RESEARCH ARTICLE

A Highway In-Transit Vehicle Position Estimation Method Considering Road Characteristics and Short-Term Driving Style

GUANGHAO LUO [1], FUMIN ZOU [1], FENG GUO[1], QIQIN CAI[2], TING YE[1], AND GEN XU[3]

[1]Fujian Key Laboratory of Automotive Electronics and Electric Drive, Fujian University of Technology, Fuzhou 350108, China
[2]School of Mechanical Engineering and Automation, Huaqiao University, Xiamen 361021, China
[3]Bohai Bank Company Ltd., Hefei Branch, Hefei 230000, China

Corresponding author: Feng Guo (mapli@fjut.edu.cn)

ABSTRACT Existing vehicle position estimation methods are mostly based on Global Positioning System (GPS) or a fusion of GPS and machine learning methods to realize vehicle position estimation. While highway tunnels are many, GPS signals are easy to be interfered, and the vehicle loading rate of GPS devices is limited, this kind of method can not be realized in a wide range of applications. In this context, taking into account the ETC equipment that has been deployed and applied in large scale in China, the vehicle equipment loading rate is over 90%, but the ETC gantry interval is large, and it is not possible to effectively perceive the vehicle driving status inside the segment. Therefore, this paper is based on the ETC transaction data to build the basic driving characteristics and short-term driving style of the vehicle history segment, using GPS positioning data to build the internal characteristics of the segment, including the characteristics of the road structure within the segment, the pattern of change of the vehicle position, so as to put forward the highway in-transit vehicle position estimation method that considers the road characteristics and short-term driving style. Firstly, the SC-Kmeans-Bilstm vehicle segment speed prediction model based on PCA optimization is constructed by fusing vehicle short-term driving styles; secondly, the road model within the segment is constructed by using moving average and wavelet smoothing methods; lastly, the vehicle position data is temporally stabilized using linear interpolation and first-order inverse difference, and vehicle position estimation within the highway segment is realized by using DLCNN-LSTM-ATTENTION fusion model based on L1 regularization by combining vehicle segment speeds, road characteristics, and vehicle base driving characteristics. Among them, the short-term driving style helps us to obtain the vehicle segment speed more accurately, and the addition of the road model makes this method better explain the variability of the data. The experimental results show that the present method can achieve on-travel vehicle position estimation within 2km with an error of less than 50m in a full-sample highway environment, and can provide over-the-horizon sensing for intelligent vehicles.

INDEX TERMS Vehicle position estimation, highway, road features, SC-Kmeans-Bilstm, spatio-temporal data smoothing, L1 feature selection, DLCNN-LSTM-ATTENTION.

I. INTRODUCTION

In recent years, with the continuous integration of information technology and the transportation industry, the concept of

The associate editor coordinating the review of this manuscript and approving it for publication was Jie Gao[ID].

intelligent transportation systems has attracted attention from all parties. Intelligent transportation systems aim to improve traffic efficiency, optimize traffic flow, and improve traffic safety. The prerequisite for achieving these functions is to have a clear understanding of the location of vehicles in transit. Therefore, in transit vehicle position estimation is a

key foundational technology in the construction of intelligent transportation systems, which has important applications in path navigation, vehicle collaborative control, vehicle collision warning, and other aspects [1], [2].

For highways, accurately estimating the location of vehicles can monitor traffic flow and congestion in real-time, which helps traffic managers understand the condition of the road, take timely measures to alleviate congestion, guide traffic flow to idle roads, and more effectively optimize road conditions. Secondly, by estimating the location of vehicles in transit, more accurate navigation services can be provided for vehicles based on their real-time location information and traffic conditions, optimizing their driving routes, avoiding congestion and cumbersome routes, and thereby improving driving efficiency and safety [3], [4], [5]. In addition, by collecting vehicle location data, traffic management departments can analyze the traffic flow patterns of different periods and regions, providing a basis for traffic policy formulation and rational planning. This helps to plan road construction and transportation facility layout reasonably to meet the growing transportation demand. Therefore, accurately obtaining the location of vehicles in transit is of great significance for ensuring the safety of highway driving and improving the efficiency of highway operation [6], and is also a key technology necessary for building a smart highway.

With the accelerated integration of new-generation information technology and the automotive industry, it has further promoted the development of the intelligent automotive industry. In transit vehicle position estimation is a crucial part of intelligent driving systems and is of great significance for the upgrading of the intelligent driving industry [7]. One of the foundations for achieving high-level intelligent driving is the real-time perception of the driving environment by intelligent vehicles [8]. The environmental composition that causes the greatest impact on vehicles while driving on highways is the surrounding vehicles. Therefore, achieving position perception of vehicles in transit is the core task of intelligent vehicles in perceiving the driving environment, and it is also an important guarantee for the safety and comfort of intelligent vehicles [9]. The current popular perception method is to perceive the location of vehicles around smart cars through sensors such as LiDAR and cameras. However, these methods are limited by the mechanical performance of the equipment, weather, and other factors, and their perception distance to the surrounding driving environment is limited and unstable. In situations where road design obstructs the view of equipment and low visibility, it is easy to cause dangerous working conditions. Secondly, due to the limited perception distance of devices, smart cars are unable to perceive ultra low speed vehicles outside the sight range, resulting in their inability to plan their speed in advance and affecting the driving experience of smart cars. Therefore, stable vehicle position perception is the key to the upgrading and development of the intelligent vehicle industry.

In summary, accurately obtaining vehicle positions is of great significance for the comprehensive construction of intelligent transportation systems and promoting the upgrading of China's automotive industry towards intelligent driving. With the continuous progress of social and economic levels, the number of private cars is bound to increase significantly, which will bring severe challenges to the transportation system. At that time, the demand for functions such as traffic flow guidance and road condition evaluation of intelligent transportation systems will become even stronger. How to achieve large-scale vehicle position estimation without increasing costs has become a key issue.

We will organize the paper as follows: In the next section, we summarize the relevant literature on vehicle position estimation at home and abroad, and summarize its drawbacks in the field of vehicle position estimation within highway segments; In the third section, we discuss the differences and innovations between our research and existing research; In the fourth section, we introduced the implementation process of our proposed method for estimating the position of vehicles in transit on highways, taking into account road features and short-term driving styles; The fifth section introduces the application effect of our method in actual highway environments; The final section summarizes the application effectiveness of our method.

II. RELATED RESEARCH

With the rapid development of the automotive industry and the increasing complexity of road design and traffic situations, vehicle location estimation methods have shown a trend of upgrading from relying solely on the Global Positioning System (GPS), to multi-sensor collaboration, and then to collaborative iteration of sensors and machine learning methods.

The precise positioning of vehicles was initially achieved by the Global Navigation Satellite System (GNSS) and the GPS. However, due to communication interruptions and multipath errors in tunnels, canyons, and other road segments, GPS and GNSS have poor performance and cannot meet the requirements of modern intelligent transportation systems [10], [11], [12], [13]. Therefore, for many application scenarios that require precise and reliable positioning, independent GNSS or GPS is considered unreliable [2], [14], [15], [16].

To solve the problem that GPS or GNSS alone is prone to interference, Zongwei Wu [15] et al. proposed a method to improve the attitude estimation accuracy of a low-cost inertial navigation system/GPS (INS/GPS) integrated vehicle by utilizing the heading angle measured by the GPS, which effectively reduces the effects of yaw angle error, sideslip angle, and the noise of the GPS measurements, and improves the positioning accuracy compared to the GPS/GNSS alone. GPS/GNSS alone improves the positioning accuracy. Due to the complementary nature of sensors [18], [19], [20], information from GPS and vehicle motion sensors is widely

used for vehicle position estimation to obtain reliable and accurate vehicle position. Rezaei and Sengupta [21] et al. proposed a vehicle position estimation scheme based on the integration of GPS and vehicle sensors and used a Kalman filter for data fusion to improve the vehicle position estimation accuracy. The loading rate and cost of vehicle motion sensors are always difficult to balance

To address the problem of GPS positioning accuracy degradation or signal loss when a vehicle moves in an area where GPS signals cannot be received (tunnels or underpasses) or in an area where very strong multipath propagation occurs (areas surrounded by buildings covered by high glass). Omar et al. [22] designed a position estimation method for the integration of GPS and waypoint projection nav-igation systems, and through the simulation to verify the effectiveness of their method. Wang et al. [23] et al. first proposed a robust Direction Of Arrival (DOA) estimation method based on sparse Bayesian learning (SBL) to achieve DOA estimation of target vehicles under nonuniform noise conditions. Then, based on the DOA estimation results, every two base stations in the localization system cross-localize the target vehicle once. Finally, based on the results of three cross-localizations, robust localization can be achieved. A large number of simulation results show the effectiveness and superiority of the method. Guo et al. [24] intro-duced multi-output (FDA-MIMO) radar into an intelligent transportation system (ITS) and used tensor decomposition to process transportation big data (TBD) to improve the real-time performance of target position estimation. An algorithm is proposed for angle and distance estimation in multi-output radar systems where array gain phase error and spatial color noise coexist. Firstly, a four-dimensional tensor was constructed using the temporal irrelevance of colored noise to eliminate the impact of colored noise on ITS. Secondly, a directional matrix containing target information is obtained through parallel factorization. An optimization problem was constructed for the array gain phase error, and the Lagrange multiplier method was used to solve the optimal solution. The influence of gain phase error is eliminated by utilizing the optimal solution and directional matrix. Finally, the position information of the car was obtained by fitting and solving using the least squares method (LS). Havyarimana et al. [25] proposed a fusion framework based on sparse Gaussian Wigner prediction (SG-WP). This method assumes that the measurement noise is a non-Gaussian distribution, and uses a generalized error distribution as an approximation of the non-Gaussian density. It combines the advantages of random matrix theory and sparse characteristics to provide enhanced vehicle positioning capabilities. Jo et al. [26] addressed a significant limitation in existing models, which often assume that vehicles travel on a flat plane without con-sidering the impact of road slopes. Furthermore, they highlighted the high cost associated with three-dimensional vehicle positioning equipment. They analyzed how road slope affects location estimation and proposed an estimation

algorithm that accounts for this influence. By compensating for errors caused by roadside slopes, this algorithm enhances the precision and reliability of location estimation. However, a common drawback of the methods mentioned above is their reliance on GPS devices or base stations for assistance. This limitation prevents scalable applications and confines them to vehicles equipped with GPS or necessitates large-scale base station deployments. Consequently, the scope of the application is significantly restricted, and the associated costs are substantially increased.

With the continuous advancement of communication tech-nology, Vehicular Ad-Hoc Networks (VANETs) have found widespread applications in intelligent transportation systems [27], [28], [29], [30], [31]. Tsai et al. [32] introduced a col-laborative positioning algorithm (CPDR) aimed at improving GPS location accuracy within VANETs by incorporating Dead Reckoning (DR) algorithms. In this work, the DR algorithm helps filter out some unreasonable GPS positions by referencing travel history records. However, VANETs pose several challenges, including highly heterogeneous vehicle network design, security, privacy concerns, and the dynamic nature of vehicular mobility. These factors create additional challenges for protocol designers. Particularly, the constantly changing scenarios due to vehicle mobility result in short lifetimes for multi-hop paths. In such situations, protocols that rely on knowledge of the system's state can be inefficient due to frequent network changes. Moreover, VANET applications may require a different protocol stack [33]. Additionally, VANETs must grapple with the trade-off between the deployment rate of vehicular hardware devices and the associated cost.

In recent years, neural network models have gained widespread adoption in intelligent transportation systems due to their outstanding performance. Wan et al. [34] introduced a novel system architecture incorporating Massive Multiple-Input Multiple-Output (MIMO) or Reconfigurable Intelligent Surfaces (RIS) along with multiple autonomous vehicles for vehicle positioning. By leveraging geometric algebra, they reformulated the Direction of Arrival (DOA) and polarization estimation problem as a new block sparse recovery problem. They achieved DOA and polarization parameter estimation for autonomous vehicles with relatively low computational complexity using the deep network architecture SBLNet. Meanwhile, Yuexia and Chong [35] proposed a high-precision vehicle localization method based on neural networks and Road Side Units (RSUs) fingerprints. They divided the localization area into uniform grid regions, collected Received Signal Strength Indicator (RSSI) data from different RSUs in each grid region, and constructed an RSU fingerprint database. During the localization phase, they utilized Backpropagation Neural Networks (BPNN) to estimate the approximate coordinates of the target vehicle. Using these estimated coordinates as the center and the maximum prediction error of the BPNN as the radius, they constructed a fingerprint-matching region. This approach

allowed them to calculate the precise coordinates of the target vehicle through local fingerprint localization. Alzyout and Alsmirat [36] introduced a short-term vehicle location prediction framework that enhances prediction accuracy and framework execution time by dynamically adjusting parameters and employing both multi-selective and single-selective ARIMA models. Anitha and Duraiswamy [37] addressed the limitations of current methods in vehicle location prediction, which often lack analysis of both current and future vehicle positions and are affected by errors in GPS location data. They proposed a heuristic mobile vehicle location prediction algorithm, demonstrating that this heuristic algorithm can accurately predict the future positions of vehicles. In response to the shortcomings of existing methods, Fan et al. [38] presented a deep learning-based approach for next-step location prediction (DLNLP), incorporating contextual information into urban-level vehicle motion prediction. Experimental results indicate that DLNLP outperforms other methods such as MM, WMM, CNN, and LSTM in terms of accuracy, recall, and balanced F-score. Long et al. [39] addressed the limitation of existing location prediction models, which focus solely on location and time, oversimplifying the regularities and preferences in human mobility. Furthermore, existing state-of-the-art RNN-based models may fail to capture long-term patterns in sparse scenarios due to the lack of sequential dependencies. They proposed an individual vehicle location prediction method that utilizes travel patterns and preferences. Experiments conducted on three real vehicle trajectory datasets, each containing over 10,000 individual vehicles, demonstrated that the proposed model outperforms state-of-the-art models by 7%-10% in terms of prediction accuracy. Xiao and Nian [40] tackled the issue of low prediction accuracy in existing methods. They introduced a vehicle location prediction algorithm based on spatiotemporal feature transformation and a hybrid LSTM neural network. This approach effectively reduces information loss in vehicle trajectories and enhances the accuracy of vehicle location prediction. However, it should be noted that this model does not fully consider the impact of road conditions.

After analyzing the literature review mentioned above, we can identify the following shortcomings in the existing methods:

(1) Most of the methods mentioned above have largely overlooked the influence of road structure characteristics on vehicle positioning, especially in provinces like Fujian, which are primarily mountainous, and where highways traverse through mountain ranges. Therefore, methods that do not consider road features may not necessarily be suitable in predominantly mountainous regions.

(2) The low adoption rate of in-vehicle positioning devices hinders their wide-spread application, thus limiting the provision of comprehensive and accurate posi-tioning information for intelligent transportation systems.

(3) The addition of extra in-vehicle positioning equipment results in an increase in overall vehicle costs. From the perspective of vehicle owners, unless there are significant benefits, they may not be willing to bear the associated expenses.

III. OUR CONTRIBUTION

In response to the challenges faced by existing methods and the existing infrastructure, this paper proposes a highway in transit vehicle position estimation method that considers road features and short-term driving style by integrating ETC data (Edata) and invehicle GPS positioning data. As of the end of 2022, Fujian Province's highways have generated over 5 million ETC transaction data daily, providing basic driving characteristics for estimating the position of vehicles in transit on highways. However, since Edata only records the status information of vehicles during information exchange with the gantry, the vehicle status information within the highway segment cannot be obtained. GPS positioning data records the driving status of vehicles within a certain time interval, providing a modeling basis for estimating the position of vehicles within a highway segment. The main contribution of this article is:

(1) This method integrates Edata and GPS positioning data (Gdata) for vehicle position estimation, excavates vehicle driving patterns in Edata to construct basic driving features, and uses corresponding vehicle Gdata to generate road segment features and target state variables. Due to the large-scale deployment of ETC devices on highways, this method has better universality.

(2) In the vehicle segment speed prediction model, based on the spatiotemporal dependence of the vehicle segment speed, the SC-Kmeans-Bilstm model based on PCA optimization is proposed, which fully takes into account the influence of the spatiotemporal dependence of the vehicle segment speed and the vehicle's short-term driving style to improve the model prediction accuracy.

(3) In vehicle location estimation, to address the instability of vehicle location data in the spatial and temporal dimensions, linear interpolation and first-order backward difference are used to improve the data smoothness and maximize the retention of the information entropy of the data; secondly, the road features are taken into account into the model to minimize the impact of the changes in spatial and temporal features of the road on the performance of the position estimation model.

(4) The DLCNN-LSTM-ATTENTION fusion module based on L1 regularization is designed to achieve multimodal feature fusion and abstraction at a deeper level to better understand the multidimensional relationships of the data. Also the inclusion of L1 regularization avoids the overfitting phenomenon. In addition, the two-layer CNN can interact and fuse different features in the second convolutional layer to capture higher-level data patterns. The LSTM layer has memory units and gating mechanisms to better capture and

utilize long-term dependencies in sequential data. Finally, by applying an attention mechanism on the output of the LSTM layer, we can adaptively learn the important weights for each time step.

(5) Although vehicle position estimation has always been a research hotspot in the academic community, to our knowledge, we are the first team to propose using Edata and Gdata to jointly achieve vehicle position estimation within highway segments.

IV. MODEL CONSTRUCTION

The highway in transit vehicle position estimation model considering road characteristics and short-term driving style of vehicles first analyzes and processes the data problems existing in Edata and Gdata to improve data matching efficiency and reduce noise that affects the fitting degree of the model, mainly including noise reduction processing for duplicate, missing, and incorrect data. Based on the denoised data, perform bidirectional matching between Edata and Gdata, and construct vehicle basic driving features based on Edata and Gdata respectively. Secondly, the short-term driving style of the vehicle is integrated into the vehicle speed prediction model to obtain accurate vehicle segment passing speeds; Next, Since the segment speed of the vehicle cannot fully reflect the driving pattern of the vehicle within the segment, a road model within the segment is constructed using moving average wavelet transform, and the vehicle segment speed is combined to jointly map the changes in vehicle position within the segment; Finally, based on the basic driving characteristics of the vehicle, the segment speed of the vehicle, and the road characteristics, a DLCNN-LSTM-Attention method for estimating the position of vehicles in transit is proposed to achieve accurate prediction of vehicles in transit. The overall framework of the model is shown in Fig 1. Table 1 summarizes the parameters and characteristic variables involved in this article.

This method integrates Edata and GPS positioning data (Gdata) for vehicle position estimation, excavates vehicle driving patterns in Edata to construct basic driving features, and uses corresponding vehicle Gdata to generate road segment features and target state variables. Due to the large-scale deployment of ETC devices on highways, this method has better universality.

A. DATA PREPROCESSING AND ANALYSIS

When building a model, duplicate data makes the model overly dependent on these data, leading to overfitting and a decrease in generalization ability to new data. Missing data may result in the model not being able to fully utilize information, thereby reducing the accuracy of the model. Data with incorrect information may cause the model to be affected by outliers or unreasonable values, leading to overfitting or errors in these data. Because the experimental data comes from Edata and Gdata in real driving environments, there are inevitably duplicates, omissions, and incorrect information in the data. Therefore, it is necessary

TABLE 1. Notation frequently used.

Symbol	Symbolic meaning
V_S	The Speed of the vehicle in the current segment
V_{S-1}	The speed of the vehicle in the previous segment of the current segment
V_{S-2}	The speed of the vehicle in the last two segments of the current segment
V_{EN}	The Speed of the segment where the vehicle is locatedt
V_{EN1}	The speed of the previous segment in which the vehicle is located
V_{EN2}	The speed of the first two segments in which the vehicle is located
F_S	Current segment flow when the vehicle passes
F_{S-1}	The traffic flow of the previous segment when the vehicle passes in the previous segment
F_{S-2}	The flow of vehicles in the upper two segments when passing through the upper two segments
T	Vehicle type
σ	Owner's short-term driving style
Θ	The slope of the road where the vehicle is located
$P_{previous}$	Distance from the front gantry of the current segment at the last moment
V_{S_pred}	Predicted current segment speed
$Traj$	Vehicle trajectory
L_{PRE}	The distance between the vehicle and the front door frame in the current segment
T_{ROAD}	Road elevation information
$Gdata$	GPS positioning data
$Edata$	ETC data

to conduct deep cleaning of the data to ensure data quality and accuracy and construct features related to vehicle position changes based on the cleaned data as input to the model, to maximize model fit.

1) DATA NOISE REDUCTION

To reduce data noise, the main objects for cleaning Gdata include repeated positioning data of the same vehicle, long positioning time interval, continuously changing speed but unchanged vehicle position, data with a continuous speed of 0km/h but unchanged vehicle position, and data points with trajectory drift outside the highway. Among them, for trajectory drift points, this question uses a drift point detection method based on Gaode map path planning.

As shown in Fig 2, Gdata is affected by factors such as satellite geometry configuration, receiver error, and noise, resulting in individual positioning data points drifting outside the highway network. The drift phenomenon can affect the regularity of vehicle position changes and affect the fitting effect of the model. Therefore, for trajectory drift points, this article proposes a drift point cleaning method based on the Gaode API, which calculates the distance between adjacent two points of the vehicle trajectory through the Gaode API and sets the calculation rule to prioritize high-speed. As long as the vehicle's positioning point drifts outside the highway, its distance to adjacent positioning points must be much greater than the distance between normal adjacent positioning points. This article takes the position change per second at the maximum speed of each segment as the threshold and records the points where the vehicle's position change per second exceeds this threshold as drift points.

By analyzing the traffic speeds of typical segments of highways in Fujian Province, it can be concluded that the

A Method for Estimating the Position of Vehicles in Transit Considering Short Term Driving Style and Road Slope
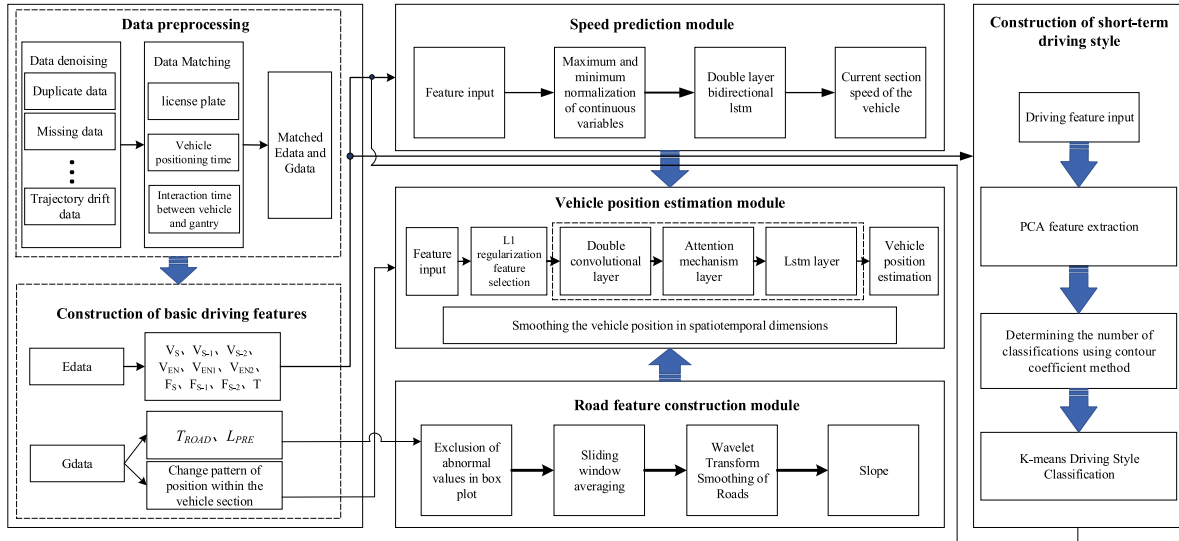

FIGURE 1. Overall architecture of the model.

FIGURE 2. Visualization of Gdata.

average speed of segment 1 is 71 km/h, the maximum speed is 111 km/h, and 85th percentile speed is 85 km/h; The average speed of segment 2 is 80km/h, the maximum speed is 121km/h, and 85th percentile speed is 92km/h; The average speed of segment 3 is 83 km/h, the maximum speed is 111 km/h, and 85th percentile speed is 94 km/h; The average speed of segment 4 is 79 km/h, the maximum speed is 110 km/h, and 85th percentile speed is 92 km/h. 85th percentile speed is a reasonable choice that can represent the speed of most vehicles. This is because some vehicles may be

traveling at slower speeds in normal traffic flow (e.g., because of problems such as traffic congestion, poor road conditions, etc.), while at the same time, some vehicles may be traveling at faster speeds (e.g., speeding). However, most vehicles typically travel at intermediate speeds, so the 85th percentile speed is considered a good indicator to describe the speed of most vehicles. However, to preserve the data samples to the greatest extent possible, this article takes the distance of position change per second at the maximum speed of each segment as the threshold, and records points where the

distance change of adjacent positions of vehicles per second exceeds this threshold as drift points, and removes them.

In addition, there are issues in Edata such as license plate garbled codes, missing transactions caused by no information exchange with the gantry when the vehicle passes through it, and duplicate transactions when passing through the gantry. Due to the occurrence of missed transactions, it becomes difficult to obtain the time for vehicles to enter and exit the gantry, resulting in the inability to perform subsequent data matching. Therefore, this article directly eliminates the missing transaction trajectory. Misregistration of license plates can also lead to data mismatch, so they are directly removed. For duplicate transaction data, this article uses "flagid" and "entity" as the de-duplication subsets to eliminate duplicate data from the data.

2) DATA MATCHING BASED ON SPATIOTEMPORAL LOCATION

The Edata comes from the vehicle information that interacts with the ETC gantry and is captured by the ETC system when the vehicle passes through it. It includes information reflecting the vehicle's attributes such as the transaction time between the vehicle and the gantry, vehicle type, and vehicle entry time, but does not include the driving characteristics of the vehicle within the segment, such as changes in position within the segment, real-time vehicle speed, and other internal vehicle status information within the segment. If the distance between the two gantries is too large, the traffic situation inside the segment changes, and it is easy to distort the results with the actual driving status by estimating the overall traffic status of vehicles inside the segment only from the Edata. The Gdata records the vehicle's position, speed, heading an-gle, and other state information that changes with time inside the segment at certain time intervals, which can reflect the regularity of the vehicle's traveling state changes inside the segment. Therefore, to obtain more accurate vehicle location information within the segment, it is necessary to open the blind spot inside the segment using Gdata in conjunction with Edata. Among them, the segment is a small road segment consisting of two adjacent gantry nodes of the highway, and the gantry that the vehicle passes through first is called the front gantry of the segment, and the gantry that passes through later is called the back gantry of the segment, as shown in Fig 3.

This section proposes two-way data matching based on spatiotemporal location, i.e., the time point of the vehicle's transaction with the front and rear gantries of the segment in Edata determines the time range of the corresponding vehicle's localization in Gdata, which in turn specifies the target vehicle located in the target segment in Gdata. However, the color description of the license plate in the Edata is at the top of the license plate characters, while the characters indicating the license plate color in the Gdata are at the end of the license plate. To match the Edata with the Gdata, it is necessary to perform a string left rotation operation on the license plate information string in the Edata,

and put the characters indicating the license plate color in the Edata to the tail, to facilitate the data matching between Edata and Gdata.

Firstly, obtain the Edata of the target segment and extract the license plate data obu_1 passing through the target segment from the data; Secondly, match the vehicle data corresponding to obu_1 in the Gdata based on obu_1. Since Gdata cannot directly filter out the data of the corresponding road segment, according to the transaction time of the vehicle and the ETC gantry before and after the segment as a qualifying condition, extract the GPS positioning data within the time range of the information interaction between the corresponding vehicle and the front and rear gantries of the target segment in Gdata, and then we can obtain the GPS positioning data inside the target segment. Finally, by matching the 'obupdate' and 'entime' fields in Gdata, reverse search the Edata for the corresponding time when the vehicle enters and exits the current segment. If there are positioning points at the same time in Gdata, no further processing will be performed. If there are no corresponding positioning points in Gdata, interpolate the time point data of the vehicle entering and exiting the segment in Gdata. Record the distance between the vehicle and the front gantry of the current segment as 0 when entering the segment; When leaving the segment, the distance between the vehicle and the current gantry is recorded as the total length of the segment.

3) CONSTRUCTION OF BASIC DRIVING CHARACTERISTICS OF VEHICLES

This section is based on the target vehicles within the target segment matched in the previous step, and excavates the hidden information in Edata to construct the basic features of vehicle position estimation. Basic features mainly refer to features that can be extracted from data without the need for deep processing by machine learning models.It mainly includes F_S, F_{S-1}, F_{S-2}, V_{S-1}, V_{S-2}, V_{EN}, V_{EN1}, V_{EN2}, T, as shown in Table 1. And the basic features need to be constructed based on the vehicle trajectory. The vehicle trajectory in Edata is mainly composed of the gantry nodes that the vehicle passes through. When a vehicle is driving on a highway, the sequence of gantry position nodes it passes through is its driving trajectory, as shown in equation (1)

$$\text{Traj} = <\text{Node}_1, \ldots, \text{Node}_n> . \tag{1}$$

where Node_1 is the starting point of the trajectory and Node_n is the end point.

Firstly, traverse the driving trajectory of each vehicle to obtain the passing time nodes of adjacent gantry frames in the vehicle trajectory, and calculate the time it takes for the vehicle to pass through the segment composed of adjacent ganties. Secondly, obtain the coordinates of adjacent gantries and use the Gaode API to calculate the distance between adjacent gantries; Then, calculate the vehicle's travel speed on historical segments based on kinematic formulas. Due to the influence of driver status and traffic conditions on vehicle speed, this article only takes the speed V_S of the vehicle
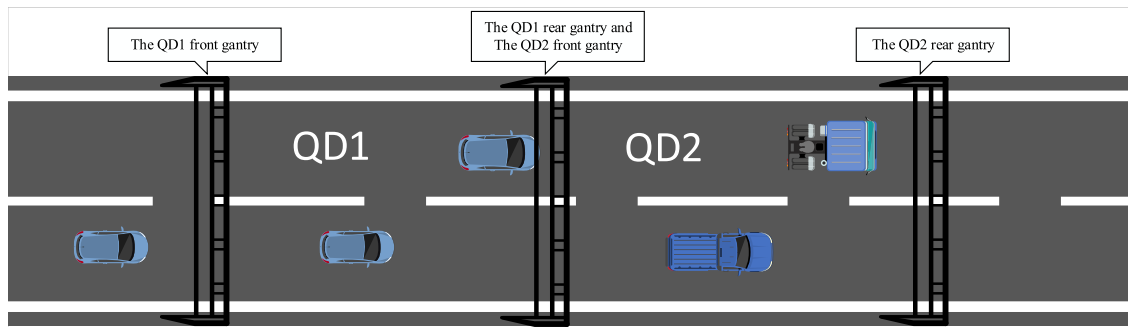
FIGURE 3. Schematic diagram of segments.

in the target segment and the speed V_{S-1} and V_{S-2} of the first two segments closest to the current segment. To better perceive the changing patterns of the target vehicle in the model, this article further calculates the segment speed V_{EN} of the vehicle from ten minutes before entering the target segment to the moment of entering the segment, as well as the corresponding segment speeds V_{EN1} and V_{EN2} when the vehicle passes through the previous two segments.

To maintain the authenticity of traffic flow to the greatest extent and avoid significant differences between the calculated traffic flow and the traffic flow during vehicle travel due to large time differences, this article only calculates the total number of vehicles passing through the front and rear gantries of the segment from the first ten minutes of the target vehicle entering the segment to the transaction time point between the target vehicle and the front gantry of the segment as the corresponding traffic flow F_S of the target vehicle during the passage period. To match the segment speed of the vehicle in the previous step, this article also calculated the traffic flow F_{S-1} and F_{S-2} of the vehicle at the corresponding time when passing through the first two segments of the current segment. In addition, the Edata includes vehicle types, so vehicle type features are directly extracted from the data.

B. PREDICTION MODEL OF VEHICLE SEGMENT SPEED CONSIDERING SHORT-TERM DRIVING STYLE

In the driving environment of highways without significant bends, vehicle speed is one of the important characteristics that reflect changes in vehicle position. The faster the vehicle speed, the greater the change in position. In actual driving environments, speed is limited by factors such as traffic flow, vehicle mechanical performance, and road structural characteristics. This section analyzes the relevant factors that affect speed changes. The input variables are required to construct a vehicle speed prediction model based on the analysis results.

1) ANALYSIS OF CORRELATION CHARACTERISTICS OF VEHICLE SPEED CHANGES
a: ANALYSIS OF SPEED CHARACTERISTICS IN THE DIMENSION OF VEHICLE FLOW
The traffic flow affects the driving environment of vehicles. When the traffic flow reaches a certain level, the speed

of vehicles will also be constrained to varying degrees. Figure 4 shows the difference in vehicle speeds between two segments with different traffic flows. As shown in Fig 4, in segments with high traffic flow, there are significantly more vehicles with speeds below 50km/h than in segments with low traffic flow, and as the flow increases, more and more low-speed vehicles appear in both segments. Therefore, when the traffic flow reaches a certain amount, it becomes an important feature that affects speed, thereby affecting changes in vehicle position.

b: ANALYSIS OF VEHICLE TYPE CHARACTERISTICS
The speed is not only constrained by the driving environment of the vehicle but also by the type of vehicle. Under the same road conditions, due to the limitations of vehicle mechanical performance, the speed of large trucks may be lower than that of small cars, and due to safety considerations, the braking distance required for large trucks is long. Drivers will also control the speed appropriately to prevent emergencies. As shown in Fig 5, by analyzing the traffic speeds of various types of vehicles in different segments of Fujian Province's highways, it is found that regardless of which segment, the average speed, 85% percentile speed, and maximum speed of passenger cars are greater than those of trucks and special operation vehicles, and the difference is significant. At the 15% percentile speed, only 15% percentile speeds for passenger cars in segment 2 are lower than that of trucks and specialized work vehicles. From this, it can be seen that different types of vehicles have significant speed differences due to differences in mechanical performance. The speed of passenger cars is much higher than that of other types of vehicles. Therefore, vehicle types should also be included in the feature library that affects speed changes.

c: ANALYSIS OF DRIVING STYLE CHARACTERISTICS
Driving style refers to the behavior and habits of the driver on the road, which directly determines the speed of the vehicle on the road [41]. Radical drivers tend to exceed speed, accelerate rapidly, and brake sharply, often ignoring traffic rules and speed limit signs. This driving method may cause the vehicle's speed to soar in a short period. On the contrary, a cautious and steady driving style often leads to slower speeds. Prudent and steady drivers tend to follow
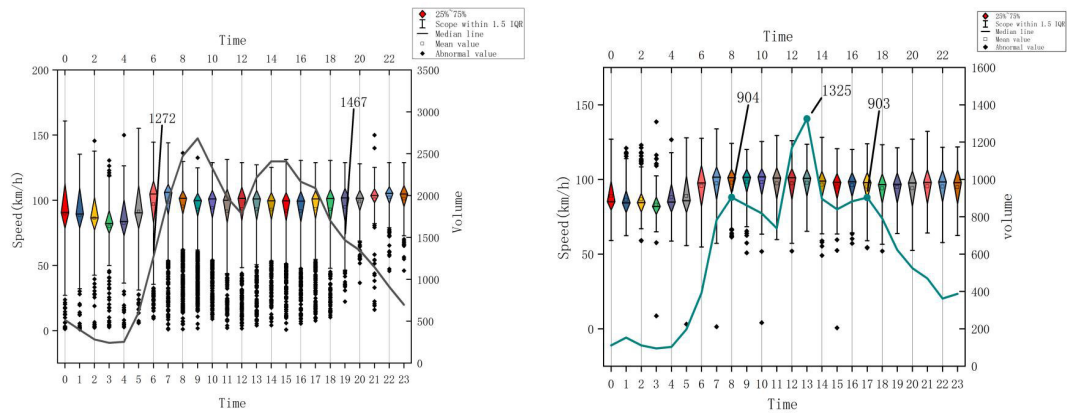
(a) **Performance chart of vehicle speed during high traffic flow.**

(b) **Performance chart of vehicle speed during low traffic flow.**

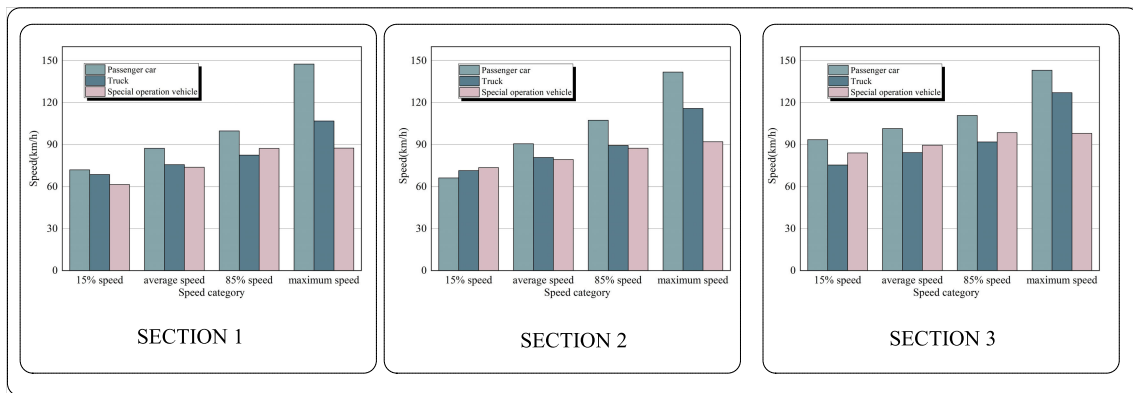**FIGURE 4.** Speed performance chart under different dimensions of traffic flow.

FIGURE 5. Statistical chart of segment speed characteristics for different types of vehicles.

traffic rules, maintain a moderate speed, and drive the vehicle steadily. They pay attention to road safety and avoid sharp turns and brakes, resulting in relatively slow speeds. An economical and energy-saving driving style can also affect vehicle speed. Economically energy-saving drivers focus on efficient driving, adopt smooth acceleration and deceleration, and plan their driving routes reasonably, effectively reducing fuel consumption.

From Fig 5, it can be seen that there is a significant difference between the maximum speed of passenger cars and trucks and the 15% percentile speed, while the difference between special operation vehicles is small. The reason for this phenomenon may be that most special operation vehicles travel with tasks, and the drivers of special operation vehicles have strict requirements during recruitment, rich driving experience, and high comprehensive driving quality, so their speed is relatively stable. Buses and trucks are mostly private cars or public vehicles owned by small businesses, and drivers can drive their vehicles according to their habits. So its speed fluctuates greatly. In summary, different driving styles have a significant impact on the fluctuation and speed of vehicle

speed, so driving style should become one of the important factors in perceiving vehicle speed.

2) SHORT-TERM DRIVING STYLE CONSTRUCTION METHOD BASED ON SC-KMEANS CLUSTERING

Driving style is affected by many factors, and driving habit is an important determinant. From the perspective of the overall journey of car owners, each car owner has a fixed driving style, including radical type, general type, and cautious type, which are determined by daily driving habits. In the actual driving process, driving style is affected by driving tasks, driving environment, driver subjective factors, and so on, and there are often changes between different styles, especially the driving style changes in specific scenes that have strong randomness. Therefore, this paper proposes a short-term driving style considering the spatiotemporal characteristics, that is, considering the driving state with the strongest correlation with the current vehicle state in time and space to build a short-term driving style, to more accurately capture the driving characteristics of the vehicle's current journey.

This section is based on the basic driving characteristics of the vehicle, including V_{S-1}, V_{S-2}, V_{EN1}, V_{EN2}, F_{S-1}, F_{S-2}, T; Then, the contour coefficient method is used to determine the optimal driving style classification number; Finally, the K-means algorithm is used to classify the spatiotemporal characteristics of vehicle driving. The overall process of the method is shown in Fig 6.

a: SELECTION OF DRIVING STYLE CLASSIFICATION NUMBER

Determining the number of driving styles is a subjective problem in theory. There is no fixed rule or standard to determine how many driving styles should be divided into. However, when classifying driving styles, we need to comprehensively consider many factors, including road environment, drivers' preferences, and driving habits. Due to the complexity and diversity of driving styles, it is difficult to get an accurate number of categories simply by subjective judgment. The contour coefficient method can provide an objective and quantitative evaluation method, help us select the most appropriate classification number without fixed rules, and increase the reliability and scientificity of the analysis results.

Contour coefficient [42], i.e. SC index, indicates the degree of compactness and dispersion among various types of samples after clustering. The smaller the distance between samples in the same class and the larger the distance between samples in different classes, the greater the value of SC(i) and the better the clustering effect. Therefore, it is often used as a performance index to evaluate the clustering results. The SC(i) calculation formula is as follows:

$$SC(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}'$$ (2)

In equation (2), $SC(i) \in [-1, 1]$ represents the contour coefficient of sample i, $b(i) = \min\{b_{i1}, b_{i2}, \ldots, b_{ik}\}$ represents the minimum value of the dissimilarity between clusters, where b_{ik} represents the average distance of all samples from the sample to other clusters C_j, and a(i) represents the average distance of the sample to other samples in the same cluster. When SC(i) is close to 1, it indicates that the sample clustering is reasonable; when SC(i) is close to -1, it indicates that the sample classification is unreasonable and should be classified to another cluster; if SC(i) is close to 0, it indicates that the sample is located at the boundary of two clusters.

b: RECOGNITION OF SHORT-TERM DRIVING STYLE

After determining the number of driving style classifications, it is necessary to classify them according to the driving characteristics of vehicles. Raw data contains high-dimensional features, while high-dimensional data sets increase model complexity and computational cost. Dimensionality reduction can map data to low dimensional space, reduce the requirements of calculation and storage, and reduce the complexity of model training and inference.

The main goal of PCA is to map the high-dimensional data into the low-dimensional space through some kind of spatial linear projection, and at the same time try to ensure the maximum variance of the data in the low-dimensional space of the target, to prevent the loss of more information of the original data [43]. Set the sample set $M = \{X_1, X_2, \ldots, X_M\}$, corresponding to the n-dimensional feature $X^i = (x_1^i, x_2^i, \ldots, x_N^i)$. For the feature centralization, the covariance matrix is as shown in Equation (3)-(4).

$$C = \begin{bmatrix} \text{cov}(x_1, x_1) & \text{cov}(x_1, x_2) \\ \text{cov}(x_2, x_1) & \text{cov}(x_2, x_2) \end{bmatrix}$$ (3)

$$\text{cov}(x_1, x_1) = \frac{\sum_{i=1}^{M}(x_1^i - \bar{x}_1)(x_1^i - \bar{x}_1)}{M - 1}$$ (4)

The eigenvalues λ and corresponding eigenvectors u, of the covariance matrix C will be ranked from largest to smallest and the first k eigenvectors corresponding to them will be selected. The formula for calculating the new eigenvectors after dimen-sionality reduction is shown in equation (5):

$$\begin{bmatrix} y_i^1 \\ y_i^2 \\ \ldots \\ y_i^k \end{bmatrix} = \begin{bmatrix} u_1^T \cdot (X_1^i, X_2^i, \ldots, X_n^i)^T \\ u_2^T \cdot (X_1^i, X_2^i, \ldots, X_n^i)^T \\ \ldots \\ u_k^T \cdot (X_1^i, X_2^i, \ldots, X_n^i)^T \end{bmatrix}$$ (5)

Based on the new feature volume after dimensionality reduction, the K-means algorithm measures the similarity of different data objects by selecting an appropriate distance formula. The distance between data is inversely proportional to the similarity, i.e. the smaller the similarity, the larger the distance.

The K-means algorithm first gives the corresponding initial clustering center C based on the number of driving style classifications and calculates the distance from the initial clustering center to the rest of the data objects, which is chosen in this paper as the Euclidean distance [44]. In this paper, the Euclidean distance is chosen, and the formula for the Euclidean distance from the clustering center to other data objects in the space is shown in equation (6):

$$d(x, C_i) = \sqrt{\sum_{j=1}^{m}(x_j - C_{ij})^2}$$ (6)

In equation (6), x is the data object, C_i is the ith clustering center, m is the dimension of the data object, and x_j, C_{ij} are the attribute values of the data object x and the jth dimension of the clustering center C_i.

Based on the Euclidean distance, the similarity is measured and the target data with the highest similarity to the clustering center is assigned to the corresponding cluster. After allocation, the data objects in the k clusters are averaged to form a new round of clustering centers, thus reducing the sum of error squares of the dataset, calculated as equation (7):

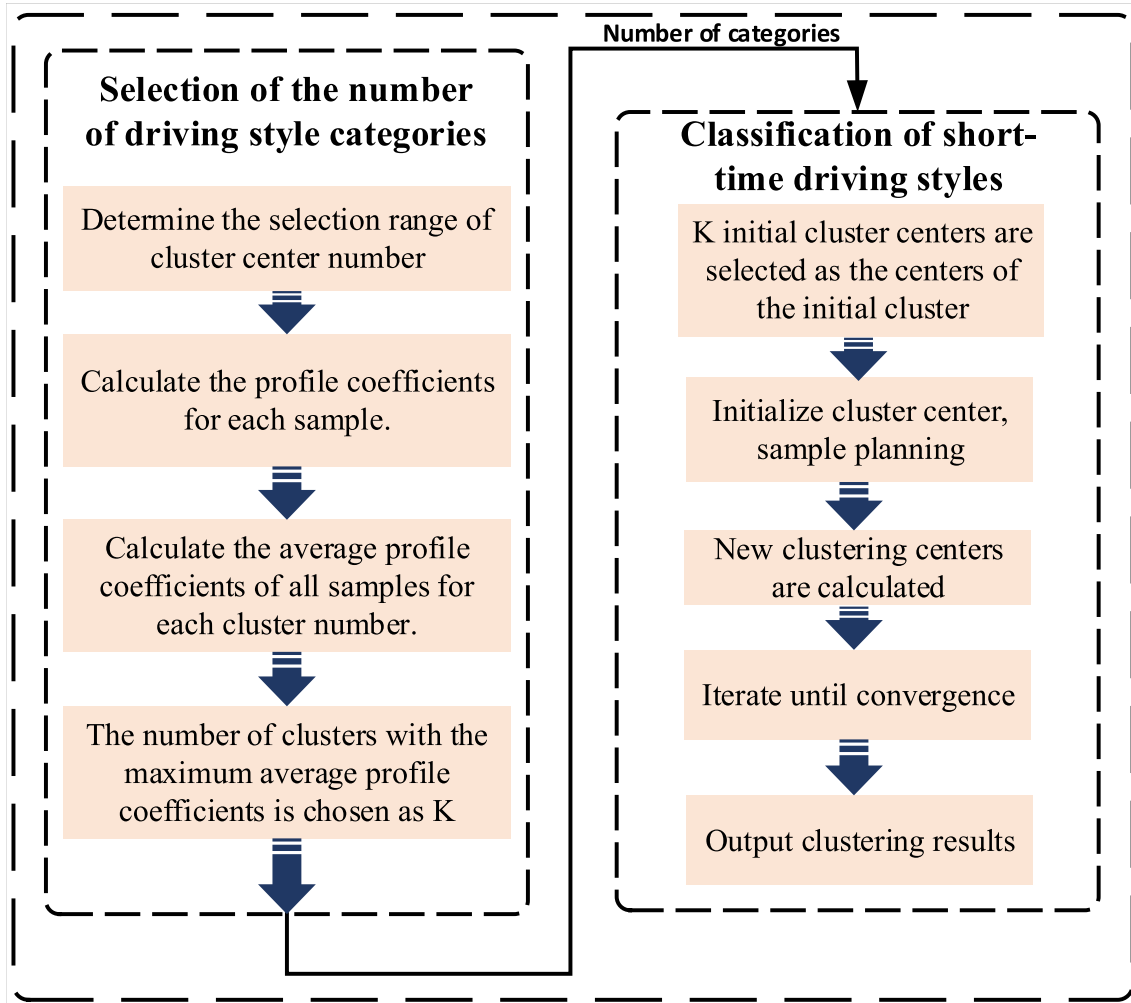$$SSE = \sum_{i=1}^{k}\sum_{x \in C}|d(x, C_i|^2$$ (7)

FIGURE 6. Flow chart of short-term driving style construction.

The magnitude of the value of SSE is used as a measure of how good the clustering results are, and when it no longer changes or converges, the iteration is stopped and the final result is obtained.

3) PREDICTION OF VEHICLE SEGMENT SPEED BASED ON BI-LSTM

The so-called bidirectional LSTM is that one LSTM unit processes forward input and the other unit processes reverse input [45]. It can retain both past information and future information, which can make the network better understand the information before and after a certain time. The principle of bidirectional LSTM is shown in Fig 7.

If used α represents the forward propagation sequence in Fig 7, using β Represents the backpropagation sequence. Given that the input of the characteristic graph is represented by $(x_1, \ldots, x_n)$, the calculation of Bi-LSTM is as follows equation (8)-(10):

$$h_t^\alpha = \text{sigmod}(u^\alpha x_t + \omega_\alpha h_{t-1} + b_\alpha) \tag{8}$$

$$h_t^\beta = \text{sigmod}(u^\beta x_t + \omega_\beta h_{t-1} + b_\beta) \tag{9}$$

$$y_t = \text{sigmod}(h_t^\alpha \upsilon_\alpha + \upsilon_\beta h_t^\beta + b_f) \tag{10}$$

The weight vectors corresponding to the forward and backward propagation of the expression are u^α, u^β, ω_α, ω_β, ω_β, υ_β, respectively; the forward and backward propagation of the hidden layer at the moment t is h_t^α, h_t^β, respectively; the output of the node at the moment t is denoted as y_t; the corresponding bias is denoted as b_α, b_β, b_f.

The bidirectional LSTM used in this paper contains two bi-directional LSTM layers and a fully connected layer. Among them, the first bi-directional LSTM layer, which contains 64 LSTM units, has an activation function of ReLU and returns complete sequence information. This layer is responsible for extracting temporal dependencies from the input data and allows information to be passed forward and backward simultaneously to fully capture the characteristics of the time series. The second bi-directional LSTM layer, also contains 64 LSTM units with an activation function of ReLU. this layer further deepens the understanding of the time
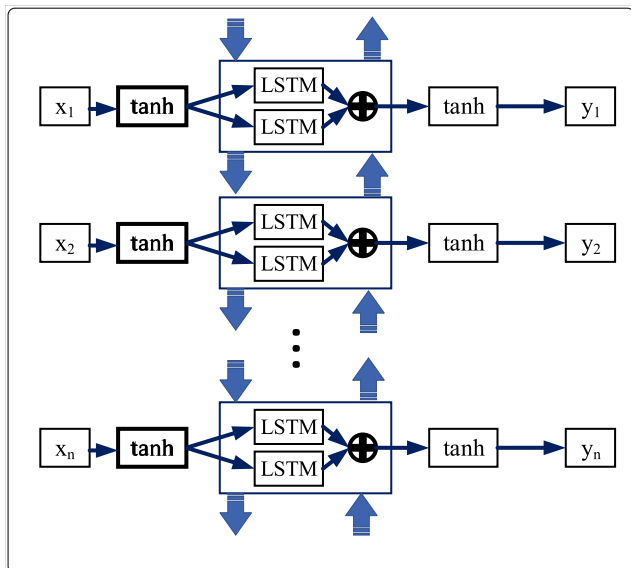
FIGURE 7. Schematic diagram of bi-directional LSTM.

series data and captures more complex temporal patterns. The output layer contains 1 neuron and is used to generate the final prediction. Finally, the model is compiled using Adam optimizer and mean square error as a loss function.

C. ROAD MODEL BASED ON MOVING AVERAGE WAVELET SMOOTHING FOR BOX GRAPH ANOMALY DETECTION

A basic vehicle motion model widely used for 2D heading projection is shown in equation (11) - (13), which adopts a 2D planar road assumption with state vectors consisting of heading ψ, east X, and west position Y. The model is based on a 2D planar road assumption.

$$\Psi_k = \Psi_{k-1} + \triangle T \cdot \omega_{XY} \tag{11}$$

$$X_k = X_{k-1} + \triangle T \cdot V_{XY} \cos(\Psi_{k-1} + \triangle T \cdot \omega_{XY}) \tag{12}$$

$$Y_k = Y_{k-1} + \triangle T \cdot V_{XY} \sin(\Psi_{k-1} + \triangle T \cdot \omega_{XY}) \tag{13}$$

However, as shown in Fig 8, when there is a gradient in the road, it will decompose the gravity force formed by the total weight of the vehicle and the passengers into the resistance of the vehicle to move forward. When the output power is kept constant and friction increases, P is a constant when the output power is kept constant as shown in equation (14). If the friction increases, then the driving force must increase to maintain a constant power. According to Newton's second law, the driving force is proportional to the acceleration (a) and mass (m) as shown in equation (15). Therefore, when friction increases, the driving force must be increased to maintain the same power, but this also results in a decrease in acceleration. The speed of the vehicle is proportional to the product of acceleration and time as shown in equation (16). As the acceleration decreases, the vehicle speed will decrease. Therefore, there is an effect of uphill position on vehicle

speed, especially for large loaded vehicles.

$$P = F_{drive} * v \tag{14}$$

$$F_{drive} = m * a \tag{15}$$
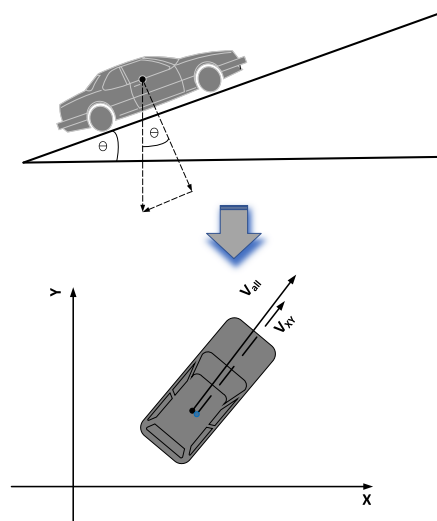
$$v = a * t \tag{16}$$

FIGURE 8. Schematic diagram of vehicle speed under the influence of road gradient.

Edata can only be used to calculate the segment speed of the vehicle by obtaining the time point and segment distance of the vehicle entering and leaving the segment and using the vehicle kinematics formula. However, the vehicle segment speed cannot reflect the real-time speed of the vehicle. For example, if the first half of a segment is uphill and the second half is downhill, the probability is that the vehicle's speed on the uphill portion will be lower than on the downhill portion. Due to the offset of uphill and downhill segments, the speed of vehicles in the whole segment may be similar to that in flat segments. Therefore, the segment speed of the vehicle can only accurately reflect the time when the vehicle arrives at the rear gantry of the segment, and cannot reflect the position change law of the vehicle within the segment. This paper further constructs the road structure characteristics and the vehicle segment speed to jointly map the vehicle position change law in the segment. The overall construction process of the road model is shown in Fig 9.

1) DETECTION OF ABNORMAL ROAD ELEVATION DATA BASED ON BOX GRAPH

There are many tunnels on the highways in Fujian Province, and the GPS signals are susceptible to interference, resulting in uneven road elevation data in the raw data, and a common error is the existence of multiple different values of elevation information at the same location. To reduce the data anomalies that lead to road model distortion, the box-and-line diagram method is used to screen the elevation data under the same location dimension for outliers.
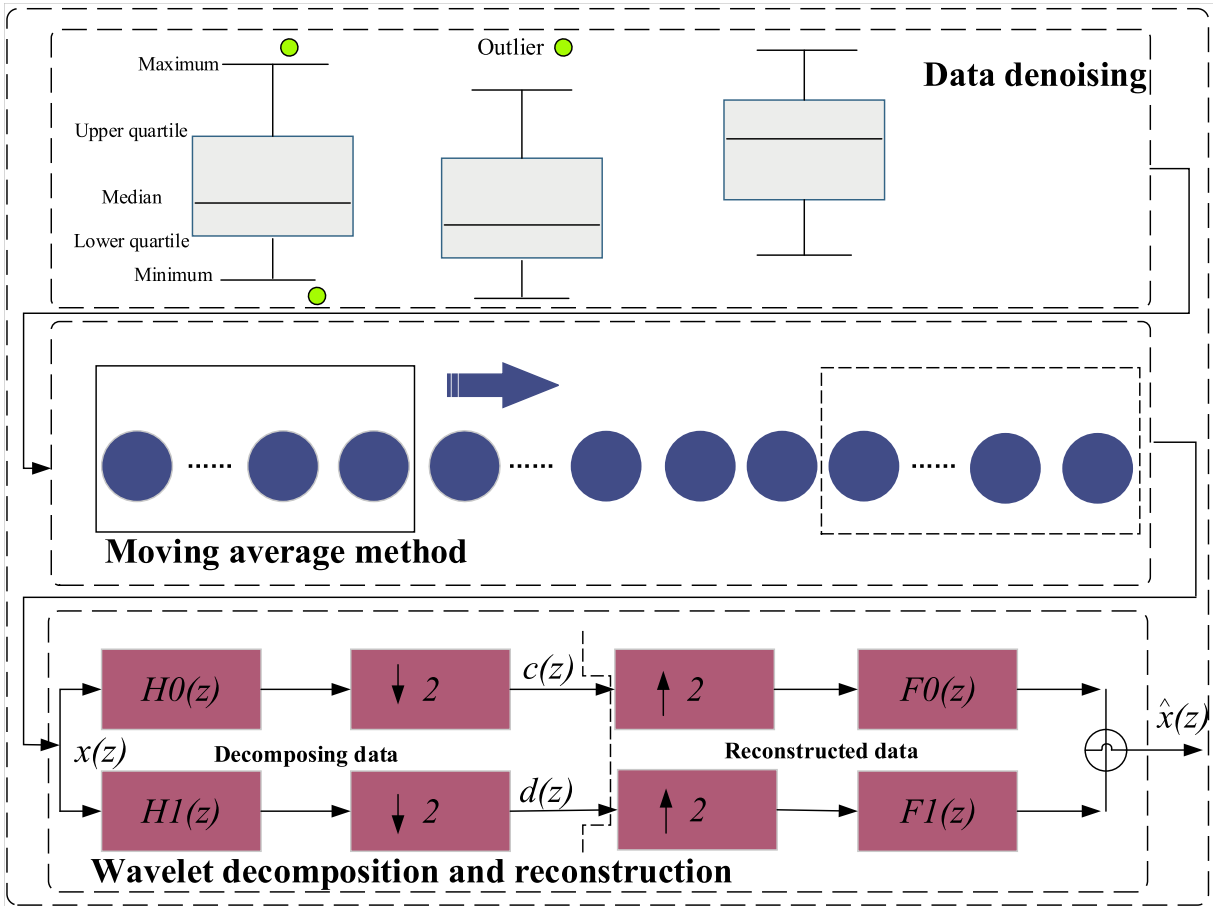
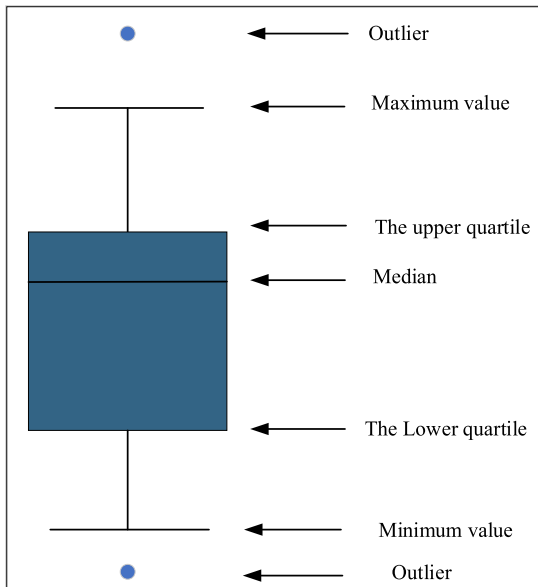**FIGURE 9.** Flowchart of road model construction.

FIGURE 10. Schematic diagram of abnormal data detection of the box-plot.

Box-plot is a method of describing the data using five statistics: upper bound (maximum value in the non-anomalous range), upper quartile, median, lower quartile,

and lower bound (minimum value in the non-anomalous range). In this paper, it is used as an adaptive threshold for determining data outliers. The basic principle of box-plot analysis is to arrange the data from small to large and calculate the quartile of the data. Calculate the outliers of the data through the quartile, and the calculation formula is shown in equations (17) - (18):

$$K \leq L_1 - 1.5(L_3 - L_1) \qquad (17)$$

$$K \geq L_3 + 1.5(L_3 - L_1) \qquad (18)$$

where: K is the abnormal value, that is, the data is in an abnormal state; L_1 is the upper quartile; L_3 is the lower quartile.

2) MOVING AVERAGE WAVELET SMOOTHING ROAD MODEL
Based on the data after outlier cleaning, the moving average method is used to slice the road according to 100m as the minimum unit, that is, taking 100m as an interval, and taking the average value of the elevation information of vehicles in this interval as the elevation data of the starting point of this interval. For example, the average value of the [100,200] interval is used as the elevation information at 100m, and so on. Then, the linear interpolation method is used to interpolate the missing elevation information in the middle of

the data after taking the average value of the moving interval. However, there are local uneven points in the interpolated data, which is inconsistent with the actual road elevation of the highway. Therefore, this paper uses wavelet transform to smooth the above road data.

Wavelet Transform [46] is a commonly used signal processing method that can be applied to smooth data. Wavelet functions have localization characteristics in both the time and frequency domains and can provide information in different time scales and frequency ranges. Utilize the multi-scale analysis characteristics of wavelet transform to reduce noise in data. By selecting appropriate wavelet functions and decomposition levels, we can filter out high-frequency noise components and retain smoothing trends at lower frequencies, thereby achieving data smoothing.

Let $x(t)$ be a square integrable function, denote $x(t) \in L^2(R)$, and the definition of its continuous wavelet transform is shown in equation (19)

$$\mathrm{WT}_x(a, b) = \int x(t)\, \psi_{a,b}(t)dt \qquad (19)$$

In equation (19), $\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a})$ is the base wavelet, $a \neq 0$ is the scaling factor, and b is the translation factor. The scaling factor compresses or extends the wavelet waveform, and the translation factor shifts the wavelet on the t-axis. The process of wavelet decomposition and reconstruction structure is shown in the wavelet transform section of Figure 9.

In the figure, $F_0(z)$ and $F_1(z)$ respectively represent the filtering coefficients corresponding to the low-pass filter and the high-pass filter, while $H_0(z)$ and $H_1(z)$ respectively represent the filtering coefficients corresponding to the mirror filter of the low-pass filter and the high-pass filter, satisfying $H_0(z) = F_0(-z)$ and $H_1(z) = F_1(-z)$. The decomposition process of the signal is as follows: on the one hand, the elevation information x(z) is "downsampled" after passing through a low-pass filter to obtain an average signal c(z) with half the scale and resolution, which is the low-frequency component; On the other hand, x(z) undergoes "downsampling" after passing through a high-pass filter to obtain a detailed signal d(z) with a scale and resolution reduced by half, which is the high-frequency component. The formula for signal decomposition is shown in equations (20) - (21):

$$c_{j+1}(z) = \sum_{m \in z} c_j(m)F_0(m - 2z) \qquad (20)$$

$$d_{j+1}(z) = \sum_{m \in z} c_j(m)F_1(m - 2z) \qquad (21)$$

The process of reconstructing elevation information is as follows: the signal is stretched by inserting zero values between two samples on average, that is, upsampling, and then passing through a low-pass filter to obtain a large-scale low-resolution approximation, that is, low-pass output; After upsampling the detail signal and passing through a

high pass filter, a high pass output can be obtained, and the recon-structed signal x(z) can be obtained by adding the two. The signal reconstruction for-mula is shown in equation (22):

$$c_j(z) = \sum_{m \in z} c_j(m)F_0(z - 2m) + d_{j+1}(m)F_1(z - 2m) \qquad (22)$$

D. VEHICLE POSITION ESTIMATION MODEL BASED ON SPATIO-TEMPORAL SMOOTHING DLCNN-LSTM-ATTENTION WITH L1 REGULARIZATION

Based on multi-dimensional features such as vehicle segment speed, road characteristics, short-term driving style, and the time the vehicle has entered the segment, this section proposes the DLCNN-LSTM-Attention method based on L1 regularization for predicting the position of vehicles on the road [47], [48]. However, the distance between the vehicle and the front door frame in the current segment is positively correlated with time and is in a non-stationary state. Non-stationary state data seriously affects the fitting performance of the model. Therefore, before conducting vehicle position estimation, this article smooths the vehicle position data from both the temporal and spatial dimensions.

1) SMOOTHING MODULE FOR THE TIME DIMENSION OF VEHICLE POSITIONING DATA

The original trajectory of a vehicle is usually composed of a series of trajectory points, which can be represented as $L = \{p_i | i = 1, 2, 3, \ldots, N\}$. Among them, $p_i = (distance_i, t_i)$ is the distance between the vehicle at the time t_i and the front gantry of the current segment. Due to the different sampling intervals and the possibility of losing trajectory points, the trajectory points of the original vehicle trajectory are uneven in terms of time dimension, that is, the positioning time interval is inconsistent and diverse, as shown in Fig 11. Due to unevenness in the time dimension, vehicle position changes can be in a non-stationary state. The non-stationary data is too disorganized, and some even have no rules to follow, making it difficult for the model to capture the patterns of vehicle position changes, which affects the model's fitting effect on the data.
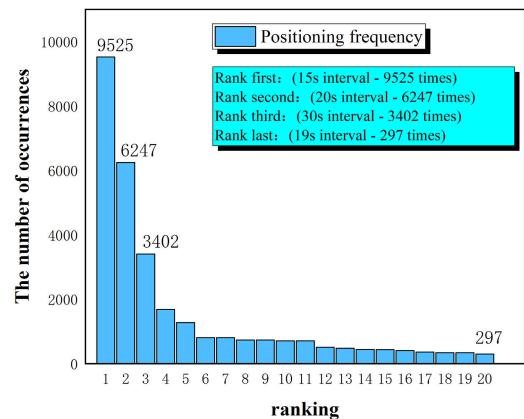

FIGURE 11. Statistical diagram of vehicle positioning time interval.

From Fig 11, it can be seen that the most frequent occurrence of the positioning time interval is 15s, followed by 20s, 30s ranked third, and most of the neighboring points have short positioning time intervals. Considering that the highway road condition is good, the traffic density is small, and the vehicle speed will not change abruptly during the vehicle driving process, except for unexpected situations. Based on this characteristic, this paper cuts the highway segment into multiple subinterval with GPS positioning points as the starting and ending points, and the vehicle's driving speed in the subinterval is regarded as an approximate uniform speed.

As shown in Fig 12, the relationship between the distance of the vehicle from the start of the subinterval and time can be viewed as linear within each subinterval. The positional relationship between p_i and p_{i-1} can be expressed as equation (23). Therefore, in this paper, the trajectory data within $space_i$ are linearly interpolated according to the time change respectively, so that the interpolated vehicle trajectory points are homogenized in the time dimension.

$$p_i = p_{i-1} + (t_i - t_{i-1})v_{i-1} \tag{23}$$

2) SMOOTHING MODULE FOR SPATIAL DIMENSION OF VEHICLE POSITIONING DATA

Since the vehicle is traveling in the segment, its distance from the front gantry of the segment is necessarily positively correlated with its traveling time in the segment, which results in the vehicle position data being in a non-smooth state in the spatial dimension. The usual treatment is to differentiate the data so that the data is in a smooth state. As for the Edata, the difference operation will result in the loss of the initial position p_0 of the vehicle when it enters the segment, which is the only accurately determined vehicle position based on the Edata, i.e., the position of the vehicle when it passes through the gantry. To preserve the information entropy of the Edata to the maximum extent, this paper uses reverse first-order differencing to spatially smooth the data.

After first-order inverse differential smoothing, the distance between the vehicle and the gantry is mapped as the amount of change in vehicle position at adjacent moments $\triangle p$. As shown in Fig 13, the change $\triangle p$ of vehicle position in the period from the previous time to the current time is estimated by the vehicle position at the previous time, vehicle short-term driving style, road characteristics, and vehicle basic driving characteristics, and then p_i is derived from p_{i-1} and $\triangle p$.

3) DLCNN-LSTM-ATTENTION FUSION MODULE BASED ON L1 REGULARIZATION

Based on the spatiotemporal stabilized data, we defined a vehicle position estimation model using DLCNN-LSTM-ATTENTION fusion module based on L1 regularization to achieve vehicle position estimation. The overall structure of the model is shown in Fig 14.

The number of input features to the model is not as high as possible, but needs to be balanced against the complexity of the problem, the characteristics of the data, and the performance of the model. An increase in the dimensionality of the features can cause the samples to become sparse in high-dimensional spaces, and more data may be required to maintain the reliability of the model. In addition, high-dimensional data tends to lead to increased computational complexity. When there are too many features, the model may become overly complex, attempting to fit accurately on the training data, but may have reduced generalization ability on the test data. This results in a model that performs well on training data but may not perform well in real-world applications. On the contrary, choosing the appropriate number and type of features can lead to a more simplified model and improved computational efficiency and generalization performance. For example, in the case of highway traffic flow is small, its multiple speed and vehicle location changes have less impact, when the traffic flow features may increase the complexity and overfitting of the model. Therefore, before regression prediction, this paper uses L1 regularization to select features for the constructed features. L1 regularization realizes the selection and sparsity adjustment of the model features by adding the penalty term of L1 norm, which makes the model more concise and explanatory, and helps to prevent overfitting.

Lasso is a linear regression method that incorporates L1 regularization to obtain the parameters of a classification or regression model by minimizing the empirical error as shown in equation (24)

$$\bar{\beta} = \operatorname{argmin} \|Y - X\beta\| + \lambda \sum_{i=1}^{n} |\beta_i| \tag{24}$$

where X is the feature vector, Y is the response vector, and β is the weight vector of the features.

The regularization term is the L1 norm, and the regularization parameter alpha controls the strength of this regularization term. When the value of alpha is small, the model prefers to keep the coefficients of all features, and the model will have a stronger fitting ability, but overfitting may occur, especially when there are more features. When the value of alpha is large, the model will more strongly drive the coefficients of some features to zero, thus realizing the effect of feature selection and the model will become more sparse. This helps to reduce the complexity of the model and improve the generalization ability, but it may also lose some predictive performance of the model. Therefore, the regularization parameter is crucial for feature selection, and the selection of the regularization parameter is not scientifically sound when realized only by experience. To achieve reasonable feature selection, this paper uses cross-validation to adjust the values of regularization parameters in the Lasso model.

First, we define a two-layer CNN layer containing convolution and pooling operations for extracting local features from the input data. We used multiple convolutional kernels and ReLU activation functions to capture nonlinear relationships in the input data, as shown in equation (25). By maximizing
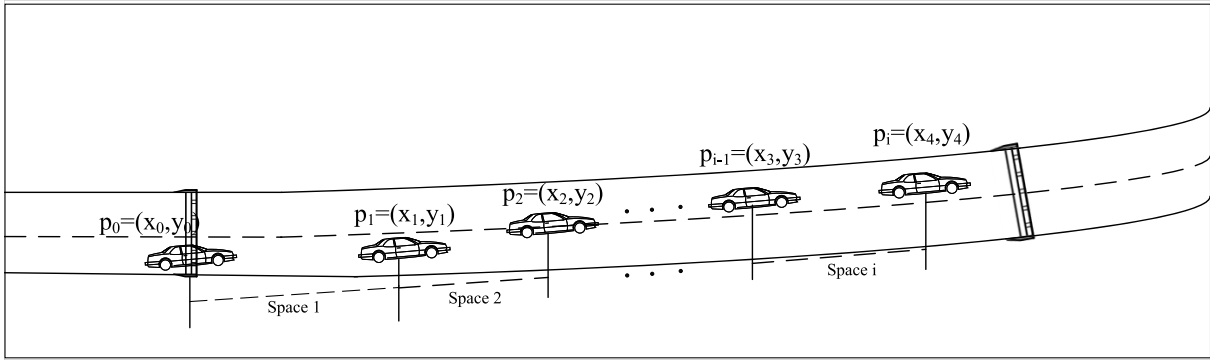
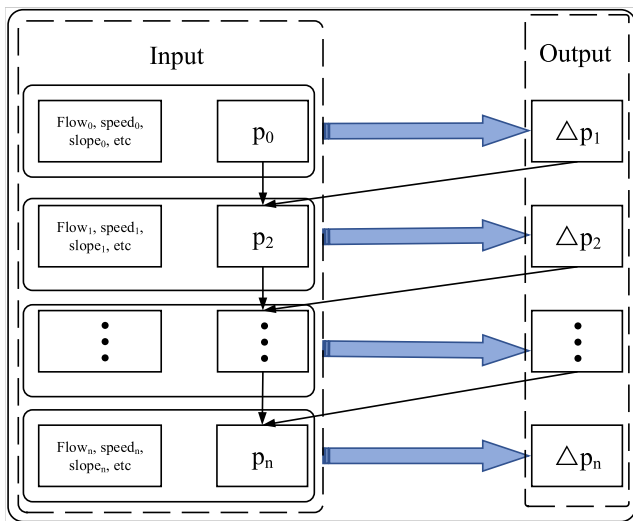**FIGURE 12.** Schematic diagram of vehicle position change.

FIGURE 13. Schematic diagram of vehicle position prediction process.

the pooling operation, we reduced the dimensionality of the features to reduce the impact of non-essential features on the model fitting effectiveness.

$$\text{RELU} = \begin{cases} x, & x > 0 \\ 0, & x < 0 \end{cases} \tag{25}$$

Double-layer CNN can simultaneously process multiple input features, such as vehicle speed, traffic flow, and vehicle type. The first layer convolution can detect the local pattern of each feature and capture the relevant information of different features. The second layer convolution can fuse and abstract these multimodal features, to better understand the multidimensional relationship of data. In addition, double-layer CNN can capture the spatio-temporal relationship. The first layer convolution can extract local features of data in time and space, while the second layer convolution can further Abstract these spatio-temporal patterns. There may be complex interrelationships among vehicle speed, traffic flow, and vehicle type. For example, the vehicle speed at a certain time may be related to traffic flow and vehicle type.
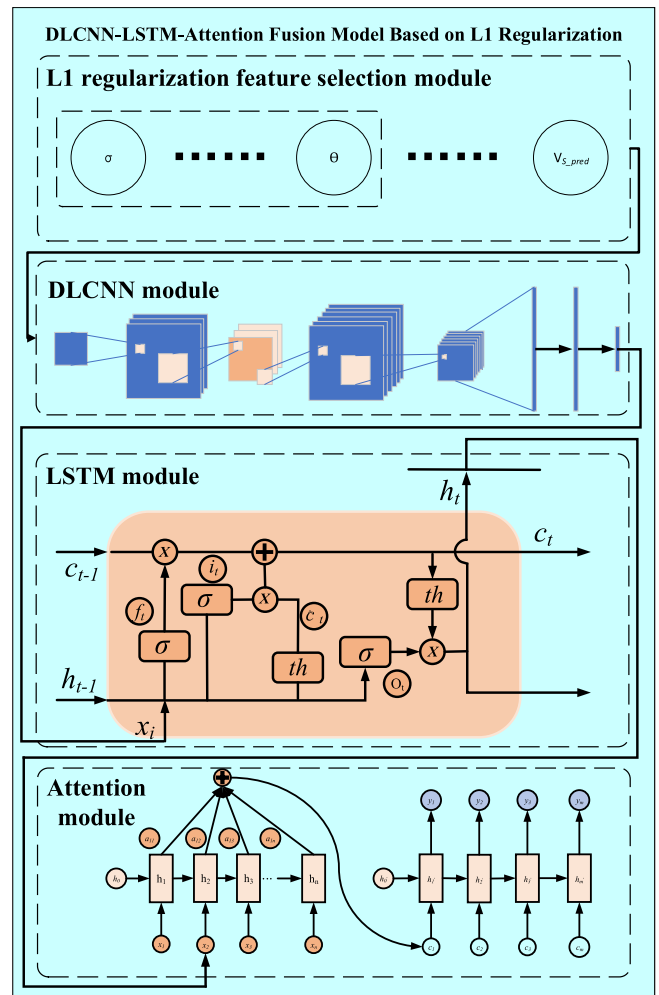

FIGURE 14. Schematic structure of DLCNN-LSTM-ATTENTION model based on L1 regularization optimization.

Double-layer CNN can interact and fuse different features in the second-layer convolution, to capture higher-level data patterns.

Next, we introduce an LSTM layer for processing sequence data with temporal dependencies [49]. The LSTM layer has

memory cells and gating mechanisms to better capture and utilize long-term dependencies in sequence data. We use 64 hidden cells and set them to return the complete output sequence to preserve the information of the sequence. The details of each computation of LSTM are as follows:

The forgetting gate can be described as shown in equation (26).

$$f_t = \text{sigmoid}(w_f[h_t, x_t] + b_f) \tag{26}$$

Here f_t represents the amount of information passed.

The input door can be described as equations (27) - (28).

$$i_t = \text{sigmoid}(w_i[h_{t-1}, x_t] + b_i) \tag{27}$$

$$\tilde{C}_t = \tanh(w_c[h_{t-1}, x_t] + b_c) \tag{28}$$

Cell state (information transmission) C_t is shown in equation (29).

$$C_t = C_{t-1}f_t + i_t C_t \tag{29}$$

The output gate O_t is shown in equations (30) - (31).

$$O_t = \text{sigmoid}(w_0[h_{t-1}, x_t] + b_o) \tag{30}$$

$$h_t = O_t * \tanh(C_t) \tag{31}$$

In equations (25) - (30), w_f, w_i, w_c, w_o are all weight indicators, and b_f, b_i, b_c, b_o are all offset.

Compared with the RNN model, LSTM adds several gate settings, especially the forgetting gate, which can filter the input information of the previous period, to retain the key information and forget part of the unimportant information, which is the key that LSTM can overcome the gradient disappearance. However, LSTM still has the phenomenon of losing important data information when the input sequence is too long, so it needs CNN to process the original data and filter out part of the unimportant information, to improve the accuracy of prediction.

To further improve the model performance, we introduce an attention mechanism to enhance the attention to the input sequence. By applying the attention mechanism on the output of the LSTM layer, we can adaptively learn the importance weights for each time step. By generating query vectors and value vectors and computing the attention scores between them, we can obtain the attention weights. Then, by weighting and summing the attention weights with the value vectors, we obtain a context vector that weights and averages the value vectors of different time steps to capture the information of the time steps with higher attention.

Finally, to synthesize the information from the output of the LSTM layer and the context vectors, we concatenate them. Through the Concatenate layer, we will obtain a synthesized feature vector that contains the raw output of the LSTM layer as well as the key information from the attention mechanism. This synthesized feature vector is spread into a one-dimensional vector and passed through a fully connected layer for final regression prediction. The model is trained using the Adam optimizer using the mean square error as the loss function.

The attention mechanism will calculate the attention weight of each input part according to the change rule of vehicle position combined with the current input and task requirements, i.e., to determine the importance of each part for the current task, and to weigh different parts, so that the model can pay more attention to the important information. Thus, the information related to the current task is strengthened. This can help the model to pay better attention to important features and improve the performance of the model when dealing with complex tasks. In addition, this dynamic weight allocation mechanism enables the model to capture the features related to the task more accurately, thus improving the model's performance and generalization ability. Thus, by combining these three models, the DLCNN-LSTM-ATTENTION model can better understand the data and improve prediction accuracy.

V. EXPERIMENTAL ANALYSIS

A. DESCRIPTION OF EXPERIMENTAL MATERIAL

The experimental data used the ETC transaction data of Fujian Provincial Expressway from September 3 to September 5, 2020, and the Gdata of two passengers and one hazardous vehicle. Edata contains information such as vehicle number, transaction time, etc., as shown in Table 2. The two-passenger and one hazardous data contain information such as real-time vehicle speed, longitude, latitude, and direction angle of the vehicle, as shown in Table 3.

TABLE 2. ETC transaction data attribute table.

Attribute Name	Examples
TRADEID	$S0 * * * 1$
TRADETIME	$2020/9/3\,20:00:01$
FLAGID	$3502 * *$
FLAG TYPE	0
OBUID	Blue Fujian A12345
ENTIME	$2020/9/3\,7:48:39$
ENSTATION	$46**$
LNG	$118.56**$
VEHCLASS	1

TABLE 3. GPS data attribute table.

Attribute Name	Examples
Positioning time	20200903000017
Reporting time	20200903000018
Speed (km/h)	95km/h
Mileage (km)	$118.56**$
Plate	Fujian A12345 Blue
Sensor speed	95km/h
Elevation (m)	43(m)
LNG	$118.56**$
LAT	$24.85***$
Direction angle (°)	180°

To verify the generalization ability of the model under different road features, the experimental road segments were selected as G3 Ningde Gutian to Fuzhou Minhou segment,

G15 Fuqing Jiangyin Port to Putian segment, and G76 Fujian Zhangzhou to Longyan segment. Among them, the coastal segment is the segment from Jiangyin Port to Putian, the mountainous segment is the segment from Ningde Gutian to Fuzhou Minhou, and the segment from Zhangzhou to Longyan, Fujian. The three segments are all segments of the same type with large traffic flow and can reflect the full sample traffic conditions of the highway. The experimental segment is shown in Fig 15.

B. SETTING OF NETWORK PARAMETERS AND SELECTION OF EVALUATION INDICATORS

1) THE SETTING OF NETWORK PARAMETERS

In the comparison experiments of different prediction models, the choice of network parameters largely determines the performance of the algorithm. In our study, the parameters involved in each experiment are chosen according to the actual situation of this study by referring to a large number of literature or using empirical and trial-and-error methods. The network parameter settings for each model in this method are shown in Table 4.

2) SELECTION OF EVALUATION INDICATORS

Appropriate evaluation indexes can accurately and intuitively reflect the prediction effect of the model and compare the predicted value of the model with the real value of the dataset to quantify the performance of the model. The evaluation metrics are selected as mean absolute error (MAE), root mean square error (RMSE), Mean Absolute Percentage Error (MAPE), and (coefficient of determination) R^2. Relative to the mean error, the mean absolute error is absolutized without positive or negative cancellation, thus better reflecting the true picture of the prediction error; RMSE reflects the data dispersion; MAPE (Mean Absolute Percentage Error) is a measure of the relative error between the predicted value and the actual value. It is used to evaluate the performance of prediction models in prediction problems, especially in scenarios involving percentage error. To improve the interpretability of the model evaluation metric and remove the influence of the size of the test set values itself, R^2 is introduced to judge the superiority of the prediction model, avoiding the influence of the degrees of freedom on the prediction results. The larger the R^2 is, the better the model fit is indicated, and its value range is [0,1]. Its calculation formula is shown in Table 5.

C. CONSTRUCTION OF SHORT-TERM DRIVING STYLE FEATURES

To more accurately capture the change rule of vehicle position, this paper first classifies the short-term driving style of vehicles according to the driving characteristics of vehicles in the historical segments during the current trip, and the construction results are shown in Fig 16(a) to Fig 16(d). In each segment, the model subdivided the driving style of the vehicle into different categories according to the traffic

flow, speed, and vehicle type. Among them, in segment 1, the model classifies the target vehicles into 4 categories based on each characteristic variable when the vehicles are traveling; in segment 2, the target vehicles are classified into 2; in segment 3, the target vehicles are classified into 2 categories; and in segment 4, the target vehicles are classified into 10 categories.

It can be seen that there are some differences in the driving characteristics of vehicles in each category in each segment. For example, as shown in Fig 16(b), category 1 vehicles indicate the passing speed characteristics of vehicles when the traffic flow is large; category 0 can be seen as the driving characteristics of vehicles when the traffic flow is small. Therefore, the short-term driving style reflects, to a certain extent, the driving pattern of vehicles in different driving environments, which provides an important estimation basis for the subsequent vehicle segment speed prediction and vehicle position estimation.

D. EXPERIMENTAL ANALYSIS OF VEHICLE SEGMENT SPEED CHARACTERIZATION CONSTRUCTION

The inputs to the model contain σ, V_{S-1}, V_{S-2}, V_{EN}, V_{EN1}, V_{EN2}, F_S, F_{S-1}, F_{S-2}, T, the outputs are the Speed of the vehicle in the current segment. Secondly, the continuous variables in the input data of the model are normalized to the maximum and minimum values to improve the performance stability of the model. Continuous variables include V_{S-1}, V_{S-2}, V_{EN}, V_{EN1}, V_{EN2}, F_S, F_{S-1}, F_{S-2}.

As shown in Fig 17(a) to Fig 17(d), our predicted velocity trends in each segment are generally consistent with the actual velocity trends. In segment 1, the average absolute error, root mean square error, correlation coefficient, and average absolute error percentage of our prediction results are 2.83%, 12.93%, 0.89%, and 12.77%, respectively. In segment 2, the average absolute error, root mean square error, correlation coefficient, and average absolute error percentage of our prediction results are 2.30%, 9.48%, 0.85%, and 11.88%, respectively. In segment 3, the average absolute error, root mean square error, correlation coefficient, and average absolute error percentage of our prediction results are 2.98%, 13.30%, 0.84%, and 12.43%, respectively. In segment 4, the average absolute error, root mean square error, model fitting degree, and average absolute error percentage of our prediction results are 3.08%, 17.52%, 0.78, and 11.30%, respectively. From the predicted performance indicators, vehicle owners will drive their vehicles more based on their driving habits when the road traffic condition is good, and our model takes into account the short-term driving style of vehicle owners, which can better capture the dynamic change of the speed pattern.

Secondly, the proposed model is subjected to ablation experiments regarding short-term driving style, and the results of the experiments are shown in Table 6. Without considering the short-term driving style, the mean absolute error, root mean square error, correlation coefficient, and mean absolute error percentage of our model are 3.19,
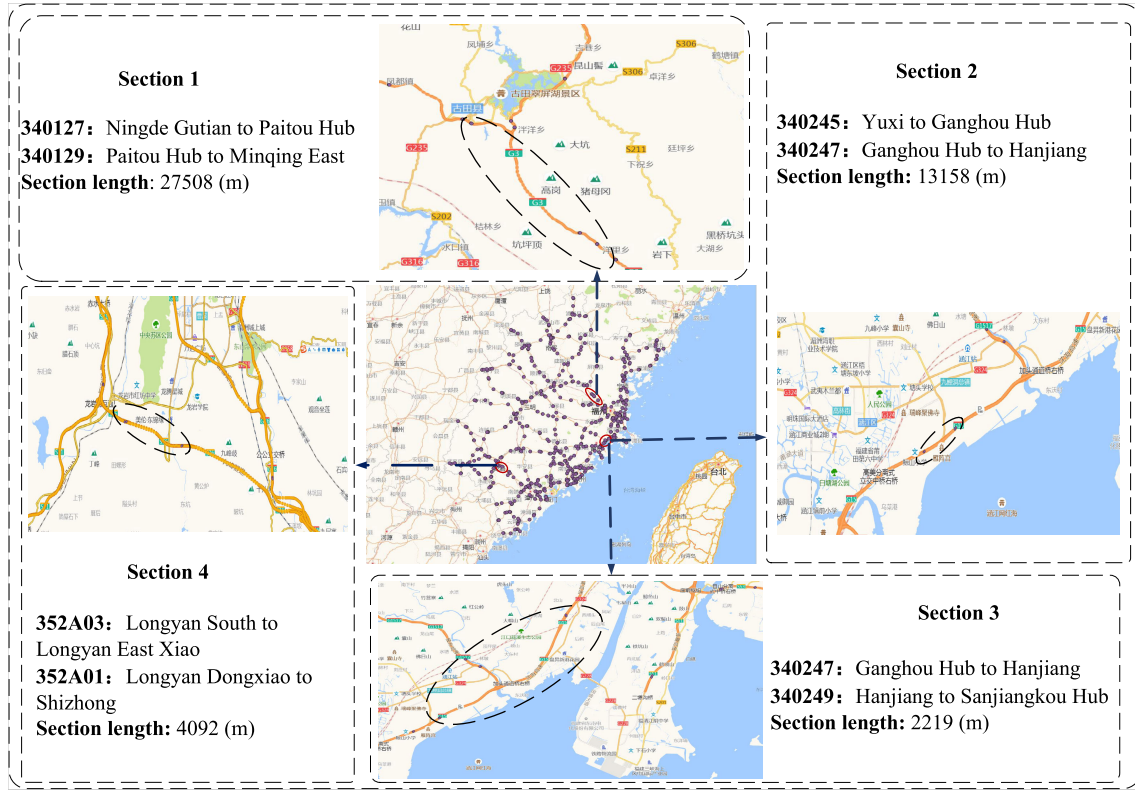
FIGURE 15. Experimental road schematic diagram.

TABLE 4. Model parameter setting table.

MODEL	Experimental parameters	values
Speed prediction model	Layers	3
	No. of neurons	Bilstm layer 1: 64 neurons; Bilstm layer 2: 64 neurons; Fully connected layer: 1 neuron
	Learning rate	0.001
	Optimizer	Adam
	Batch size	300
	Epochs	16
Road model	Smoothness coefficient	100
DLCNN-LSTM-ATTENTION	Layers	7
	No.of neurons	CNN Layer 1: 32 neurons; CNN Layer 2: 32 neurons; LSTM layer: 64 hidden units; Full connectivity layer: 1 neuron
	Learning rate	0.001
	Optimizer	Adam
	Batch size	32
	Epochs	200

TABLE 5. Model evaluation indicators.

Metric	Equation	Definition
MAE	$MAE = \frac{1}{N}\sum_{n=1}^{N}|y_n - \hat{y}_n|$	The average absolute forecast error of n times forecast results
RMSE	$RMSE = \left(\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2\right)^{1/2}$	The root-mean-square forecast error
MAPE	$MAPE = \frac{1}{N}\sum_{n=1}^{N}\left|\frac{y_n - \hat{y}_n}{y_n}\right| \times 100\%$	The percentage of average absolute error
R^2	$R^2 = \frac{\sum_{n=1}^{N}(\hat{y}_n - \bar{y})^2}{\sum_{n=1}^{N}(y_n - \bar{y})^2}$	The degree of fit of the regression line to the observed values

15.27, 0.85, and 15.72%, respectively, in segment 1; in segment 2, the mean absolute error, root mean square error, correlation coefficient, and mean absolute error percentage of our model are 2.05, 8.67, 0.86, and 11.62%; in segment 3, the mean abso-lute error, root mean square error, correlation coefficient, and mean absolute error percentage of our model were 3.23, 13.54, 0.84, and 13.72%; in segment 4, the mean absolute error, root mean square error, correlation coefficient, and mean absolute error percentage of our model were 3.31, 18.80, 0.77, and 11.40%, respectively.
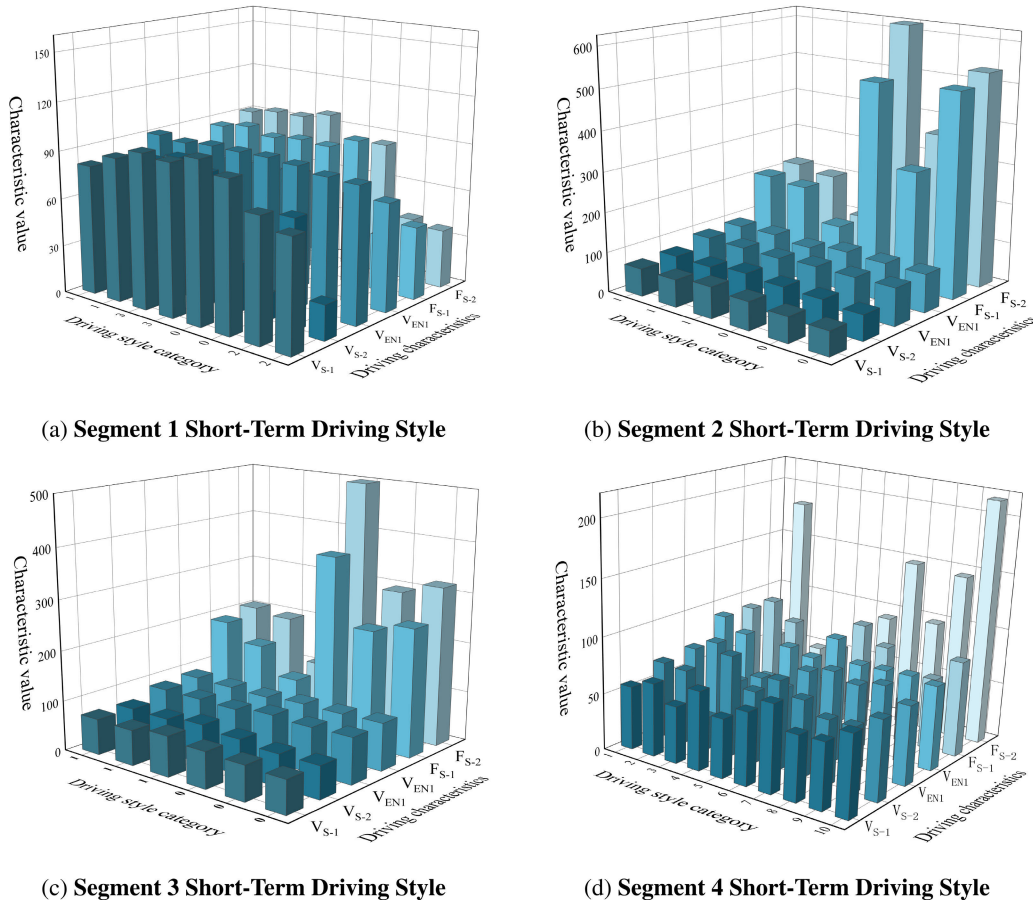
(a) **Segment 1 Short-Term Driving Style**

(b) **Segment 2 Short-Term Driving Style**

(c) **Segment 3 Short-Term Driving Style**

(d) **Segment 4 Short-Term Driving Style**

FIGURE 16. Short-term driving style classification chart.

In the experiments of segment 1, segment 3, and segment 4, the performance of the model without considering the short-term driving style decreased to a certain extent compared with that considering the short-term driving style. In segment 2, the performance of the model without considering the short-term driving style is slightly better than that of the model considering the short-term driving style, but the improvement is small. According to the overall comparative experiment of the four segments, taking short-term driving style into account in the model helps explain the degree of data variation and accurately obtain the predicted target value.

E. CONSTRUCTION OF ROAD SLOPE CHARACTERISTICS

The road structure constructed by the road model based on moving average-wavelet smoothing for anomaly detection of the box-and-line diagram is shown in Fig 18. Among them, Fig 18(a), 18(c), 18(e), and 18(g) show the original road elevation information in Gdata data, and Fig 18(b), 18(d), 18(f), and 18(h) show the road structure after moving average-wavelet smoothing. As can be seen from the figures, the original road elevation information is disorganized,

TABLE 6. Effect table of vehicle speed prediction model ablation experiment.

Indicator	Not considering short-term driving style	Our model
segment 1		
MAE	3.19	2.83
RMSE	15.27	12.93
MAPE	15.72%	12.77%
R^2	0.85	0.89
segment 2		
MAE	2.05	2.30
RMSE	8.67	9.48
MAPE	11.62%	11.88%
R^2	0.86	0.85
segment 3		
MAE	3.23	2.98
RMSE	13.54	13.30
MAPE	13.72%	12.43%
R^2	0.84	0.84
segment 4		
MAE	3.31	3.08
RMSE	18.80	17.52
MAPE	11.40%	11.30%
R^2	0.77	0.78

and the heights under the same geographic location are uneven, which makes it impossible to obtain an effective road structure. The road model constructed by moving
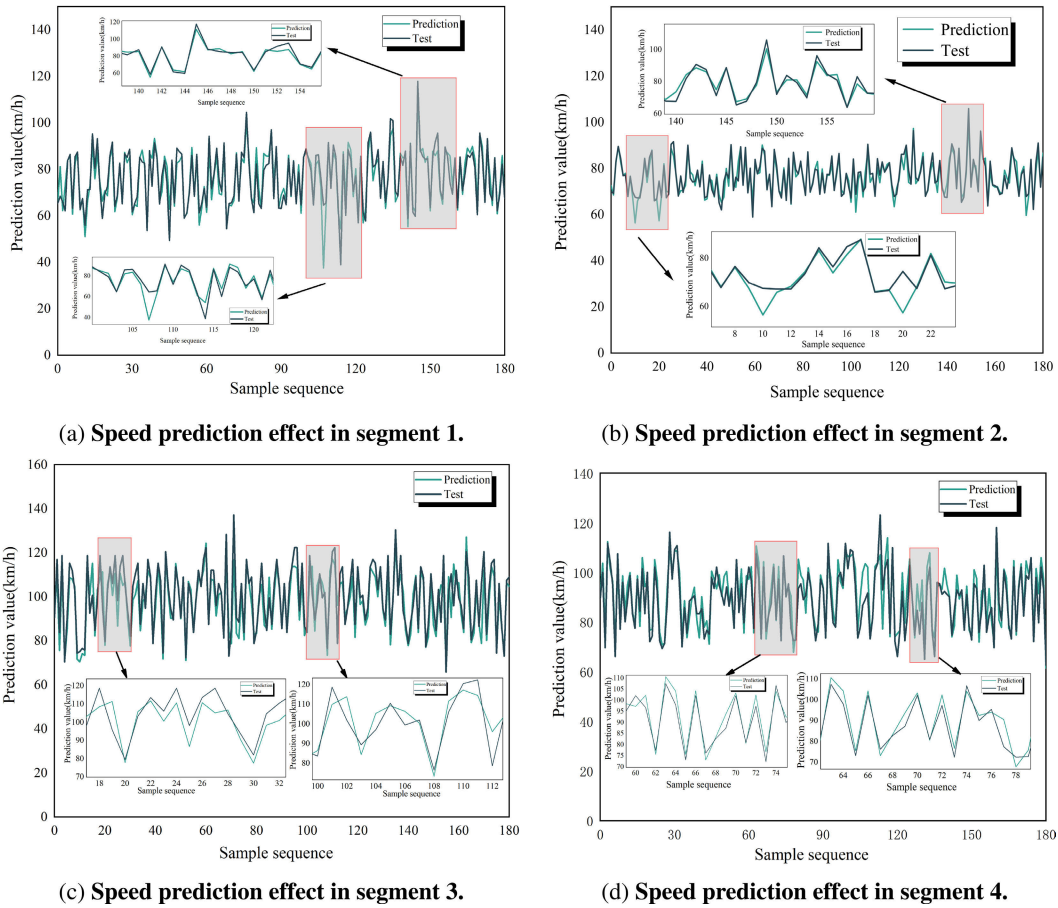
(a) **Speed prediction effect in segment 1.**

(b) **Speed prediction effect in segment 2.**

(c) **Speed prediction effect in segment 3.**

(d) **Speed prediction effect in segment 4.**

FIGURE 17. Effect diagram of vehicle speed prediction.

average-wavelet transform is similar to the road structure stacked with elevation information in the original data in terms of contour, which proves the similarity between the constructed slope and the actual slope. In this paper, the main consideration for road features is the effect of slope. Therefore, only the slope of the constructed road model is concerned.

Since the vehicle elevation information is more cluttered and there are trajectory drift points, if the GPS trajectory drift points are too concentrated in a small interval in the segment, these drift points will be eliminated, resulting in sparse elevation information in some of the intervals. This may lead to a large gap between the slope derived from the model and the actual slope. To avoid the influence of a few slope distortions caused by data sparsity, this paper will obtain the slope value for the maximum and minimum normalization, the uphill segment is categorized as between [0,1], and the downhill segment is categorized as between [−1,0], to avoid the influence of the abnormal slope to the maximum extent, and to improve the antinoise ability of the model. Finally, the effectiveness of the slope features constructed by this method is also further

verified in the subsequent vehicle position estimation ablation experiments.

F. EXPERIMENTAL ANALYSIS OF FEATURE SELECTION

To improve the efficiency of the model and reduce phenomena such as overfitting. Feature selection is performed using L1 regularization based on predicted vehicle segment speeds, roadway characteristics, and vehicle-based driving characteristics. The results of the L1 regularization feature selection for each segment are shown in Table 7. In segment 1, five features are finally selected as inputs to the model, which are V_{S_pred}, F_S, T, $P_{previous}$, and Θ; in segment 2, all features are selected as inputs to the model; in segment 3, six features are finally selected, which are V_{S_pred}, F_S, F_{S-1}, F_{S-2}, $P_{previous}$, Θ; in segment 4, there are seven features finally selected, which are V_{S_pred}, V_{S-1}, V_{EN2}, F_S, $P_{previous}$, $P_{previous}$, Θ.

Lasso tends to produce a sparse model, i.e., only a small number of β_i are non-zero, and these small number of features with non-zero weights are extracted as important for classification. Thus Lasso can explicitly extract features,
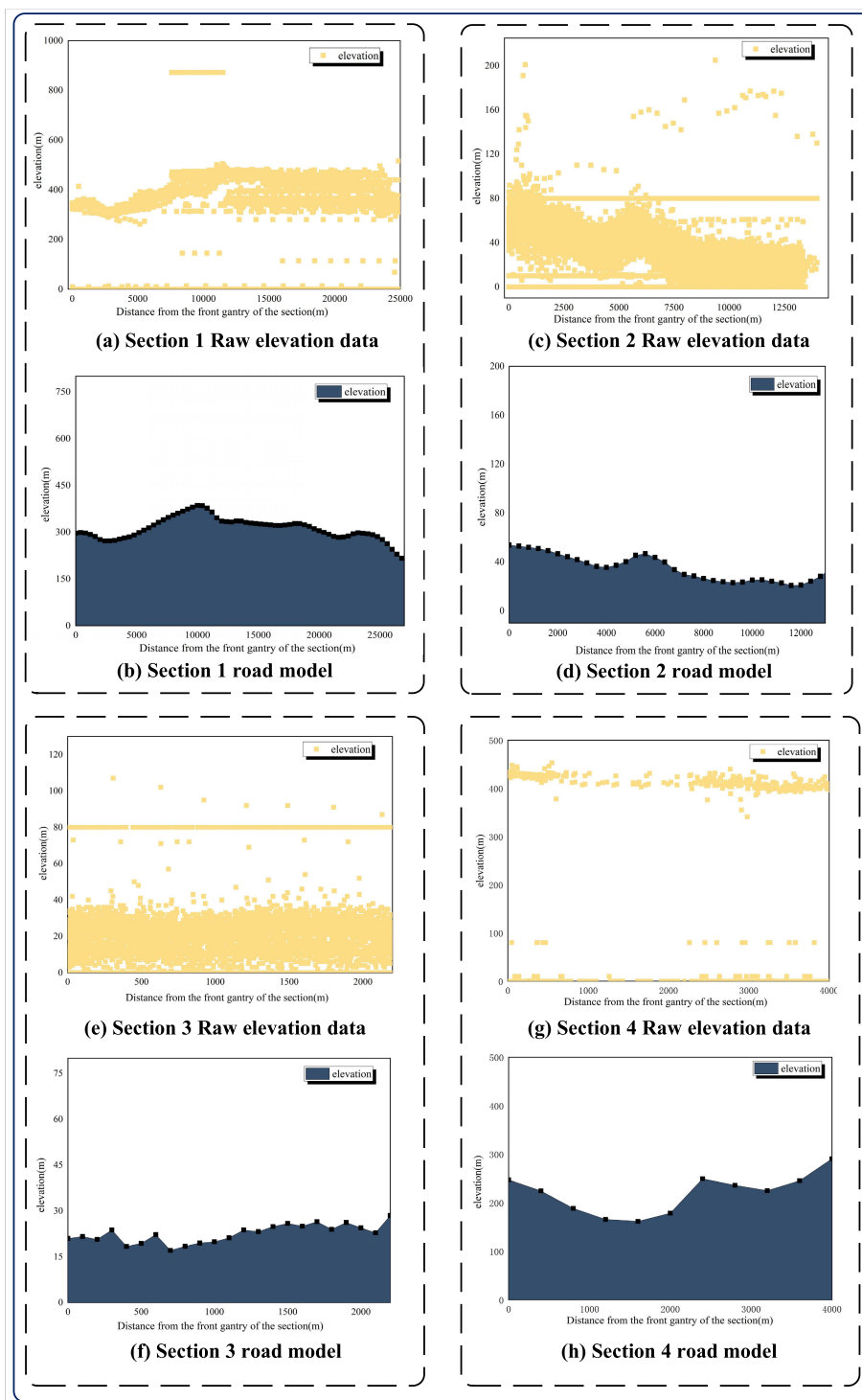
FIGURE 18. Road structure diagram.

i.e., those feature terms with non-zero weights, while constructing a classification model. However, in segment 2, all the features proposed in this paper are input into the speed prediction model. The reason for this phenomenon may be that segment 2 is located in the port, and the overall traffic flow is large, which leads to the weak correlation between the features of this segment and the corresponding features of the segment before this segment, resulting in L1 regularization selecting all features into the model.

TABLE 7. Table of L1 regularization feature selection results.

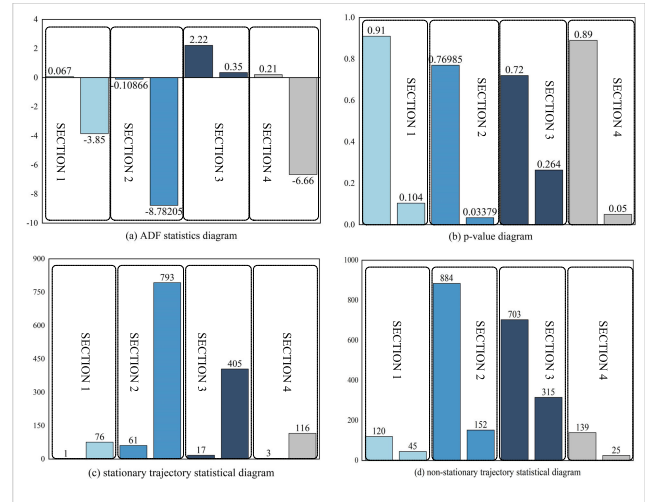| characteristic	Segment 1	Segment 2	Segment 3	Segment 4
V_{S_pred}	✓	✓	✓	✓
V_{S-1}		✓		✓
V_{S-2}		✓		
V_{EN}		✓		
V_{EN1}		✓		
V_{EN2}		✓		✓
F_S	✓	✓	✓	✓
F_{S-1}		✓	✓	
F_{S-2}		✓	✓	✓
T	✓	✓		
$P_{previous}$	✓	✓	✓	✓
Θ	✓	✓	✓	✓

FIGURE 19. Smoothing effect of vehicle position data processing.

G. EXPERIMENTAL ANALYSIS OF SPATIO-TEMPORAL DATA STABILIZATION

To verify whether the data after spatiotemporal stabilization meets the stability requirements, the ADF test is performed on the data. The results of the test are shown in Fig 19, where each subplot has four parts corresponding to the indicator values of the four segments. The left side of each section shows the values of each indicator obtained from the original data and the right side shows the values of each indicator obtained from the smoothed data. As can be seen from Fig 19(a), after smoothing the data, the ADF statistics of segment 1, segment 2, and segment 4 are smaller than the critical values, and the ADF statistics of segment 3 are also substantially reduced from the original data. As can be seen from Fig 19(b), the p-value value of the data after the smoothing process is significantly reduced and the p-value values of segment 2 and segment 4 are less than the level of significance, which indicates that the data of segment 2 and segment 3 have been smoothed after the process. While the p-value values of segment 1 and segment 3 are greater than the significance level, the number of their steady-state trajectories has been substantially increased after the smoothing treatment as shown in Fig 19(c) and Fig 19(d). To retain the information entropy of the data to the greatest extent, the data are not further smoothed in this paper.

H. MODEL PREDICTION PERFORMANCE COMPARISON AND SPATIAL DIMENSION GENERALIZATION ABILITY ASSESSMENT

This experiment aims to validate the prediction performance of the model. In this experiment, CNN, bilstm, DNN, CNN-ATTENTION, LSTM-ATTENTION, and TGCN are used to compare with the proposed model, DLCNN-LSTM-ATTENTION, and the recursive prediction distance is uniformly set to 2km, and the vehicle enters the target roadway with the front doorframe of the vehicle as the initial node, and the vehicle's position at the next moment is predicted recursively position of the vehicle at the next moment. The results of the model's prediction in each

segment are shown in Fig 20(a) to Fig 20(d) in comparison with the rest of the prediction models.

From Table 8, it can be seen that the values of MAE, RMSE, MAPE, and correlation coefficient R^2 for the DLCNN-LSTM-ATTENTION prediction method used in this paper in segment 1 are 25.32,42.15,5.05%,0.99 respectively, which are the best values for all the prediction models. In segment 2, the MAPE value of this model is 8.26%, which is 4.49%, 2.93%, 1.92%, 14.88%, and 0.78% lower than the other five prediction models, respectively. This model has the highest prediction accuracy and the best prediction effect. In segment 3, the values of MAE, RMSE, MAPE and R^2 of the models used in this paper are 41.35,65.53,9.38%, and 0.98 respectively, which also show the best performance among all models. In segment 4, the MAE value of this model is 40.82, which is smaller than the MAE values of CNN, Bi-lstm, CNN-ATTENTION, and LSTM-ATTENTION by 13.97, 17.12, 6.24, and 46.47, respectively, but the MAE value is significantly higher by 6.09 when compared to the DNN. The DLCNN-LSTM-ATTENTION model is more complex than the DNN model with more layers and parameters to better fit the data and capture its complex relationships. However, in some cases, model complexity can lead to overfitting, especially with less training data. Thus, in some cases, DNN models may perform better due to their relatively simpler structure.

From the correlation coefficient R^2 of the four road segments, the R^2 of our proposed model in each road segment reaches 0.99, indicating that the model in the highway full-sample traffic environment, a better fit to the pattern of change in the position of the vehicle, and the speed of the highway vehicle is more stable, less sudden changes in the phenomenon of the change in the law of the position of the model is better captured.

From the spatial dimension, the four segments are located in different directions in Fujian Province, from north to south
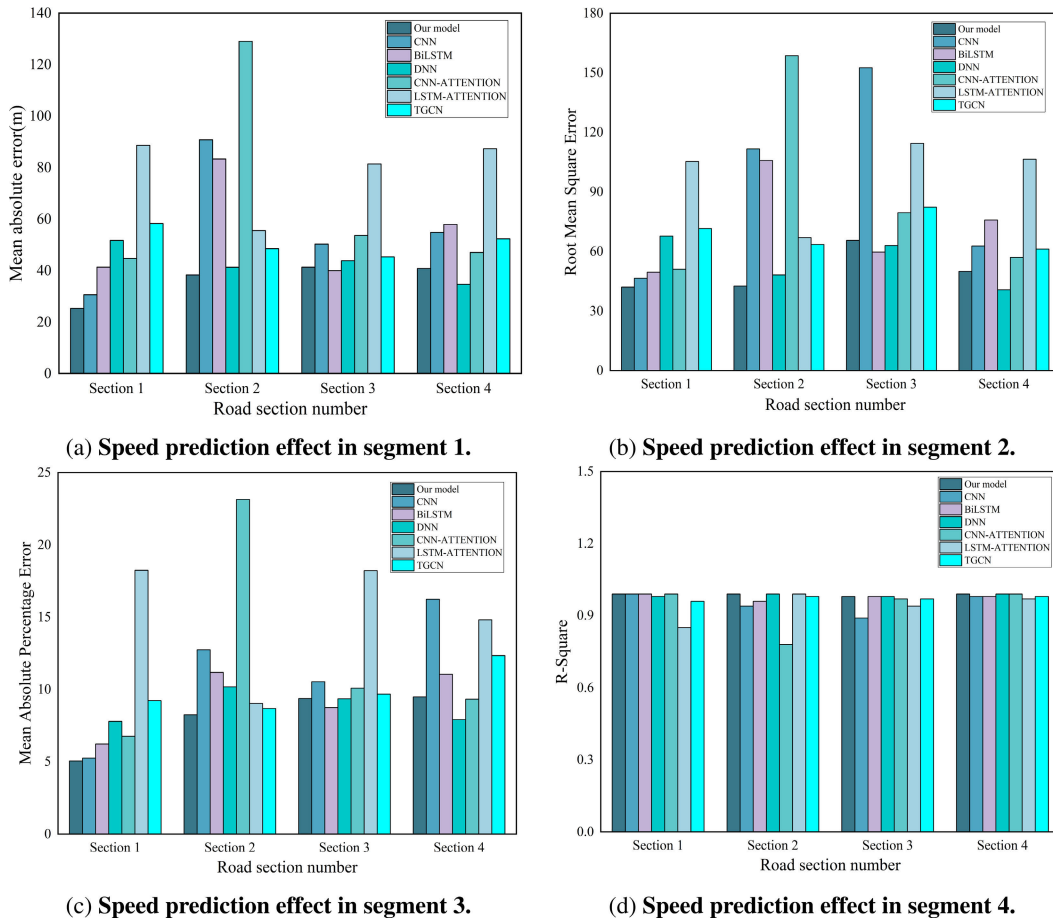
(a) **Speed prediction effect in segment 1.**

(b) **Speed prediction effect in segment 2.**

(c) **Speed prediction effect in segment 3.**

(d) **Speed prediction effect in segment 4.**

FIGURE 20. Experimental diagram of vehicle position estimation.

TABLE 8. Model performance comparison table.

Indicators	CNN	Bilstm	DNN	CNN-Attention	LSTM-Attention	TGCN	Our model
segment 1							
MAE	30.68	41.36	51.69	44.72	88.65	58.32	25.32
RMSE	46.59	49.59	67.71	51.07	105.36	71.51	42.15
MAPE	5.24%	6.23%	7.79%	6.76%	18.25%	9.23%	5.05%
R^2	0.99	0.99	0.98	0.99	0.85	0.96	0.99
segment 2							
MAE	90.76	83.33	41.33	129	55.48	48.55	38.35
RMSE	111.67	105.88	48.25	158.59	66.94	63.52	42.68
MAPE	12.75%	11.19%	10.18%	23.14%	9.04%	8.69%	8.26%
R^2	0.94	0.96	0.99	0.75	0.99	0.98	0.99
segment 3							
MAE	50.28	40.01	43.86	53.61	81.44	45.32	41.35
RMSE	152.57	59.72	62.97	79.6	114.43	82.31	65.53
MAPE	10.55%	8.75%	9.36%	10.09%	18.23%	9.68%	9.38%
R^2	0.89	0.98	0.98	0.97	0.94	0.97	0.98
segment 4							
MAE	54.79	57.94	34.73	47.06	87.29	52.35	40.82
RMSE	62.74	75.89	40.79	57.1	106.46	61.22	50.03
MAPE	16.24%	11.06%	7.92%	9.33%	14.81%	12.35%	9.49%
R^2	0.98	0.98	0.99	0.99	0.97	0.98	0.99

and from east to west; the overall altitude ranges from tens of meters to hundreds of meters. In four different segments under the 2km recursive prediction, this paper proposed model and vehicle position change law of the fit are above 0.98, the average absolute error is less than 50, and MAPE is 10% or less, indicating that the model in the highway

full sample spatial environment can achieve a high degree of fitting to the vehicle position, the spatial generalization ability is better.

I. ABILITY TO GENERALIZE THE TIME DIMENSION

The traffic flow of the highway changes regularly with time, and the difference between the peak period and idle time is large, When the traffic flow reaches a certain level, it will have a limiting effect on the speed of the car. To verify the performance of the model in different driving environments, this paper regards 8:00 a.m. to 10:00 a.m. and 4:00 p.m. to 6:00 p.m. as the peak period of highway travel, and 5:00 a.m. to 7:00 a.m. and 7:00 p.m. to 9:00 p.m. as the idle period, and examines the model's performance by time.

As can be seen from Fig 21(a) to Fig 21(d), during the peak period, the average absolute error of the model is larger than that of the idle period, and the RMSE and MAPE are larger than that of the idle period. This is because vehicles traveling in the peak period, high traffic flow on the highway, will inevitably cause some restrictions on the speed of the vehicle, and at this time travel more vehicles. When the traffic flow is saturated, normal traveling vehicles encounter low-speed vehicles, and the side of the vehicle behind the vehicle to follow the car close, may not be able to change lanes in time, this time is bound to need to slow down. This also leads to the highway in which the traffic flow saturation speed fluctuation is large, and the vehicle position change is not smooth, thus reducing the predictive effect of the model.

The R^2 value of the model is greater than the peak period in the idle time segment, which means that the model fits better in the idle time segment than in the peak period. This is because the vehicle travels in the idle time segment, the traffic flow on the highway is small, and the highway has no substantial curves, The owner can drive steadily at a smooth speed, and the highway are one-way multi-lane road when the flow is small, the owner can change lanes ahead of time when vehicle owner meets a low-speed vehicle to reduce the dilemma of limiting the speed due to the conflict of driving styles, so that he can maintain a stable speed. Another reason for the vehicle to maintain a stable speed is that the highway speed is faster, sudden changes in speed can easily lead to accidents such as rear-end collisions, and emergency braking has a big impact on the occupants, so the vehicle in the highway traffic conditions are good, more to maintain a stable speed, which means that the vehicle position change is in a smooth state so that the model can be a better fit to the data.

J. MULTI-STEP ESTIMATION CAPACITY ASSESSMENT

This experiment aims to test the multi-step estimation capability of the proposed model. The evaluation results are shown in Table 9. Among the estimation distances from 2km to 5km, the MAE value at 2km is the smallest, and the MAE value at 5km is the largest. Overall, the MAE value increases with the increase of the estimation distance, and the RMSE shows the same pattern of change as the MAE. This indicates that the prediction accuracy of the model gradually decreases

with the increase in distance. The reason for this phenomenon is that, due to the limitation of Edata, this paper adopts a recursive prediction of vehicle position and takes the output of the model as the input of the next estimation step. As the number of estimation steps increases, the estimation error is constantly accumulated. In recursive estimation, the error tends to accumulate as each step propagates. This may cause the estimation to become unstable over a long period as the error may keep on amplifying. This problem may affect the accuracy and stability of the estimation.

Although the larger the number of estimation steps, the larger the gap between the vehicle position values estimated by the model and the true values, the MAPE, which measures the mean absolute percentage error between the estimated values and the true values, decreases with the growth of the estimation distance, which indicates that the ratio of estimation error to estimation distance decreases to a certain extent with the growth of the estimation distance, and the model performs well in capturing long-term regularity in the changes of vehicle positions on the highway. The model performs well in capturing the long-term regularity of highway vehicle location changes. In addition, from the value of the correlation coefficient R^2 of the model, in the estimation distance of 2km-5km, the model of the vehicle position change law has achieved a better fitting effect, indicating that the model has a better multi-step estimation ability, and can estimate the position of highway over-the-horizon vehicles.

K. MODEL ABLATION EXPERIMENTS

To test the effect of each model on location estimation in the proposed highway in-transit vehicle location estimation method considering the road characteristics and short-term driving style, this method is subjected to ablation experiments in block 3. The role of each module in the estimation performance is revealed by removing different modules step by step. We have used the control variable method to perform the ablation experiment on the models. The experimental results are shown in Table 10.

It can be seen that both the slope and the vehicle segment speed, when not taken into account, have some impact on the performance of the model. However, the performance improvement due to slope is smaller than the performance improvement due to segment speed. The main reason for this is that the main factor affecting the change of vehicle position is the vehicle traveling speed. The faster the vehicle speed, the greater the vehicle position change per unit time. In addition, the segment where the ablation experiments were conducted is located on the southeast coast with no substantial up-and-down grades. For small uphill and downhill slopes, the vehicle owner will adjust the output power of the vehicle to offset the influence of the uphill and downhill slopes, which in turn leads to a weaker influence of the slope in the model.
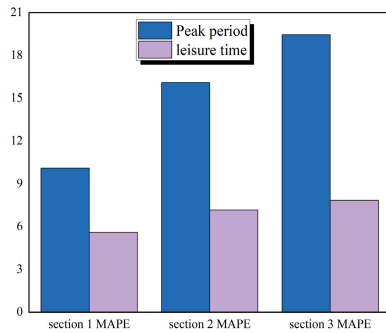
In addition, the performance of the modules in the DLCNN-LSTM-Attention fusion model used individually for
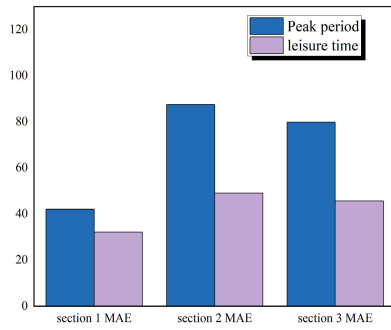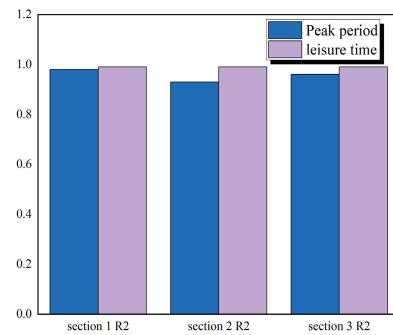
(a) **Comparison chart of mean absolute error.**
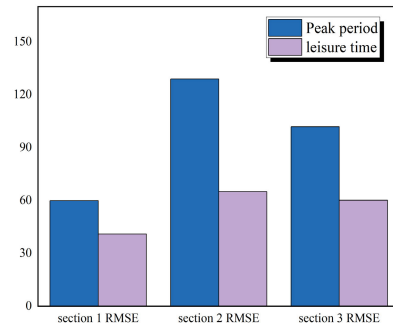
(b) **Root Mean Square Error Comparison Chart.**

(c) **Comparison chart of mean absolute percentage error.**

(d) **Correlation Coefficient Comparison Chart.**

FIGURE 21. Comparison of model estimation effects during busy and idle hours.

TABLE 9. Multistep estimation performance evaluation table.

Indicator \ Distance(m)	2000	2500	3000	3500	4000	4500	5000
segment 1							
MAE	25.32	35.59	41.12	48.15	54.24	58.77	64.88
RMSE	42.15	49.88	58.68	68.10	79.03	88.12	100.16
MAPE	5.05%	5.01 %	4.66%	4.37%	4.11 %	3.89 %	3.75%
R^2	0.99	0.99	0.99	0.99	0.99	0.99	0.99
segment 2							
MAE	38.35	40.92	43.72	48.60	55.18	62.30	68.21
RMSE	42.68	45.64	49.15	54.63	68.89	79.72	86.16
MAPE	8.26%	7.43%	6.71%	6.25%	5.94%	6.68%	5.45%
R^2	0.99	0.99	0.99	0.99	0.99	0.99	0.99
segment 3							
MAE	41.35	47.84					
RMSE	65.53	69.67					
MAPE	9.38%	10.05%					
R^2	0.98	0.98					
segment 4							
MAE	40.80	50.14	57.99	65.78	74.13	84.81	
RMSE	50.03	59.66	67.74	76.70	88.36	103.18	
MAPE	9.49%	8.55 %	7.39 %	6.45%	5.81 %	5.37%	
R^2	0.99	0.99	0.99	0.99	0.99	0.99	

vehicle position estimation shows that the MAE of DLCNN is 50.28, the MAE of LSTM is 46.42, and the MAE of Attention is 82.38. This indicates that DLCNN and LSTM perform better relative to Attention in terms of the better performance on the mean absolute error between predicted and true values. Whereas, the RMSE of DLCNN is 152.57,

TABLE 10. Effect table of vehicle position estimation model ablation experiment.

Indicator \ Model	$Not considering \Theta$	$Not considering V_{S\_pred}$	DLCNN	LSTM	Attention	DLCNN-LSTM	DLCNN-Attention	LSTM-Attention
MAE	43.47	59.95	50.28	46.42	82.38	82.27	53.61	81.44
RMSE	68.21	88.27	152.57	68.26	116.01	116.16	79.60	114.43
MAPE	9.86	11.38	11.55	10.48	14.23	19.14	10.09	18.23
R^2	0.97	0.96	0.89	0.97	0.94	0.94	0.97	0.94

the RMSE of LSTM is 68.26, and the RMSE of Attention is 116.01. The relatively low RMSE of LSTM indicates that the LSTM model performs better in terms of overall prediction error. Finally, the R^2 of the DLCNN model was 0.89, the R^2 of the LSTM model was 0.97, and the R^2 of the Attention model was 0.94. Thus, the LSTM and Attention better explained the variability of the target variable.

From the two-by-two combinations of the modules, the MAE of the DLCNN-LSTM model is 82.27, and the RMSE is 116.16; the MAE of the DLCNN-Attention model is 53.61, and the RMSE is 79.6; and the MAE of the LSTM-Attention model is 81.44, and the RMSE is 114.43. The R^2 of each model exceeds 0.9, and the models achieve good results, but the overall prediction error is still much larger than that of our proposed model.

VI. CONCLUSION AND OUTLOOK

This paper is based on the ETC equipment that has been massively deployed on the highway, considering that the ETC gantry interval is large and cannot accurately perceive the real driving state of the vehicles inside the segment, using Gdata to construct the pattern of change of the position of the vehicles inside the segment, the road structural features, and based on the Edata to construct the historical driving characteristics of the vehicles, the short-term driving style, and then to predict the speed of the vehicles passing through the segment, and on the basis, a highway in-transit vehicle position estimation method considering road characteristics and short-term driving style is proposed. The model performance comparison experiments show that the model proposed in this paper can better extract the change rule of vehicle position from highway multi-featured data, and more accurately estimate the vehicle position in the highway segment.

In terms of estimation error, as one of the practical application scenarios of this method is to assist intelligent driving on highways, and the detection range of the vehicle lidar for intelligent driving can reach more than 100m at present, errors within 100 m can be corrected by the vehicle lidar. Therefore, the minimum detection range of the vehicle lidar is regarded as the acceptable maximum error value in this paper. The experimental results of spatial and temporal dimensions show that this method can realize the accurate estimation of vehicle position with an acceptable error range. The multi-step estimation capability evaluation experiments show that the model can realize long-range vehicle position estimation. Although the estimation error is accumulated as the estimation distance increases, the growth ratio of the error is much smaller than the growth ratio of the estimation distance, which lays a solid foundation for the over-the-horizon estimation capability of the model. Finally, the

ablation experiment of the model also proves the rationality of the model structure design.

In addition, as the research object of this article is the estimation of vehicle position within the highway segment, interference from intersections was excluded during the experiment. In actual driving environments, intersections are common. Therefore, when encountering an intersection in the road, due to the diversity of vehicle path selection, accurate estimation of vehicle position cannot be effectively achieved at this time. Finally, in actual highways, the deployment intervals of ETC gantry vary from a few kilometers to more than ten kilometers, which also leads to the accumulation of vehicle position estimation errors in the long-distance estimation of this model. In future research, this article will further expand the research by considering the impact of vehicle owner's path selection on vehicle position estimation and how to reduce the accumulation of recursive estimation method errors.

REFERENCES

[1] R. D. Kumar and A. Rammohan, "Revolutionizing intelligent transportation systems with cellular vehicle-to-everything (C-V2X) technology: Current trends, use cases, emerging technologies, standardization bodies, industry analytics and future directions," *Veh. Commun.*, vol. 43, Oct. 2023, Art. no. 100638.

[2] K. Jo, K. Chu, and M. Sunwoo, "Interacting multiple model filter-based sensor fusion of GPS with in-vehicle sensors for real-time vehicle positioning," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 329–343, Mar. 2012.

[3] X. Tao, B. Zhu, S. Xuan, J. Zhao, H. Jiang, J. Du, and W. Deng, "A multi-sensor fusion positioning strategy for intelligent vehicles using global pose graph optimization," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 2614–2627, Mar. 2022.

[4] D. Reebadiya, R. Gupta, A. Kumari, and S. Tanwar, "Blockchain and AI-integrated vehicle-based dynamic parking pricing scheme," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2021, pp. 1–6.

[5] R. Gupta, A. Kumari, and S. Tanwar, "A taxonomy of blockchain envisioned edge-as-a-connected autonomous vehicles," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 6, p. e4009, Jun. 2021.

[6] I. Skog and P. Handel, "In-car positioning and navigation technologies—A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 4–21, Mar. 2009.

[7] J. Azimjonov and A. Özmen, "A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways," *Adv. Eng. Informat.*, vol. 50, Oct. 2021, Art. no. 101393.

[8] Z. Li, C. Gong, Y. Lin, G. Li, X. Wang, C. Lu, M. Wang, S. Chen, and J. Gong, "Continual driver behaviour learning for connected vehicles and intelligent transportation systems: Framework, survey and challenges," *Green Energy Intell. Transp.*, vol. 2, no. 4, Aug. 2023, Art. no. 100103.

[9] A. Chand, S. Jayesh, and A. B. Bhasi, "Road traffic accidents: An overview of data sources, analysis techniques and contributing factors," *Mater. Today*, vol. 47, pp. 5135–5141, Jan. 2021.

[10] C. Hancock, G. Roberts, and A. Taha, "Satellite mapping in cities: How good can it get?" *Proc. Inst. Civil Eng.-Civil Eng.*, vol. 162, no. 3, pp. 122–128, Aug. 2009.

[11] S. Ji, W. Chen, X. Ding, Y. Chen, C. Zhao, and C. Hu, "Potential benefits of GPS/GLONASS/GALILEO integration in an urban canyon—Hong Kong," *J. Navigat.*, vol. 63, no. 4, pp. 681–693, Oct. 2010.

[12] P. D. Groves, "Shadow matching: A new GNSS positioning technique for urban canyons," *J. Navigat.*, vol. 64, no. 3, pp. 417–430, Jul. 2011.

[13] C. Hancock, P. Zhang, L. Lau, G. Roberts, and H. de Ligt, "Satellite mapping in cities and below cities: How good is it now?" *Proc. Inst. Civil Eng.-Civil Eng.*, vol. 170, no. 1, pp. 33–38, Feb. 2017.

[14] Y. Gu, L.-T. Hsu, and S. Kamijo, "GNSS/onboard inertial sensor integration with the aid of 3-D building map for lane-level vehicle self-localization in urban canyon," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4274–4287, Jun. 2016.

[15] A. Eskandarian, C. Wu, and C. Sun, "Research advances and challenges of autonomous and connected ground vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 683–711, Feb. 2021.

[16] S. Rafatnia, H. Nourmohammadi, J. Keighobadi, and M. A. Badamchizadeh, "In-move aligned SINS/GNSS system using recurrent wavelet neural network (RWNN)-based integration scheme," *Mechatronics*, vol. 54, pp. 155–165, Oct. 2018.

[17] Z. Wu, M. Yao, H. Ma, and W. Jia, "Improving accuracy of the vehicle attitude estimation for low-cost INS/GPS integration aided by the GPS-measured course angle," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 553–564, Jun. 2013.

[18] H. Ahmed and M. Tahir, "Accurate attitude estimation of a moving land vehicle using low-cost MEMS IMU sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1723–1739, Jul. 2017.

[19] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Ubeda-Minarro, and A. F. Gomez-Skarmeta, "High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 491–511, Sep. 2007.

[20] I. Skog and P. Handel, "Time synchronization errors in loosely coupled GPS-aided inertial navigation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1014–1023, Dec. 2011.

[21] S. Rezaei and R. Sengupta, "Kalman filter-based integration of DGPS and vehicle sensors for localization," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 6, pp. 1080–1088, Nov. 2007.

[22] H. M. Omar, Z. Yanzhong, Z. Bo, and H. U. Gul, "Integration of GPS and dead reckoning navigation system using moving horizon estimation," in *Proc. IEEE Inf. Technol., Netw., Electron. Autom. Control Conf.*, May 2016, pp. 553–556.

[23] H. Wang, L. Wan, M. Dong, K. Ota, and X. Wang, "Assistant vehicle localization based on three collaborative base stations via SBL-based robust DOA estimation," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5766–5777, Jun. 2019.

[24] Y. Guo et al., "Traffic target location estimation based on tensor decomposition in intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, Mar. 2022.

[25] V. Havyarimana, Z. Xiao, A. Sibomana, D. Wu, and J. Bai, "A fusion framework based on sparse Gaussian–Wigner prediction for vehicle localization using GDOP of GPS satellites," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 680–689, Feb. 2020.

[26] K. Jo, M. Lee, and M. Sunwoo, "Road slope aided vehicle position estimation system based on sensor fusion of GPS and automotive onboard sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 250–263, Jan. 2016.

[27] M. Bechler, S. Jaap, and L. Wolf, "An optimized TCP for Internet access of vehicular ad hoc networks," in *Proc. Int. Conf. Res. Netw.* Berlin, Germany: Springer, 2005, pp. 869–880.

[28] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, "ADHOC MAC: New MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services," *Wireless Netw.*, vol. 10, no. 4, pp. 359–366, Jul. 2004.

[29] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Vehicular ad hoc networks: A new challenge for localization-based systems," *Comput. Commun.*, vol. 31, no. 12, pp. 2838–2849, Jul. 2008.

[30] L. Briesemeister, L. Schafers, and G. Hommel, "Disseminating messages among highly mobile hosts based on inter-vehicle communication," in *Proc. IEEE Intell. Vehicles Symp.*, Oct. 2000, pp. 522–527.

[31] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *J. Netw. Comput. Appl.*, vol. 37, pp. 380–392, Jan. 2014.

[32] M. F. Tsai et al., "Improving positioning accuracy for VANET in real city environments," *J. Supercomput.*, vol. 71, pp. 1975–1995, 2015.

[33] F. Domingos, L. Villas, and A. Boukerche, "Data communication in VANETs: Survey, applications and challenges," *Ad Hoc Networks*, vol. 44, pp. 90–103, Sep. 2016.

[34] L. Wan, Y. Sun, L. Sun, Z. Ning, and J. J. P. C. Rodrigues, "Deep learning based autonomous vehicle super resolution DOA estimation for safety driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4301–4315, Jul. 2021.

[35] Z. Yuexia and L. Chong, "A vehicle location method based on neural networks and roadside unit fingerprints," *Comput. Simul.*, vol. 39, no. 4, pp. 114–118, 2022.

[36] M. S. Alzyout and M. A. Alsmirat, "Performance of design options of automated ARIMA model construction for dynamic vehicle GPS location prediction," *Simul. Model. Pract. Theory*, vol. 104, Nov. 2020, Art. no. 102148.

[37] E. B. Anitha and K. Duraiswamy, "A heuristic moving vehicle location prediction technique via optimal paths selection with aid of genetic algorithm and feed forward back propagation neural network," *J. Comput. Sci.*, vol. 8, no. 12, p. 2008, 2012.

[38] X. Fan, L. Guo, N. Han, Y. Wang, J. Shi, and Y. Yuan, "A deep learning approach for next location prediction," in *Proc. IEEE 22nd Int. Conf. Comput. Supported Cooperat. Work Design ((CSCWD))*, May 2018, pp. 69–74.

[39] W. Long, T. Li, Z. Xiao, D. Wang, R. Zhang, A. C. Regan, H. Chen, and Y. Zhu, "Location prediction for individual vehicles via exploiting travel regularity and preference," *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 4718–4732, May 2022.

[40] Y. Xiao and Q. Nian, "Vehicle location prediction based on spatiotemporal feature transformation and hybrid LSTM neural network," *Information*, vol. 11, no. 2, p. 84, Feb. 2020.

[41] Y. Ma, W. Li, K. Tang, Z. Zhang, and S. Chen, "Driving style recognition and comparisons among driving tasks based on driver behavior in the online car-hailing industry," *Accident Anal. Prevention*, vol. 154, May 2021, Art. no. 106096.

[42] Y. Zhang, N. Liu, and S. Wang, "A differential privacy protecting K-means clustering algorithm based on contour coefficients," *PLoS ONE*, vol. 13, no. 11, Nov. 2018, Art. no. e0206832.

[43] D. Huang, F. Jiang, K. Li, G. Tong, and G. Zhou, "Scaled PCA: A new approach to dimension reduction," *Manage. Sci.*, vol. 68, no. 3, pp. 1678–1695, Mar. 2022.

[44] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Inf. Sci.*, vol. 622, pp. 178–210, Apr. 2023.

[45] A. Kulshrestha, V. Krishnaswamy, and M. Sharma, "Bayesian BILSTM approach for tourism demand forecasting," *Ann. Tourism Res.*, vol. 83, Jul. 2020, Art. no. 102925.

[46] Q. Pan, L. Zhang, G. Dai, and H. Zhang, "Two denoising methods by wavelet transform," *IEEE Trans. Signal Process.*, vol. 47, no. 12, pp. 3401–3406, 1999.

[47] W. H. Chung, Y. H. Gu, and S. J. Yoo, "District heater load forecasting based on machine learning and parallel CNN-LSTM attention," *Energy*, vol. 246, May 2022, Art. no. 123350.

[48] A. Wan, Q. Chang, A. L. B. Khalil, and J. He, "Short-term power load forecasting for combined heat and power using CNN-LSTM enhanced by attention mechanism," *Energy*, vol. 282, Nov. 2023, Art. no. 128274.

[49] A. Kumari and S. Tanwar, "ρ reveal: An AI-based big data analytics scheme for energy price prediction and load reduction," in *Proc. 11th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence)*, Jan. 2021, pp. 321–326.

GUANGHAO LUO received the bachelor's degree in electrical engineering from the Computer Science Department, Minjiang University, in 2014. He is currently pursuing the master's degree in electrical engineering with the School of Electrical Engineering, Fujian University of Technology.

He was a DCS Engineer with Fujian Qingshan Paper Industry Company Ltd. His research interests include big data technology, traffic information, and control engineering.

FUMIN ZOU received the Ph.D. degree from Central South University, a renowned youth in Fujian Province, with a talent of millions and millions in Fujian Province.

Formerly, he was the Director of the Scientific Research Department, Fujian University of Engineering. Since December 2020, he has been the Dean of the School of Electrical Engineering, Fujian University of Technology, with a focus on big data technology, traffic information, and control engineering.

TING YE received the bachelor's degree in electrical engineering from the Zhicheng College, Fuzhou University, in 2022. She is currently pursuing the master's degree in electrical engineering from the School of Electrical Engineering, Fujian University of Technology. Her research interests include transportation big data mining and deep learning.

FENG GUO received the master's degree in computer science and technology from Tsinghua University, in 2013, and the Ph.D. degree in computer science and technology from Fuzhou University, in 2023.

He has served as a Senior Product Architect for Kingdee (China) Software and Lanling Software. He is currently with the School of Electrical Engineering, Fujian University of Technology. His research interests include big data technology, traffic information, and control engineering.

QIQIN CAI received the master's degree in electrical engineering from the Fujian University of Technology, in 2017. He is currently pursuing the Ph.D. degree with Huaqiao University.

He has participated in major projects, such as the Research on Key Technologies for Expanding the Application of Highway ETC. His current research interests include big data mining and deep learning in transportation.

GEN XU received the master's degree in transportation engineering from the Fujian University of Technology, in 2023. He is currently with Bohai Bank Company Ltd. His research interests include transportation big data mining and deep learning.

• • •