of different functional classifications.

The vehicle detection dataset was collected using the Pleiades satellite, which captured images with a resolution of 0.5 m of Yuen Long, Hong Kong. This dataset comprises four large-scale raw images, each covering approximately 47 km^2 of the Yuen Long area, including major roads. The images were acquired on November 29, 2021; May 4, 2022; October 26, 2022; and November 1, 2023. After obtaining the satellite raw images, we utilized ENVI software to apply a series of pre-processing techniques, including radiometric calibration, atmospheric correction, ortho-rectification, and color correction. These steps were meticulously executed to ensure the images' quality and suitability for further analysis. Subsequently, the pre-processed images were divided into multiple tiles, each measuring 512×512 pixels. These tiles were then fed into the advanced YOLOv8 model from Ultralytics (https://github.com/ultralytics) for vehicle detection. As shown in Figure 7, the YOLOv8 model can accurately detect nearly all vehicles on the roads.

In the remote sensing images corresponding to these four time points, the YOLO model detected 5,480, 4,621, 3,282, and 3,499 vehicle targets, respectively.

Additionally, we need to process the road segments (Figure 8) and construct the topological road network within the study area. This includes editing the nodes and segments, ultimately retaining 593 major road segments.

Subsequently, we perform a spatial join of the vehicle targets detected by the YOLO model, assigning different vehicle densities to the road segments as their initial features. The calculation of the spatial

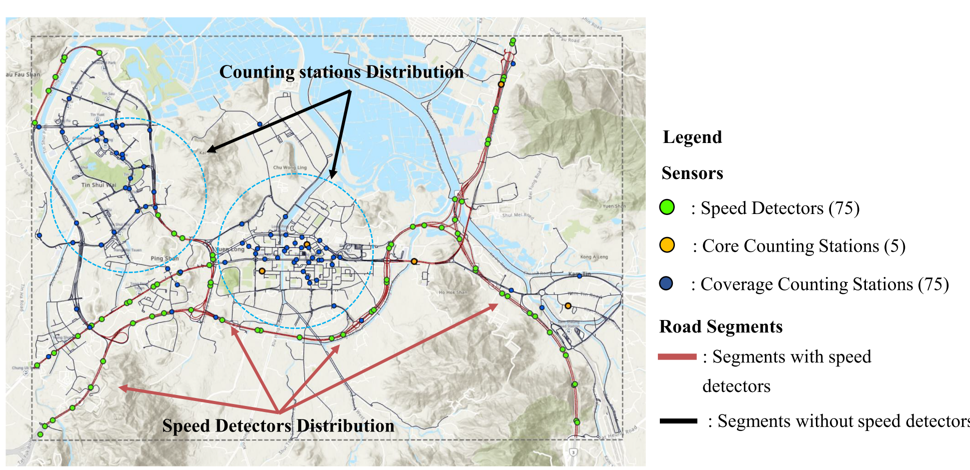

Figure 5. Illustration of the study area and ground sensor deployment.

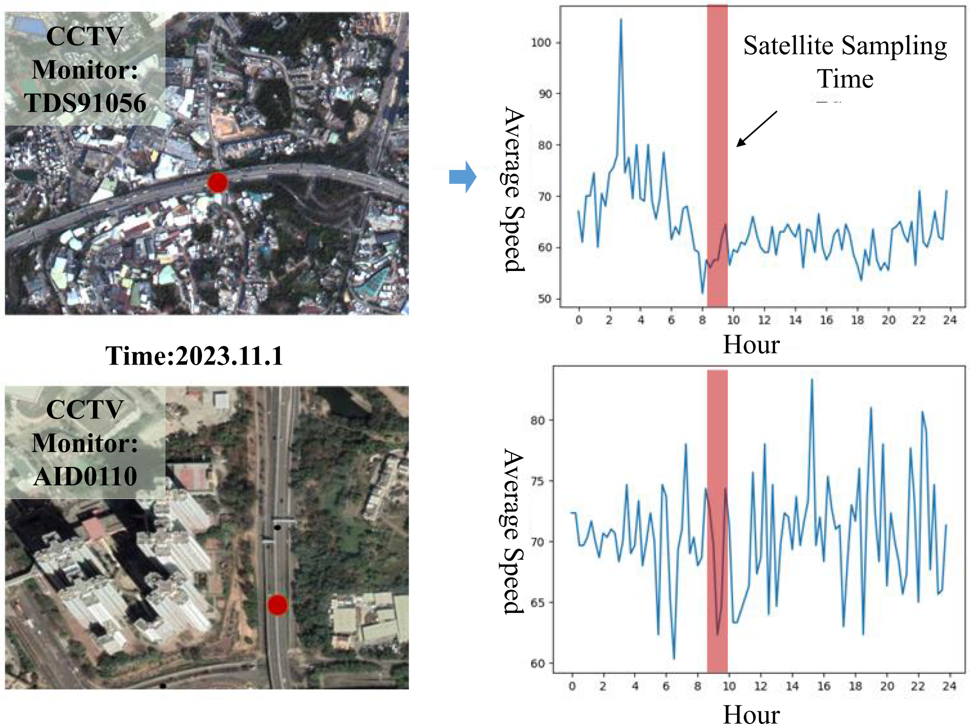

Figure 6. Road average speeds recorded by different ground sensor stations at the same moment.

adjacent matrix (Figure 9) between road segments is as follows: if there is a connection between segments i and j, then the values of $E\_(i, j)$ and $E\_(j, i)$ are set to 1, otherwise they are 0.

Subsequently, based on the results of the YOLO model's object detection, we calculated the traffic density of each road segment at different time points. As shown in Figure 10, we combined the remote sensing data with the ground sensor data. At time1, time2, and time3, there were approximately 38, 34, and 37 road segments, respectively, that had ground sensor data. The ground sensor data from time4 (November 1, 2023) will be used as the test set to compare the performance of different models.

Due to the sparse coverage of the ground sensors, on average, only about 6% of the road segments had actual measured ground data at each time point, which poses challenges for our modeling work.

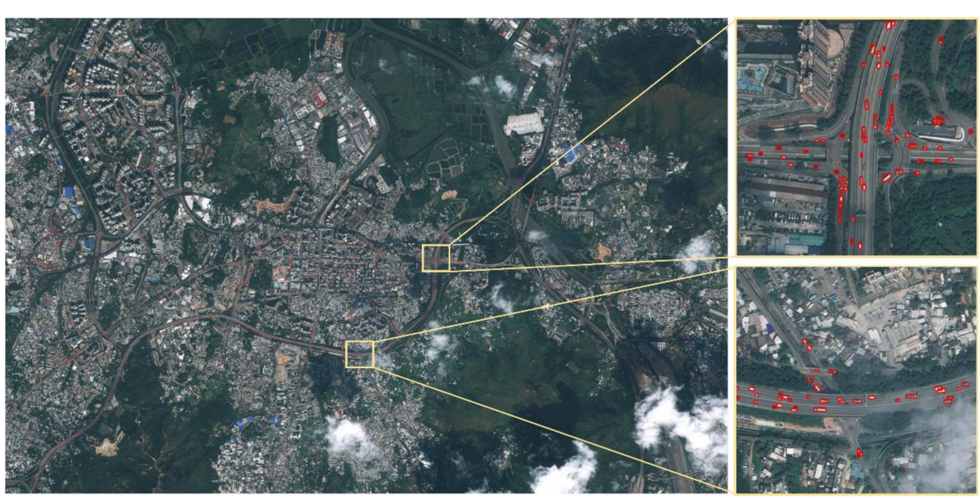

Figure 7. Vehicle Detection Visual Results (Based on the detected vehicle locations, we can match them to the nearest roads and compute the road density at this time).

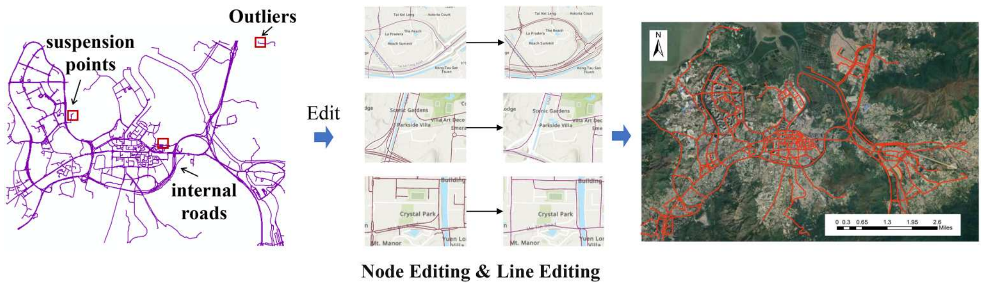

Figure 8. Editing and Pre-processing of the road network topology (Retaining 593 road segmentation).

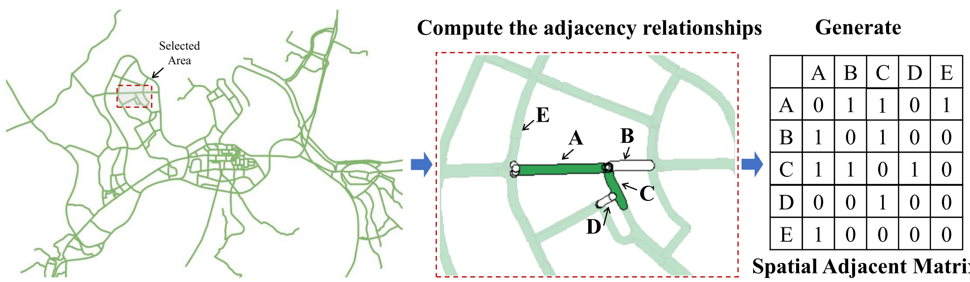

Figure 9. Constructing the spatial adjacent matrix based on the road network topology.

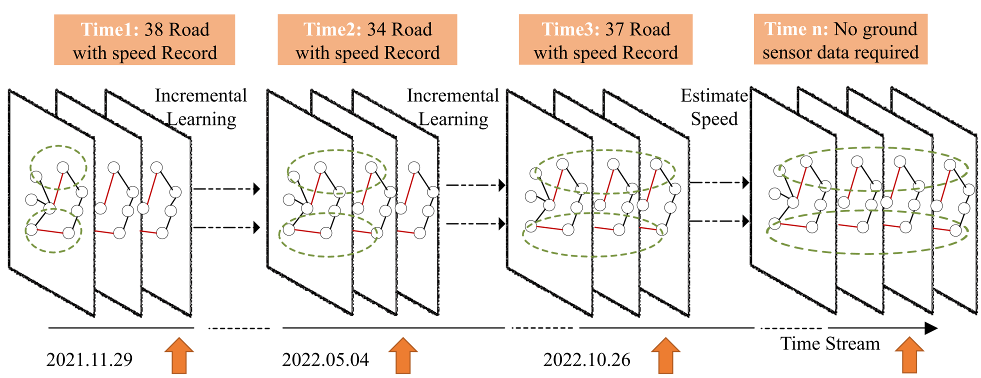

Figure 10. Ground sensor coverage at different time points.

4.2. Comparison of CrossGraphNet and other graph models

To thoroughly evaluate the effectiveness of our proposed method, we conducted two sets of comparative experiments (one focusing on pure graph-based methods with incremental learning, and another combining sequence models with graph-based approaches (Abdelraouf, Abdel-Aty, and Mahmoud 2022)) and one set of ablation experiment. For these three sets of experiments, we implemented all models using PyTorch and conducted training on identical hardware configurations to ensure fair comparison. The same hardware configuration: NVIDIA GeForce RTX 4090 GPU, Intel Core i9-13980HX CPU, and 32GB RAM. The experiments were conducted used a dataset comprising 4 remote sensing images and the corresponding ground sensor data. The first three time steps were used as the training set, while the last time step served as the test set. Each model was trained for a maximum of 200 epochs with early stopping based on validation performance to prevent overfitting. Performance evaluation was conducted using multiple metrics including MSE, MAE, RMSE, and RSE. These diverse metrics provide a comprehensive assessment of each model's predictive capabilities across different aspects of performance. The results of these experiments are summarized in Tables 1 and 2, respectively, showcasing the comparative performance of our proposed method against these established baselines.

Table 1. Performance comparison of graph neural network models with incremental learning.

Model	Test MSE	Test MAE	Test RMSE	Test RSE
AGG	233.373	12.178	15.277	0.978
ChebNet	266.308	13.107	16.319	1.117
GCN1	232.880	12.185	15.260	0.976
GCN2	232.705	12.234	15.255	0.976
GAT	545.799	18.455	23.362	2.288
GraphSage	233.398	12.246	15.277	0.979
Average	290.744	13.401	16.792	1.219
	± 51.299	± 1.021	± 1.325	± 0.215
MaskSTGraph1	56.781	5.724	7.534	0.308
MaskSTGraph2	27.719	3.981	5.265	**0.133**
MaskSTGraph3	**27.367**	**3.927**	**5.231**	0.138
MaskSTGraph4	40.117	4.768	6.334	0.228

Note: The last four experiments were based on our CrossGraphNet model. MaskSTGraph means that the data from a certain time point is masked (i.e. unavailable), and data from other moments is required to reconstruct and estimate its traffic state. The difference between GCN1 and GCN2 is that they have a different number of hidden layers.

Table 2. Performance comparison of sequence models with graph-based approaches.

Model	Test MSE	Test MAE	Test RMSE	Test RSE
ST-RNN	265.460	11.956	16.293	1.146
ST-LSTM	263.799	12.119	16.242	1.139
CTLE	264.679	11.940	16.269	1.142
BiLSTM-CNN	252.766	12.238	15.899	1.091
LSTM-GCN	262.973	12.143	16.216	1.135
Transformer	285.125	13.704	16.886	1.231
Attention LSTM	250.429	12.712	15.825	1.081
Average	263.604	12.402	16.233	1.138
(exclude LSTM)	± 4.252	± 0.238	± 0.130	± 0.018
LSTM	1689.841	38.153	41.108	7.293
MaskSTGraph4	**40.117**	**4.768**	**6.334**	**0.228**

Note: The last experiment was based on our CrossGraphNet model. The first three times data was for training, and the fourth step data was for testing, which is the same with sequence model.

Comparative experiment set 1: graph neural network models

In the first set of experiments, we compared our method against six efficient graph neural network models:

(1) *Chebyshev Graph Convolution Network (ChebNet)* (Sahbi 2021): Utilizes Chebyshev polynomials to approximate graph convolutions, enabling efficient spectral filtering on graphs.
(2) *Graph Convolutional Network1 (GCN1)* (Zhang et al. 2022): Employs a first-order approximation of spectral graph convolutions for semi-supervised learning on graph-structured data.
(3) *Graph Convolutional Network 2 (GCN2)*: An enhanced version of GCN1, typically with deeper architecture or additional features to improve performance.
(4) *Graph Attention Network (GAT)* (Veličković et al. 2018): Introduces attention mechanisms to graph convolutions, allowing nodes to attend differently to their neighbors.
(5) *GraphSage* (Hamilton, Ying, and Leskovec 2017): Uses a node embedding approach that leverages node feature information to efficiently generate embeddings for previously unseen data.
(6) *Attention Graph Neural Network (AGG)*: Focuses on learning graph representations while preserving privacy by anonymizing node identities.

These models were selected based on their proven capabilities in processing graph-structured data and their suitability for incremental learning. The dataset for this experiment consisted of four remote sensing images with corresponding ground sensor data. We adopted an incremental learning approach, whereby the graph models were trained sequentially based on data from three time steps. The trained models could then predict the road network speed conditions at the final time step based on the road vehicle density attributes (detected from the remote sensing images), without requiring actual ground sensor observations.

Figure 11 demonstrates that over time, the loss of most models exhibited a decreasing trend, indicating that the incremental learning strategy effectively enabled the models to gradually optimize and improve

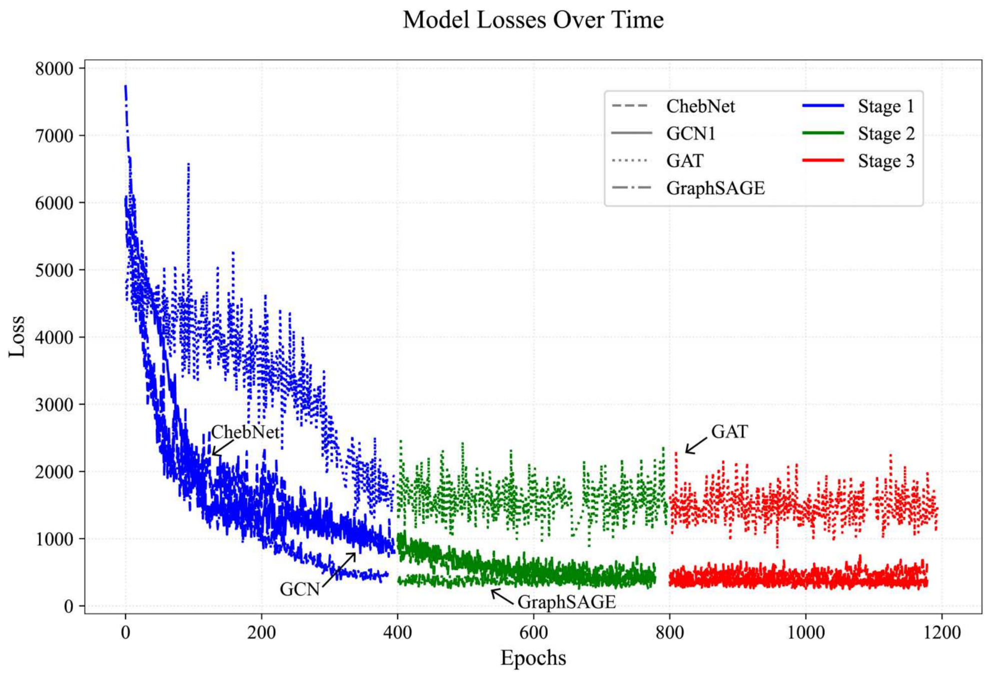

Figure 11. Loss values of different graph models during incremental learning.

their road network speed prediction capabilities. Different models exhibited varying performance during the incremental learning process. For example, GraphSAGE demonstrated higher loss initially but achieved superior performance subsequently, suggesting enhanced learning adaptability.

As mentioned in the Method section, the proposed CrossGraphNet supports bidirectional information flow. For these four spatio-temporal graphs, we sequentially apply masking to each one, and then predict the masked spatio-temporal graph based on the data from the other time points. The evaluation results of the models are shown in Table 1.

Comparative experiment set 2: sequence models with graph-based approaches

In this set of experiments, we evaluated the sequence models with various encoders, all of which used GCN for decoding. We expanded our comparison to include models that combine sequence processing with graph-based methods:

(1) *ST-RNN (Spatio-Temporal RNN)* (Yu, Yin, and Zhu 2018): Extends traditional RNNs to capture both spatial and temporal dependencies in sequence data.
(2) *ST-LSTM (Spatio-Temporal LSTM)* (X. Shi et al. 2015): Enhances LSTM architecture with convolutional structures to model spatial-temporal relationships.
(3) *CTLE (Convolutional-LSTM with Edge)* (Seo et al. 2018): Combines Convolutional LSTM with edge information to better capture spatial dependencies in graph-structured data.
(4) *BiLSTM-CNN* (M. Shi, Wang, and Li 2019): Integrates bidirectional LSTM with CNN to leverage both sequential and local spatial features.
(5) *LSTMGCN* (X. Wang et al. 2023): Combines LSTM for temporal modeling with Graph Convolutional Networks to capture both temporal and spatial relationships.
(6) *Transformer Model* (Vaswani et al. 2017): Utilizes self-attention mechanisms to process sequential data, allowing for parallel computation and capturing long-range dependencies.
(7) *LSTM* (Gers and Schmidhuber 2001): A recurrent neural network architecture designed to address the vanishing gradient problem, capable of learning long-term dependencies.
(8) *Attention LSTM* (Bahdanau, Cho, and Bengio 2015): Augments LSTM with an attention mechanism, enabling the model to focus on different parts of the input sequence when producing output.

These models were selected for their ability to capture both temporal dependencies and spatial relationships. The same dataset and train-test split were used as in the first experiment set.

Table 3. Ablation experiment on different model module.

Model	Test MSE	Test MAE	Test RMSE	Test RSE
Complete model (Proposed method)				
MaskSTGraph1	56.781	5.724	7.534	0.308
MaskSTGraph2	27.719	3.981	5.265	**0.133**
MaskSTGraph3	**27.367**	**3.927**	**5.231**	0.138
MaskSTGraph4	40.117	4.768	6.334	0.228
Without Algorithm 1 spatial smoothing function				
MaskSTGraph1	62.329	5.806	7.895	0.373
MaskSTGraph2	32.640	4.312	5.713	0.170
MaskSTGraph3	**22.186**	**3.565**	**4.710**	**0.128**
MaskSTGraph4	41.052	4.992	6.407	0.267
Without Algorithm 1 CrossAttention function				
MaskSTGraph1	132.923	7.997	11.529	0.545
MaskSTGraph2	**41.121**	**4.740**	**6.413**	**0.188**
MaskSTGraph3	93.694	6.819	9.680	0.381
MaskSTGraph4	76.898	5.714	8.769	0.445

Note: This ablation experiment compares three variants of our model: (1) the complete proposed model with all components, (2) the model without spatial smoothing, and (3) the model without both cross-attention and spatial smoothing. Results show that both components contribute significantly to model performance.

Analysis of Tables 1 and 2 reveals that Sequence Models with Graph outperform Graph Neural Network Models with Incremental Learning across most metrics, including MSE, MAE, and RMSE. The exception to this trend is observed in the RSE metric. Among the comparative experiments, the standard LSTM demonstrated the poorest performance in processing sequential data. In contrast, Attention LSTM, GCN, AGG, and CTLE exhibited marginally superior results.

And we can see that our proposed method achieves the best overall accuracy. Regardless of the time point, the MAE is less than 6, which means the average difference between the predicted and true road speeds is within 6 km/h. Our bi-directional spatio-temporal graph model can make predictions for the traffic conditions at any given time point. Notably, when masking the third spatio-temporal graph, the model leveraging information from the previous and subsequent three time points achieves the best performance, with an MAE of 3.927 (indicates an average prediction error of only 3.927 km/h). Even when compared to the MaskSTGraph4 (predict the time4 road condition) baseline under equivalent conditions, our model still outperforms the others. This further validates the superiority of our model in capturing dynamic traffic condition variations, thereby enabling more accurate prediction of network states at any given temporal stage.

Ablation experiment set 1: ablation experiments of CrossGraphNet with different modules

Table 3 presents the ablation experiments of different modules of CrossGraphNet, including the ablation experiment without considering the Spatial Smoothing Function in Algorithm 1 (Section 3.2) and the ablation experiment without considering the CrossAttention Function in Algorithm 1 (Section 3.3). The results show that our method is relatively robust, and both key functions significantly improve the model's performance. Taking the MAE metric as an example, in the control group of MaskSTGraph4, the MAE decreased from 5.714 and 4.992 to 4.768, resulting in an accuracy improvement of approximately 16.5% and 4.5%, respectively.

In summary, our method achieved the test MSE of 40.117, which indicates the model has the minimal deviation from the true target values. The test MAE of 4.768 further corroborates the model's reliability, suggesting an average prediction error of less than 5 units (km/h). The low RMSE of 6.334 and RSE of 0.228 also underscore the model's superior performance compared to the baseline approaches.

In summary, the proposed bidirectional CrossGraphNet demonstrates excellent performance in the traffic speed prediction task. It not only achieves higher predictive accuracy than existing methods but also exhibits its ability in the non-continuous and missing observation scenarios.

4.3. Prediction results and interpretability of CrossGraphNet computations

Based on the prediction results of the MaskSTGraph1, MaskSTGraph2, MaskSTGraph3, and MaskSTGraph4, we have plotted the model predictions at these four time points (Figure 12). The results show

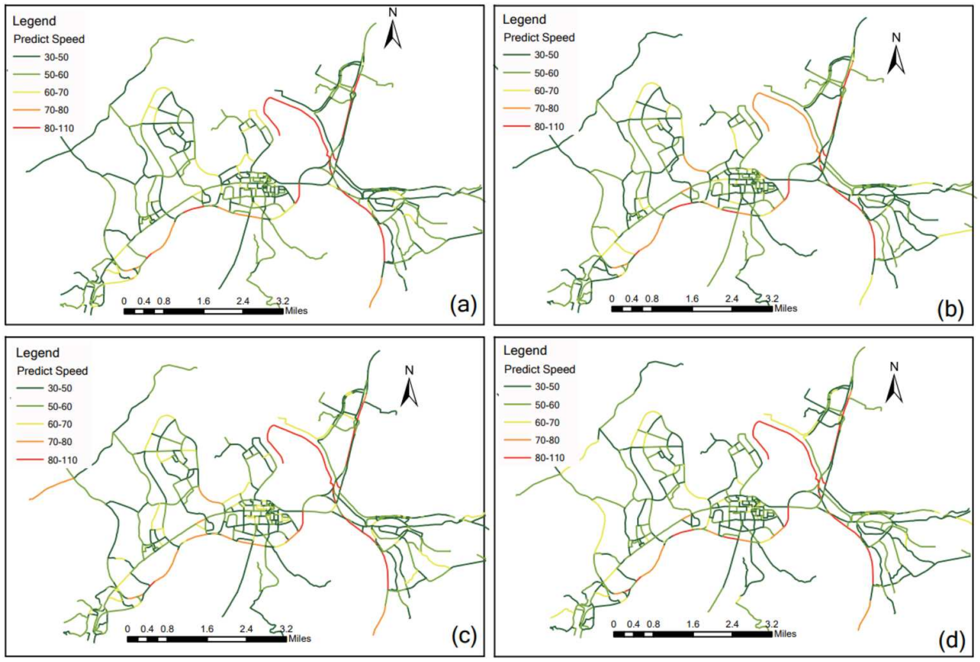

Figure 12. Traffic Speed Reconstruction Results of CrossGraphNet with Different Time Step Masked (MaskSTGraph1-4). (a), (b), (c), and (d) represent the model's prediction results when the data from time points 1, 2, 3, and 4 are masked (i.e. made unavailable), respectively.

that our bidirectional spatio-temporal graph neural network (CrossGraphNet) can effectively capture the bidirectional spatio-temporal information, leading to more stable and accurate predictions. The model also can be readily extended to quickly reconstruct traffic speeds when new remote sensing images become available.

This figure also demonstrates the performance of the CrossGraphNet model under different scenarios where the data from specific time points is unavailable. The results show the model's ability to leverage the remaining spatio-temporal information to make accurate traffic speed predictions, even when facing partial data availability.

To further elucidate the inner workings of our model, we analyze the attention matrix employed in CrossGraphNet. Figure 13 presents a visualization of attention heatmaps (presented in Figure 3) across four sequential time steps.

The attention maps demonstrate the model's capability to effectively synthesize information from different time steps. When predicting the first time step (MaskSTGraph1), information from the initial two time steps exerts a stronger influence on the model's output, as evidenced by higher attention weights (exceeding 0.3). Conversely, for predictions of the latter two time steps (MaskSTGraph3,4), information from these later steps shows a greater impact. When predicting the second time step (MaskSTGraph2), information from all four time steps exhibits comparable influence weights on the model's output.

This visualization highlights a key feature of our approach in handling non-continuous information: it could process all available data uniformly, allocating higher 'attention' to time steps that most significantly impact the target step, based on the specific context. This method overcomes the vanishing gradient issues inherent in unidirectional sequence models and incremental graph models during sequential computation (Kerg et al. 2020).

Furthermore, our approach demonstrates the ability to filter information, addressing a common bias in traditional bidirectional models where time steps adjacent to the target step often exert disproportionate influence. Instead, our model could achieve a more balanced acquisition of relevant information across all time steps (MaskSTGraph2). In total, this enhanced ability to selectively process and integrate temporal information allows CrossGraphNet to capture complex spatio-temporal dependencies more effectively.

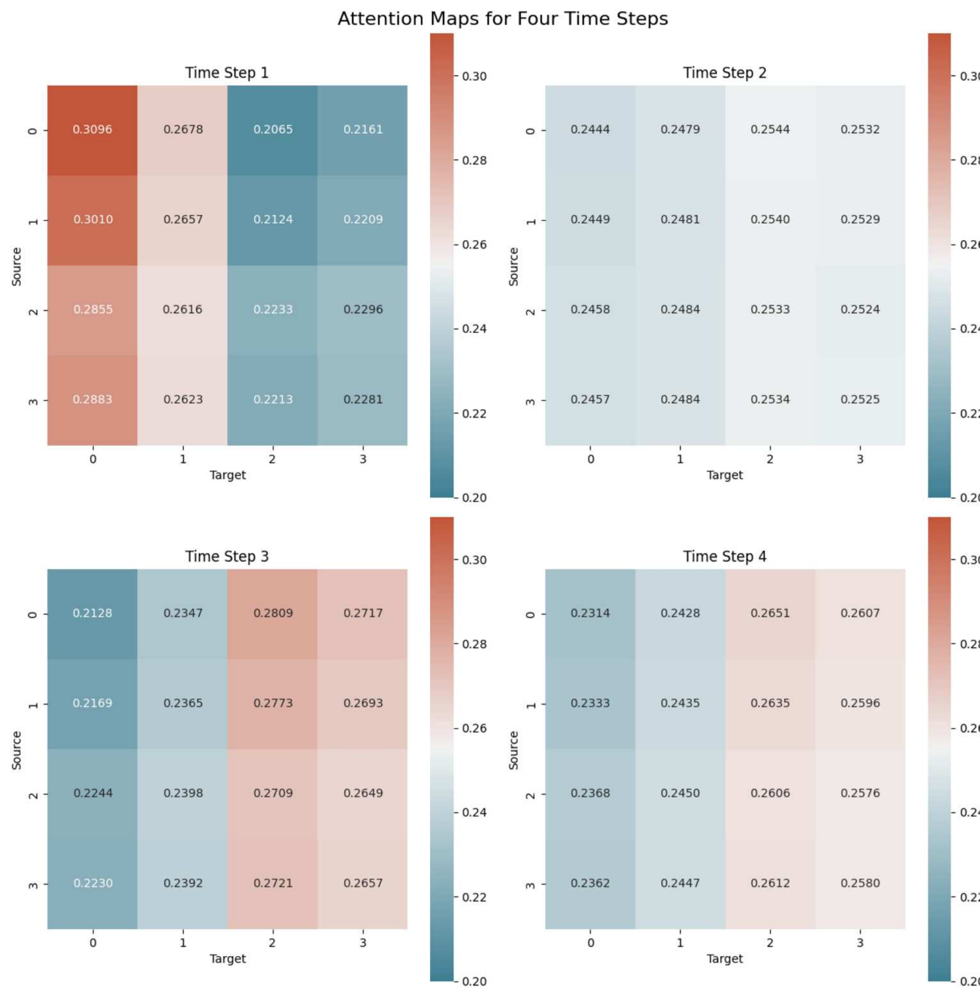

Figure 13. Visualization of attention maps across four sequential time steps (MaskSTGraph1-4). Each subplot represents the attention distribution at a specific time step, with rows indicating source spatial-temporal graph and columns representing target spatial-temporal graph. Color intensity and numerical values in each grid denote the strength of attention, ranging from 0.2 (cooler colors) to 0.31 (warmer colors). The evolving patterns across time steps illustrate the dynamic nature of the attention mechanism in our model.

5. Discussion and conclusion

In this work, we proposed a novel bidirectional spatio-temporal graph neural network, named CrossGraphNet, for accurate and robust traffic speed reconstruction based on remote sensing images. Our model is able to effectively capture the complex spatio-temporal dependencies in traffic data, outperforming several state-of-the-art baselines across various evaluation metrics (MSE=40.117, MAE=4.768, RMSE=6.334, RSE=0.228 in the testing dataset). Notably, the model's exceptional performance when masking data from specific time steps demonstrates its strong ability to leverage the available spatio-temporal context to make accurate predictions, even in the presence of missing sensor observations.

The superior results of CrossGraphNet highlight its potential for practical deployment in real-world traffic management applications. It can seamlessly interpret remote sensing data. Further, the model's robustness to incomplete data makes it well-suited for handling the challenges inherent in traffic prediction, such as sensor failures or irregular data availability. This traffic conditions estimating method can be effectively applied in the field of disaster management, especially when ground station data is missing due to various emergencies.

While CrossGraphNet demonstrates robust performance within the study area, its generalization capability is limited by the static graph assumption. The proposed method is designed for urban traffic reconstruction in specific regions. When applied to new areas with different road network configurations or traffic patterns, the model requires retraining, including recalculating the spatial adjacency matrix and adapting node features. While the model demonstrates strong performance in this urban context, its generalizability to other regions with different road networks, urban layouts, or traffic characteristics has not been explicitly tested. The model assumes a fixed road network topology, which may not hold in regions with evolving road layouts or newly constructed streets. In such cases, the adjacency matrix and node representations must be recalculated, leading to significant computational overhead and potential loss of historical information.

In addition, cloud cover and orbital constraints may lead to data gaps, further limiting temporal coverage. This restricts the analysis to fixed-time traffic characteristics and prevents the capture of intraday variations (e.g. morning/evening peaks). While our current model demonstrates strong performance using vehicle density and sparse ground sensor data, future work could incorporate supplementary data sources to further enhance model performance. Meteorological conditions such as precipitation, temperature, and visibility significantly affect traffic speeds, while land use and land cover information also serve as important supplements (H. Chen et al. 2022; S. Zhao et al. 2024). Roads near schools, shopping centers, or industrial areas exhibit unique temporal speed patterns that differ from general traffic flow.

Overall, the scalability of CrossGraphNet to longer time series and varied revisit cadences is supported by the cross-spatiotemporal attention mechanism, which is designed to integrate information from arbitrarily spaced time points. Future work will focus on testing the model's performance with longer and more continuous time series to further validate its adaptability and robustness.

CRediT Author statement

Yan Zhang: conceptualization, methodology, formal analysis, writing – original draft, and writing – review & editing; Jiannan Cai: writing – review & editing; Mei-Po Kwan: supervision, funding acquisition, conceptualization, methodology, resources, and writing – review & editing. Peifeng Ma and Jianying Wang: conceptualization, methodology. All authors have read and agreed to the final manuscript.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This research was supported by a Smart Traffic Fund (PSRI/44/2208/PR).

ORCID

Yan Zhang http://orcid.org/0000-0002-2059-4171
Mei-Po Kwan http://orcid.org/0000-0001-8602-9258

References

Abdelraouf, A., M. Abdel-Aty, and N. Mahmoud. 2022. "Sequence-to-sequence Recurrent Graph Convolutional Networks for Traffic Estimation and Prediction Using Connected Probe Vehicle Data." *IEEE Transactions on Intelligent Transportation Systems* 24 (1): 1395–1405. https://doi.org/10.1109/TITS.2022.3168865.

Abduljabbar, R. L., H. Dia, P.-W. Tsai, and J. Tang. 2021. "Unidirectional and Bidirectional LSTM Models for short-term Traffic Prediction." *Journal of Advanced Transportation* 2021:5589075. https://doi.org/10.1155/2021/5589075.

Asensio, O. I., K. Alvarez, A. Dror, E. Wenzel, C. Hollauer, and S. Ha. 2020. "Real-Time Data from Mobile Platforms to Evaluate Sustainable Transportation Infrastructure." *Nature Sustainability* 3:463–471. https://doi.org/10.1038/s41893-020-0533-6.

Bahdanau, D., K. Cho, and Y. Bengio. 2015. "Neural Machine Translation by Jointly Learning to Align and Translate." 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 7–9.

Balid, W., H. Tafish, and H. Refai. 2017. "Intelligent Vehicle counting and Classification Sensor for Real-Time Traffic Surveillance." *IEEE Transactions on Intelligent Transportation Systems*19:1784–1794. https://doi.org/10.1109/TITS.2017.2741507.

Bouguettaya, A., H. Zarzour, A. Kechida, and A. M. Taberkit. 2021. "Vehicle Detection from Uav Imagery with Deep Learning: A Review." *IEEE Transactions on Neural Networks and Learning Systems*33:6047–6067. https://doi.org/10.1109/TNNLS.2021.3080276.

Boukerche, A., Y. Tao, and P. Sun. 2020. "Artificial Intelligence-Based Vehicular Traffic Flow Prediction Methods for Supporting Intelligent Transportation Systems." *Computer Networks* 182:107484. https://doi.org/10.1016/j.comnet.2020.107484.

Cao, M., H. Ji, Z. Gao, and T. Mei. 2020. "Vehicle Detection in Remote Sensing Images Using Deep Neural Networks and Multi-task Learning." *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 2:797–804. https://doi.org/10.5194/isprs-annals-V-2-2020-797-2020.

Chen, Y., R. Qin, G. Zhang, and H. Albanwan. 2021. "Spatial Temporal Analysis of Traffic Patterns during the Covid-19 Epidemic by Vehicle Detection Using Planet Remote-Sensing Satellite Images." *Remote Sensing* 13:208. https://doi.org/10.3390/rs13020208.

Chen, H., N. Yokoya, C. Wu, and B. Du. 2022. "Unsupervised Multimodal Change Detection Based on Structural Relationship Graph Representation Learning." *IEEE Transactions on Geoscience and Remote Sensing* 60:1–18.

Chen, Z., T. Zou, Z. Xu, Y. Zhang, and N. Chen. 2024. "Sage-Gsan: A Graph-Based Method for Estimating Urban Taxi Co Emissions Using Street View Images." *Journal of Cleaner Production*474:143543. https://doi.org/10.1016/j.jclepro.2024.143543.

Cheng, Q., H. Liu, H. Shen, P. Wu, and L. Zhang. 2017. "A Spatial and Temporal Nonlocal Filter-Based Data Fusion Method." *IEEE Transactions on Geoscience and Remote Sensing* 55:4476–4488. https://doi.org/10.1109/TGRS.2017.2692802.

Cui, Z., K. Henrickson, R. Ke, and Y. Wang. 2019. "Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting." *IEEE Transactions on Intelligent Transportation Systems* 21:4883–4894. https://doi.org/10.1109/TITS.6979.

Deng, J., Y. Cui, X. Chen, C. Bachmann, and Q. Yuan. 2023. "Who Are on the Road? A Study on Vehicle Usage Characteristics Based on One-week Vehicle Trajectory Data." *International Journal of Digital Earth* 16 (1): 1962–1984. https://doi.org/10.1080/17538947.2023.2218117.

Gao, Z., H. Zeng, X. Zhang, H. Wu, R. Zhang, Y. Sun, Q. Du, et al. 2024. "Exploring Tourist Spatiotemporal Behavior Differences and Tourism Infrastructure Supply–demand Pattern Fusing Social Media and Nighttime Light Remote Sensing Data." *International Journal of Digital Earth* 17 (1): 2310723. https://doi.org/10.1080/17538947.2024.2310723.

Gers, F. A., and E. Schmidhuber. 2001. "LSTM Recurrent Networks Learn Simple Context-Free and Context-Sensitive Languages." *IEEE Transactions on Neural Networks* 12 (6): 1333–1340. https://doi.org/10.1109/72.963769.

Guo, S., Y. Lin, N. Feng, C. Song, and H. Wan. 2019. "Attention based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting." In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, Honolulu, Hawaii, USA. AAAI Press. https://doi.org/10.1609/aaai.v33i01.3301922.

Guo, Y., C. Wu, B. Du, and L. Zhang. 2022. "Density Map-Based Vehicle counting in Remote Sensing Images with Limited Resolution." *ISPRS Journal of Photogrammetry and Remote Sensing* 189:201–217. https://doi.org/10.1016/j.isprsjprs.2022.05.004.

Hamilton, W. L., R. Ying, and J. Leskovec. 2017. "Inductive Representation Learning on Large Graphs." In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 1025–1035. Red Hook, NY: Curran Associates Inc.

He, H., S. Chen, and K. Li. 2011. "Incremental Learning from Stream Data." *IEEE Transactions on Neural Networks* 22:1901–1914. https://doi.org/10.1109/TNN.2011.2169087.

Huo, G., Y. Zhang, B. Wang, J. Gao, Y. Hu, and B. Yin. 2023. "Hierarchical Spatio–temporal Graph Convolutional Networks and Transformer Network for Traffic Flow Forecasting." *IEEE Transactions on Intelligent Transportation Systems* 24:3855–3867. https://doi.org/10.1109/TITS.2023.3234512.

Jia, X., X. Dong, M. Chen, and X. Yu. 2021. "Missing Data Imputation for Traffic Congestion Data Based on Joint Matrix Factorization." *Knowledge-Based Systems* 225:107114. https://doi.org/10.1016/j.knosys.2021.107114.

Kan, Z., L. Tang, M.-P. Kwan, and X. Zhang. 2018. "Estimating Vehicle Fuel Consumption and Emissions Using Gps Big Data." *International Journal of Environmental Research and Public Health* 15:566. https://doi.org/10.3390/ijerph15040566.

Kashinath, S. A., S. A. Mostafa, A. Mustapha, H. Mahdin, D. Lim, M. A. Mahmoud, M. A. Mohammed, B. A. S. Al-Rimy, M. F. M. Fudzee, and T. J. Yang. 2021. "Review of Data Fusion Methods for Real-Time and Multi-sensor Traffic Flow Analysis." *IEEE Access* 9:51258–51276. https://doi.org/10.1109/ACCESS.2021.3069770.

Kerg, G., B. Kanuparthi, A. Alias Parth Goyal, K. Goyette, Y. Bengio, and G. Lajoie. 2020. "Untangling Tradeoffs between Recurrence and Self-attention in Artificial Neural Networks." *Advances in Neural Information Processing Systems* 33:19443–19454.

Leitloff, J., D. Rosenbaum, F. Kurz, O. Meynberg, and P. Reinartz. 2014. "An Operational System for Estimating Road Traffic Information from Aerial Images." *Remote Sensing* 6:11315–11341. https://doi.org/10.3390/rs61111315.

Li, J., and B. Chen. 2020. “Global Revisit Interval Analysis of Landsat-8-9 and Sentinel-2a-2b Data for Terrestrial Monitoring.” *Sensors* 20:6631. https://doi.org/10.3390/s20226631.

Li, Y., Y. Zhang, and M. Chen. 2022. “Highway Vehicle Route Reconstruction Using Sparse and Noisy Communication Base Station Fingerprints.” *IEEE Sensors Journal* 22:22040–22052. https://doi.org/10.1109/JSEN.2022.3212400.

Li, M., and Z. Zhu. 2021. “Spatial-Temporal Fusion Graph Neural Networks for Traffic Flow Forecasting.” *Proceedings of the AAAI Conference on Artificial Intelligence* 35 (5): 4189–4196. https://doi.org/10.1609/aaai.v35i5.16542.

Lin, J., Y. Zhao, S. Wang, and Y. Tang. 2023. “Yolo-Da: An Efficient Yolo-Based Detector for Remote Sensing Object Detection.” *IEEE Geoscience and Remote Sensing Letters* 20:1–5.

Lv, M., Z. Hong, L. Chen, T. Chen, T. Zhu, and S. Ji. 2020. “Temporal Multi-graph Convolutional Network for Traffic Flow Prediction.” *IEEE Transactions on Intelligent Transportation Systems*22:3337–3348. https://doi.org/10.1109/TITS.2020.2983763.

Ma, L., Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson. 2019. “Deep Learning in Remote Sensing Applications: A Meta-analysis and Review.” *ISPRS Journal of Photogrammetry and Remote Sensing*152:166–177. https://doi.org/10.1016/j.isprsjprs.2019.04.015.

Macioszek, E., and A. Kurek. 2021. “Extracting Road Traffic Volume in the City before and during Covid-19 through Video Remote Sensing.” *Remote Sensing* 13:2329. https://doi.org/10.3390/rs13122329.

Min, R., Y. Chen, H. Wang, and Y. Chen. 2024. “Das Vehicle Signal Extraction Using Machine Learning in Urban Traffic Monitoring.” *IEEE Transactions on Geoscience and Remote Sensing* 40:1–14.

Ming, Q., L. Miao, Z. Zhou, and Y. Dong. 2021. “Cfc-Net: A Critical Feature Capturing Network for Arbitrary-Oriented Object Detection in Remote-Sensing Images.” *IEEE Transactions on Geoscience and Remote Sensing* 60:1–14. https://doi.org/10.1109/TGRS.2021.3095186.

Sahbi, H. 2021. “Learning Laplacians in Chebyshev Graph Convolutional Networks.” In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Montreal, BC, Canada, 2064–2075. IEEE.

Scheibenreif, L., M. Mommert, and D. Borth. 2022. “Toward Global Estimation of Ground-Level No 2 Pollution with Deep Learning and Remote Sensing.” *IEEE Transactions on Geoscience and Remote Sensing* 60:1–14. https://doi.org/10.1109/TGRS.2022.3160827.

Seo, Y., M. Defferrard, P. Vandergheynst, and X. Bresson. 2018. “Structured Sequence Modeling with Graph Convolutional Recurrent Networks.” In *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part I*, Siem Reap, Cambodia, 362–373. Berlin, Heidelberg: Springer-Verlag. https://doi.org/10.1007/978-3-030-04167-0_33.

Shi, X., Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. 2015. “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting.” *Advances in Neural Information Processing Systems* 28:1.

Shi, Y., H. Tao, C. Deng, L. Zhuo, Y. Ma, and Q. Shi. 2024. “FBS-TGCN: A Temporal Graph-convolutional-Network Model for Spatiotemporal Prediction of Spam Messages from Fake Base Stations.” *International Journal of Geographical Information Science* 38 (4): 751–777. https://doi.org/10.1080/13658816.2024.2310118.

Shi, M., K. Wang, and C. Li. 2019. “A C-LSTM with Word Embedding Model for News Text Classification.” In *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)*, 253–257. IEEE. https://doi.org/10.1109/ICIS46139.2019.8940289.

Stokes, J. M., K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. M. Donghia, C. R. MacNair, S. French, L. A. Carfrae, and Z. Bloom-Ackermann. 2020. “A Deep Learning Approach to Antibiotic Discovery.” *Cell* 180:688–702. https://doi.org/10.1016/j.cell.2020.01.021.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. 2017. “Attention Is All You Need.” In *Advances in Neural Information Processing Systems*. Vol. 30, edited by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. 2018. “Graph Attention Networks.” In *International Conference on Learning Representations*. ICLR. https://openreview.net/forum?id=rJXMpikCZ.

Wang, S., J. Cao, and S. Y. Philip. 2020. “Deep Learning for Spatio-Temporal Data Mining: A Survey.” *IEEE Transactions on Knowledge and Data Engineering* 34:3681–3700. https://doi.org/10.1109/TKDE.2020.3025580.

Wang, T., Z. Shen, Y. Cao, X. Xu, and H. Gong. 2021. “Taxi-Cruising Recommendation via Real-Time Information and Historical Trajectory Data.” *IEEE Transactions on Intelligent Transportation Systems*24:7898–7910. https://doi.org/10.1109/TITS.2021.3093207.

Wang, P., Z. Tong, and S. Nie. 2023. “A Causal Graph Convolutional Network considering Missing Values for Spatio-Temporal Prediction.” *Acta Geodaetica et Cartographica Sinica* 52:818.

Wang, X., X. Wang, X. Yin, K. Li, L. Wang, R. Wang, and R. Song. 2023. “Distributed Lstm-GCN-based Spatial–temporal Indoor Temperature Prediction in Multizone Buildings.” *IEEE Transactions on Industrial Informatics* 20:482–491. https://doi.org/10.1109/TII.2023.3268467.

Wu, Y., Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, and Y. Fu. 2019. “Large Scale Incremental Learning.” In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Wu, C., Y. Guo, H. Guo, J. Yuan, L. Ru, H. Chen, B. Du, and L. Zhang. 2021. "An Investigation of Traffic Density Changes Inside Wuhan during the Covid-19 Epidemic with Gf-2 Time-Series Images." *International Journal of Applied Earth Observation and Geoinformation* 103:102503. https://doi.org/10.1016/j.jag.2021.102503.

Wu, X., W. Li, D. Hong, J. Tian, R. Tao, and Q. Du. 2020. "Vehicle Detection of Multi-Source Remote Sensing Data Using Active Fine-Tuning Network." *ISPRS Journal of Photogrammetry and Remote Sensing* 167:39–53. https://doi.org/10.1016/j.isprsjprs.2020.06.016.

Wu, Z., S. Pan, F. Chen, G. Long, C. Zhang, and P. Yu. 2020. "A Comprehensive Survey on Graph Neural Networks." *IEEE Transactions on Neural Networks and Learning Systems* 32:4–24. https://doi.org/10.1109/TNNLS.5962385.

Wu, Y., H. Tan, L. Qin, B. Ran, and Z. Jiang. 2018. "A Hybrid Deep Learning Based Traffic Flow Prediction Method and Its Understanding." *Transportation Research Part C: Emerging Technologies*90:166–180. https://doi.org/10.1016/j.trc.2018.03.001.

Wu, C., S. Zhu, J. Yang, M. Hu, B. Du, L. Zhang, L. Zhang, C. Han, and M. Lan. 2021. "Traffic Density Reduction Caused by City Lockdowns across the World during the Covid-19 Epidemic: From the View of High-Resolution Remote Sensing Imagery." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14:5180–5193. https://doi.org/10.1109/JSTARS.2021.3078611.

Xu, S., S. Li, W. Huang, and R. Wen. 2022. "Detecting Spatiotemporal Traffic Events Using Geosocial Media Data." *Computers, Environment and Urban Systems* 94:101797. https://doi.org/10.1016/j.compenvurbsys.2022.101797.

Xu, S., S. Li, and R. Wen. 2018. "Sensing and Detecting Traffic Events Using Geosocial Media Data: A Review." *Computers, Environment and Urban Systems* 72:146–160. https://doi.org/10.1016/j.compenvurbsys.2018.06.006.

Yan, P., X. Liu, F. Wang, C. Yue, and X. Wang. 2021. "Lovd: Land Vehicle Detection in Complex Scenes of Optical Remote Sensing Image." *IEEE Transactions on Geoscience and Remote Sensing* 60:1–13.

Yan, S., Y. Xiong, and D. Lin. 2018. "Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition." *Proceedings of the AAAI Conference on Artificial Intelligence* 32 (1). https://doi.org/10.1609/aaai.v32i1.12328.

Yi, Z., X. C. Liu, N. Markovic, and J. Phillips. 2021. "Inferencing Hourly Traffic Volume Using Data-Driven Machine Learning and Graph Theory." *Computers, Environment and Urban Systems*85:101548. https://doi.org/10.1016/j.compenvurbsys.2020.101548.

Yi, D., J. Su, and W. H. Chen. 2021. "Probabilistic Faster R-CNN with Stochastic Region Proposing: Towards Object Detection and Recognition in Remote Sensing Imagery." *Neurocomputing* 459:290–301. https://doi.org/10.1016/j.neucom.2021.06.072.

Yu, L., B. Du, X. Hu, L. Sun, L. Han, and W. Lv. 2021. "Deep Spatio-Temporal Graph Convolutional Network for Traffic Accident Prediction." *Neurocomputing* 423:135–147. https://doi.org/10.1016/j.neucom.2020.09.043.

Yu, B., Y. Lee, and K. Sohn. 2020. "Forecasting Road Traffic Speeds by considering Area-Wide Spatio-Temporal Dependencies Based on a Graph Convolutional Neural Network (gcn)." *Transportation Research Part C: Emerging Technologies* 114:189–204. https://doi.org/10.1016/j.trc.2020.02.013.

Yu, J., M. E. J. Stettler, P. Angeloudis, S. Hu, and X. Chen. 2020. "Urban Network-Wide Traffic Speed Estimation with Massive Ride-Sourcing Gps Traces." *Transportation Research Part C: Emerging Technologies* 112:136–152. https://doi.org/10.1016/j.trc.2020.01.023.

Yu, B., H. Yin, and Z. Zhu. 2018. "Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting." In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, 3634–3640. International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2018/505.

Zhang, Y., Z. Chen, X. Zheng, N. Chen, and Y. Wang. 2021. "Extracting the Location of Flooding Events in Urban Systems and Analyzing the Semantic Risk Using Social Sensing Data." *Journal of Hydrology*603:127053. https://doi.org/10.1016/j.jhydrol.2021.127053.

Zhang, Y., X. Dong, L. Shang, D. Zhang, and D. Wang. 2020. "A Multi-modal Graph Neural Network Approach to Traffic Risk Forecasting in Smart Urban Sensing." In *2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. Como, Italy, 1–9. IEEE Press. https://doi.org/10.1109/SECON48991.2020.9158447.

Zhang, Y., Y. Li, and F. Zhang. 2024. "Multi-level Urban Street Representation with Street-View Imagery and Hybrid Semantic Graph." *ISPRS Journal of Photogrammetry and Remote Sensing* 218:19–32. https://doi.org/10.1016/j.isprsjprs.2024.09.032.

Zhang, Y., P. Liu, and F. Biljecki. 2023. "Knowledge and Topology: A Two Layer Spatially Dependent Graph Neural Networks to Identify Urban Functions with Time-Series Street View Image." *ISPRS Journal of Photogrammetry and Remote Sensing* 198:153–168. https://doi.org/10.1016/j.isprsjprs.2023.03.008.

Zhang, H., G. Lu, M. Zhan, and B. Zhang. 2022. "Semi-Supervised Classification of Graph Convolutional Networks with Laplacian Rank Constraints." *Neural Processing Letters* 54 (4): 2645–2656. https://doi.org/10.1007/s11063-020-10404-7.

Zhang, H., Y. Zou, X. Yang, and H. Yang. 2022. "A Temporal Fusion Transformer for short-term Freeway Traffic Speed Multistep Prediction." *Neurocomputing* 500:329–340. https://doi.org/10.1016/j.neucom.2022.05.083.

Zhao, S., Z. Chen, Z. Xiong, Y. Shi, S. Saha, and X. X. Zhu. 2024. "Beyond Grid Data: Exploring Graph Neural Networks for Earth Observation." *IEEE Geoscience and Remote Sensing Magazine* 13:175–208. https://doi.org/10.1109/MGRS.2024.3493972.

Zhao, T., Z. Huang, W. Tu, F. Biljecki, and L. Chen. 2023. "Developing a Multiview Spatiotemporal Model Based on Deep Graph Neural Networks to Predict the Travel Demand by Bus." *International Journal of Geographical Information Science* 37 (7): 1555–1581. https://doi.org/10.1080/13658816.2023.2203218.

Zheng, C., X. Fan, C. Wang, and J. Qi. 2020. "GMAN: A Graph Multi-Attention Network for Traffic Prediction." *Proceedings of the AAAI Conference on Artificial Intelligence* 34 (1): 1234–1241. https://doi.org/10.1609/aaai.v34i01.5477.