Article

Research on Vehicle-Driving-Trajectory Prediction Methods by Considering Driving Intention and Driving Style

Liming Shao [1], Meining Ling [1], Ying Yan [2], Guangnian Xiao [3], Shiqi Luo [4] and Qiang Luo [4],*

1   Guangzhou Urban Planning & Design Survey Research Institute Co., Ltd., Guangzhou 510060, China; shaoliming@gzpi.com.cn (L.S.); lingmeining@gzpi.com.cn (M.L.)
2   School of Automobile, Key Laboratory of Automobile Transportation Safety Support Technology, Chang'an University, Xi'an 710064, China; yanying2199@chd.edu.cn
3   School of Economics and Management, Shanghai Maritime University, Shanghai 201306, China; gnxiao@shmtu.edu.cn
4   School of Civil Engineering and Transportation, Guangzhou University, Guangzhou 510006, China; 2112216340@e.gzhu.edu.cn
*   Correspondence: luoq_yan@gzhu.edu.cn

Abstract: With the rapid advancement of autonomous driving technology, the accurate prediction of vehicle trajectories has become a research hotspot. In order to accurately predict vehicles' trajectory, this study comprehensively explores the impact of driving style and intention on trajectory prediction, proposing a novel prediction method. Firstly, the dataset AD4CHE was selected as the research data, from which the required trajectory data of vehicles were extracted, including 1202 lane-changing and 1137 car-following driving trajectories. Secondly, a long short-term memory (LSTM) network based on the Keras framework was constructed by using the TensorFlow deep-learning platform. The LSTM network integrates driving intention, driving style, and historical trajectory data as inputs to establish a vehicle-trajectory prediction model. Finally, the mean absolute error (MAE) and root-mean-square error (RMSE) were selected as the evaluation indicators for the models, and the prediction results of the models were compared under two conditions: not considering driving style and considering driving style. The results demonstrate that models incorporating driving style significantly outperformed those that did not, highlighting the critical influence of driving style on vehicle trajectories. Moreover, compared to traditional kinematic models, the LSTM-based approach exhibits notable advantages in long-term trajectory prediction. The prediction method that accounts for both driving intention and style effectively reduces RMSE, significantly enhancing prediction accuracy. The findings of this research provide valuable insights for vehicle-driving risk assessment and contribute positively to the advancement of autonomous driving technology and the sustainable development of road traffic.

Keywords: traffic safety; trajectory prediction; driving intention and style; sustainable transportation; long short-term memory

1. Introduction

With continuous improvements in standard of living, the consumer desire for cars has been steadily increasing, leading to a rapid rise in the number of motor vehicles. Statistics show that, in 2023, the number of motor vehicles in China exceeded 400 million, an increase of nearly 25 million compared to the previous year [1]. While automobiles have brought convenience to people's travel, they have also caused problems such as traffic congestion and traffic safety, posing a significant threat to the safety of people's lives and property. With the advent of intelligent, connected vehicles and big data, the development of autonomous driving and intelligent driver assistance technologies provides solutions to improve road traffic safety and alleviate congestion. Among these technologies, vehicle-trajectory prediction is a key component of autonomous driving and intelligent driver

assistance. By predicting the future trajectory of a vehicle and assessing driving risks using traffic conflict or safety technologies, it is possible to identify potential dangers in real time. This enables corrective actions by the driver, helping to prevent traffic accidents. However, accurately predicting the future trajectory of a vehicle in a dynamic and complex traffic environment presents numerous challenges and difficulties [2,3].

Early methods for vehicle-trajectory prediction were mostly based on physical models, utilizing vehicle kinematic models to relate steering angles, acceleration, and other factors to the vehicle's motion trends [4–6]. For example, Zyner et al. [7] predicted vehicle-driving behavior using parameters such as vehicle speed, position, and heading angle; Xin et al. [8] addressed issues such as sensor limitations and obstacle occlusion by using a deep-learning model with single-vehicle motion state information as the input for the trajectory prediction. However, methods based on physical models have significant limitations in application, as they are only reliable for short-term trajectory prediction. Later, researchers applied Markov models [9,10] and Kalman filtering [11,12] to predict trajectories, achieving promising results. For instance, Nishiwaki et al. [9] used a hidden Markov model trained with vehicle longitudinal speed, lateral position, and dynamic characteristics to generate vehicle trajectories. Xiao et al. [13] proposed an interactive multi-model vehicle-trajectory prediction method based on the fusion of the motion model and the maneuver model, by combining the constant-turn-rate and acceleration (CTRA) motion model and unscented Kalman filter to predict uncertain vehicle trajectories in the future.

Nonetheless, these methods also suffer from decreased accuracy in long-term predictions. Consequently, scholars began exploring other methods for vehicle-trajectory prediction. Xu et al. [14] proposed an extended dynamic model to represent evasive lane-changing behavior, which generated more accurate lane-change trajectories compared to the quintic polynomial lane-change model. Xie et al. [15] predicted vehicle trajectories based on behavior recognition and curvature constraints, using lane curvature as a constraint to predict the optimal trajectory.

With the development of machine learning and deep learning, data-driven trajectory prediction has increasingly gained attention and importance [16–19]. Under conditions with large datasets, Dai et al. [20] considered spatial interactions between different vehicles and temporal relationships in trajectory time series, proposing a spatiotemporal LSTM-based trajectory prediction model and verifying its effectiveness. Wen et al. [21] proposed a generative-adversarial-network (GAN)-based vehicle lane-change trajectory prediction model, which significantly improved prediction accuracy compared to traditional models. Yang et al. [22] proposed a vehicle lane-changing trajectory prediction model based on temporal convolutional networks with an attention mechanism to improve the accuracy of vehicle lane-changing trajectory prediction, which can predict the trajectory of the lane change with high accuracy. Xue et al. [23] focused on the limited consideration of traffic environment influences on vehicle maneuvers in current models and developed a machine-learning-based integrated lane-change prediction model that incorporates the traffic context, making it applicable to various traffic environments. Guo et al. [24] developed a transformer-based vehicle lane-change prediction model using large-scale real trajectory data collected from connected vehicles.

To solve more complex problems in prediction, the use of trajectory prediction methods based on hybrid models is gradually increasing. Deo et al. [25] proposed a trajectory prediction method based on an LSTM model, which learns the dynamics of the predicted vehicle motion through an encoder, captures the interdependence of surrounding target movements in the scene through a convolutional social pooling layer, and combines the two to decode and output a multimodal prediction probability distribution of the predicted vehicle. To explain the influence of historical trajectories and adjacent vehicles on the target vehicle, Lin et al. [26] proposed an LSTM model STA-LSTM with a spatiotemporal attention mechanism based on the literature [25] for vehicle-trajectory prediction. The spatiotemporal attention weights provided by the model enhance its interpretability. To further obtain multidimensional data and extract effective information, research based on

graph convolutional networks (GCN) and attention mechanisms is gradually increasing. Liang et al. [27] used Lane GCN and Actor Net to extract map features and target motion features and then used a fusion network composed of four types of interactions to generate multimodal trajectory predictions. Messaoud et al. [28] proposed a model that combines grid-based trajectory encoding with LSTM and multi-head self-attention.

Another data-driven approach is to learn the motion behavior of vehicles from a large amount of trajectory data, accurately identify the driving intention of the vehicle, and perform trajectory prediction to improve the accuracy of the trajectory prediction [29]. Ji et al. [30] designed a driving-intention recognition and vehicle-trajectory prediction model based on an LSTM network, introducing a mixed-density network to represent the vehicle's future position. Dai et al. [20] considered the spatiotemporal interaction relationship between vehicles and embedded spatial interaction into the model, proposing a trajectory prediction model based on spatiotemporal LSTM. Shi et al. [31] proposed a trajectory prediction algorithm that combines car-following and lane-changing, including a lane-changing-intention prediction module, trajectory encoder, and trajectory decoder. Due to the influence of multiple modules on the prediction results of model, the overall trajectory prediction accuracy of the model is not significantly improved.

In summary, early research focused on using kinematic models for modeling and prediction and fitting trajectories with polynomials, which could not adapt to complex and changing scenarios or meet the demands of real-time prediction. With advancements in data collection and processing technologies, many researchers have used machine-learning methods to predict vehicle trajectories based on large-scale trajectory datasets. Numerous studies have shown that vehicle trajectory is influenced by different driving intentions and driving styles. That is, even in the same driving environment, there can be significant differences in the trajectories controlled by drivers with different styles. Therefore, considering the increasingly complex road traffic environment, this paper comprehensively explores the relationship between driving style, driving intention, and vehicle-trajectory prediction. The study aims to develop a precise, effective, and adaptive vehicle-trajectory prediction model, which holds significant implications for vehicle risk assessment, driving decision making, and autonomous driving.

The remainder of this study is structured as follows: Section 2 introduces the concept of the trajectory prediction model and the research schematic overview of this study. Section 3 presents the vehicle-trajectory datasets and extracted trajectory data. Section 4 introduces the construction of the trajectory prediction model, which includes the model input data, model evaluation metrics, and model parameter settings. Section 5 presents the predicting results of the model as well as the comparison to other models, and the experimental results of experiments are analyzed and discussed. Finally, the conclusions and future perspectives are presented in Section 6.

2. Overview of Prediction Models

2.1. Recurrent Neural Network

Traditional neural networks do not have connections between neurons in different layers, meaning that the output of the current state does not influence subsequent states. This lack of continuity and learning accumulation renders traditional neural networks inadequate for modeling time-series data. The advent of the recurrent neural network (RNN) addressed this limitation. The RNN can accept variable-length sequences as input and possesses memory capabilities, allowing it to capture temporal dependencies within sequences. RNN have found wide applications in fields such as natural language processing and time-series analysis.

An RNN is a chain-like neural network consisting of three layers: an input layer, a hidden layer, and an output layer. Data enters through the input layer, undergoes multiple iterations within the hidden layer, and is, finally, output through the output layer. The basic structure of an RNN is illustrated in Figure 1. The distinctive feature of RNN is their ability to carry information from one time step to the next, accumulating information over time

through internal connections within the hidden layer. This chain-like structure enables the RNN to effectively handle sequential data, such as vehicle-trajectory data. However, as time accumulates, the RNN may face issues such as vanishing gradients or exploding gradients during backpropagation, making it difficult to effectively learn information from long historical sequences, ultimately impairing their learning capability [32,33].
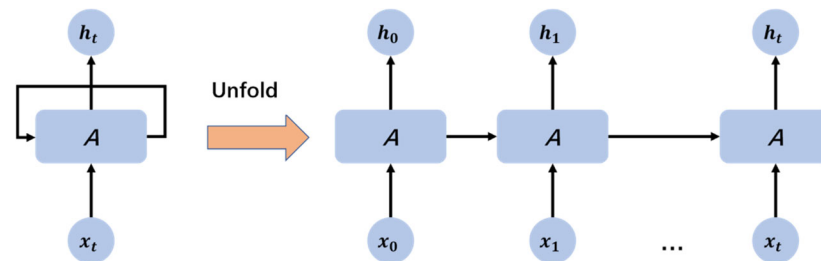

Figure 1. RNN network basic structure.

2.2. Long Short-Term Memory Network

As an improved version of RNN, long short-term memory (LSTM) networks not only retain the ability of an RNN to handle time-series problems but also effectively address the issues of vanishing gradients and exploding gradients [34,35]. An LSTM network is composed of multiple cells, each responsible for maintaining and updating the cell state. The structure of a single LSTM unit, illustrated in Figure 2, includes three critical gates: the forget gate, the input gate, and the output gate. These gates interact with the cell state and, by introducing nonlinear elements, enable the network to more efficiently extract and learn relevant information from long sequences of data.
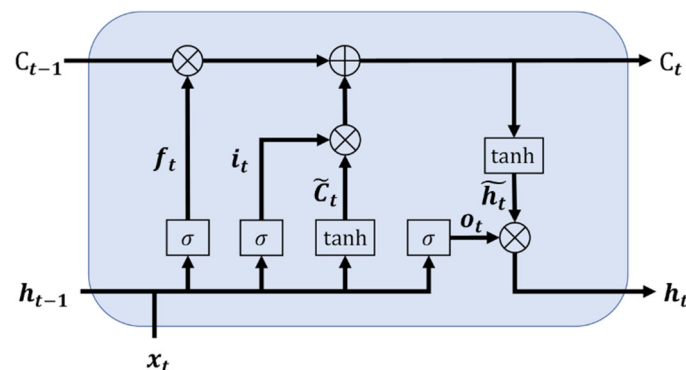

Figure 2. LSTM network unit structure.

The forget gate determines which information from the previous cell state should be discarded, the input gate decides what new information should be stored in the cell state, and the output gate controls the output of the current cell state. This gate mechanism allows LSTM networks to selectively retain or discard information over long periods, making them particularly effective for learning dependencies in long-term sequences.

The forget gate is used to control whether the information in the cell state should be forgotten, allowing the network to better handle long-term dependencies. By passing the current input x_t and the hidden state from the previous time step h_{t-1} through an activation function, the forget gate regulates the extent to which the current input and the previous hidden state are forgotten. This process helps to mitigate the influence of earlier information on subsequent information. The process of information transmission through the forget gate is as follows:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

$$\sigma = \frac{1}{1 + e^{-x}} \tag{2}$$

In this context, σ represents the *sigmoid* activation function, W_f is the weight matrix for the forget gate, and b_f is the bias term.

The input gate controls how the current input updates the cell state, thereby affecting the network's memory and learning capabilities. It consists of two main components:

Sigmoid Layer: This layer controls the selection of values that should contribute to the state update by filtering the input. It applies the *sigmoid* activation function to determine which parts of the current input and previous hidden state will influence the new state.

Tanh Layer: This layer uses the *tanh* activation function to compute a new candidate value $\widetilde{C}_t$ for the cell state. The *tanh* function scales the input to a range between -1 and 1, allowing the network to add or subtract information in a more controlled manner.

The results from the *sigmoid* and *tanh* layers are then multiplied to form the new state for the current time step. This process enables the input gate to flexibly filter out irrelevant information from the current input while incorporating relevant historical information. The process of updating the cell state through the input gate is as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\widetilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{4}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \tag{5}$$

$$\tanh = \frac{e^x - e^{-x}}{e^x - e^{-x}} \tag{6}$$

In this context, W_c and W_i are the weight matrices associated with the candidate state and the input gate, respectively. Similarly, b_c and b_i are the bias terms corresponding to the candidate state and the input gate.

The output gate determines the final output of the cell state and consists of both a *sigmoid* layer and a *tanh* layer. The process works as follows:

Sigmoid layer calculation: The current input x_t and the previous hidden state h_{t-1} are passed through a *sigmoid* activation function, which decides the extent to which the cell state should be expressed in the output. The *sigmoid* function generates an output gate value o_t, which ranges between 0 and 1, indicating how much of the cell state will contribute to the final output.

Tanh layer calculation: Simultaneously, the *tanh* activation function is applied to the cell state to compute a new candidate hidden state h_t. This candidate hidden state represents the filtered version of the cell state, constrained within the range of -1 to 1.

Final output calculation: The final hidden state $\widetilde{h}_t$ of the LSTM is then obtained by multiplying the output of the *sigmoid* function o_t by the candidate hidden state h_t. This operation selectively filters the cell state to produce the final output, which balances the contribution of the cell's memory and the current input.

The process of the output gate can be summarized as follows:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{7}$$

$$\widetilde{h}_t = \tanh(C_t) \tag{8}$$

$$h_t = o_t \cdot \widetilde{h}_t \tag{9}$$

In this context, W_o is the weight matrix associated with the output gate, and b_o is the bias term for the output gate.

2.3. Research Schematic Overview

Through the above analysis, it can be seen that LSTM has strong information mining and deep representation capabilities in dealing with temporal problems and has significant advantages in vehicle-trajectory prediction. Therefore, this study is based on the LSTM network, comprehensively considering the influence of driving style and driving intention on vehicle-driving trajectory and jointly using both as inputs with historical trajectory data to construct a vehicle-trajectory prediction method based on the LSTM network. The research flowchart for the proposed framework is shown in Figure 3.
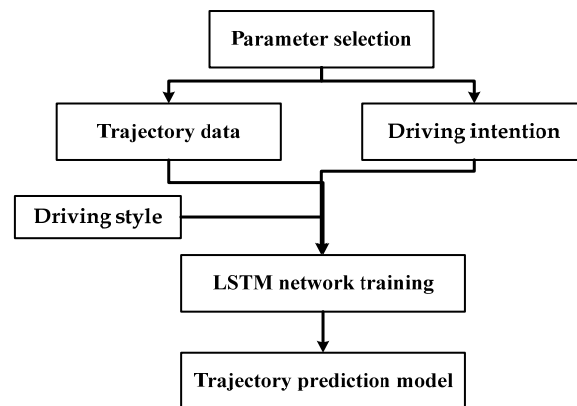

Figure 3. Research flow chart.

3. Trajectory Data Extraction

With the advancement of data collection devices and processing technologies, numerous natural vehicle-trajectory datasets have emerged, providing strong data support for the study of microscopic driving behaviors. The AD4CHE dataset was collected in 2022 and includes vehicle trajectories from multiple highways and expressways in five cities in different regions of China, covering different road types and structures. The acquisition frame rate of this dataset is 30 Hz, the overall positioning accuracy is about 5 cm, and the maximum error is below 10 cm. At the same time, in order to solve the problem of slight shaking during drone shooting, the data collection team performed video alignment preprocessing on the captured video data. Thus, after comparative analysis, this study selects the AD4CHE dataset as the data source, from which the lane-changing and car-following trajectory data are extracted. Specifically, the dataset includes 150 instances of right-lane-change trajectories, 1052 instances of left-lane-change trajectories, and 1137 instances of car-following trajectories, as illustrated in Figures 4–6. In actual driving, drivers tend to drive at a faster or more ideal speed; thus, the number of trajectories for left-lane changing is much greater than that for right-lane changing under normal conditions [36].
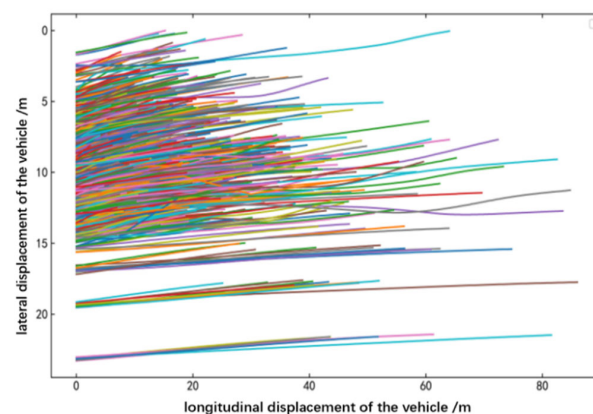

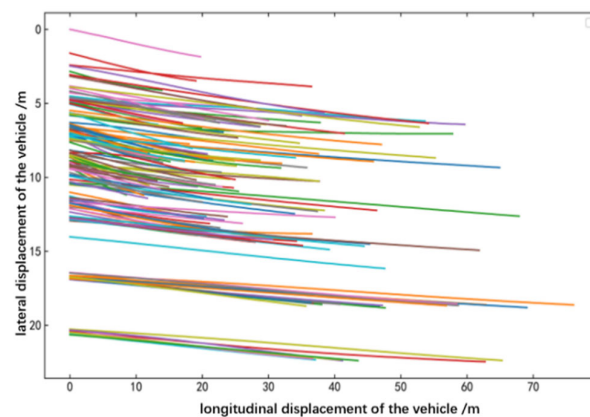**Figure 4.** Left-lane-change trajectories cluster.

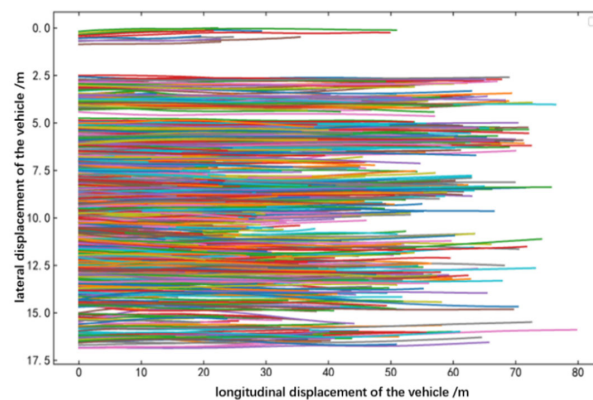**Figure 5.** Right-lane-change trajectories cluster.

Figure 6. Car-following trajectories cluster.

To ensure the balance of research samples and to avoid a poor learning performance during model training due to differences in the number of data points across categories, 100 trajectories were randomly selected from each of the extracted left-lane-change, right-lane-change, and car-following trajectory data. These selected trajectories serve as the research sample data for this study.

4. Trajectory Prediction Model

4.1. Model Input Data

In this study, an LSTM network based on the Keras framework was constructed using the TensorFlow deep-learning platform. The input sequence to the model consisted of three components: the number of samples, the time steps, and the number of features. Key feature parameters that effectively describe changes in the vehicle's trajectory, such as the vehicle's lateral and longitudinal coordinates, speed, and acceleration, were selected as inputs for model training and prediction.

Since driving decisions and execution during the driving process depend on driving intentions, which in turn influence the vehicle's future trajectory, driving intentions were also converted into the one-hot encoded format and included as feature parameters in the model. Specifically, the encoding is as follows: (1,0,0) represents a left-lane change, (0,1,0) represents car-following behavior, and (0,0,1) represents a right-lane change.

$$X = [X_{T-N}, X_{T-N+1}, X_{T-N+2}, ..., X_T] \tag{10}$$

$$X_T = \left[x^T, y^T, v_x{}^T, v_y{}^T, a_x{}^T, a_y{}^T, itt^T \right] \tag{11}$$

In this context, X represents the input to the model, and X_T denotes the input at time step T. The parameter N is the length of the input historical trajectory. x^T and y^T represent

the longitudinal and lateral positions at time T, $v_x{}^T$ and $v_y{}^T$ represent the longitudinal and lateral velocities at time T, and $a_x{}^T$ and $a_y{}^T$ represent the longitudinal and lateral accelerations at time T. The variable itt^T represents the driving intention at time T.

Additionally, to explore the impact of driving style on the prediction of future vehicle trajectories, driving style was included as a new feature parameter in the LSTM network, along with historical trajectory data and driving intentions, as part of the model's input. For the clustering of driving styles, this study selected three feature parameters: the lateral velocity variation coefficient, the acceleration impact coefficient, and the collision time coefficient and used the k-means clustering algorithm to divide driving styles into three categories: conservative, normal, and aggressive driving style. Before being input into the model, the three types of driving styles were converted into a one-hot encoded format, where (1,0,0) represents a conservative driving style, (0,1,0) represents a normal driving style, and (0,0,1) represents an aggressive driving style.

$$X_T{}' = \left[x^T, y^T, v_x{}^T, v_y{}^T, a_x{}^T, a_y{}^T, itt^T, ds^T \right] \tag{12}$$

In this context, ds^T represents the driving style at time T.

The trajectory sample data was divided into testing and training sets in a 1:4 ratio. The LSTM model was then used to train the constructed time-series data. Based on the research team's previous work on the optimal recognition window length for driving intention [37], historical trajectory data with a duration of 2 s was selected as the input to the model. The future trajectory was predicted using a sliding time window method, with each sliding step corresponding to 1 frame in the trajectory data and a step size of 1.

The model predicted the next frame of trajectory data based on 2 s of historical trajectory data. The predicted trajectory data from this step was then combined with the previous historical trajectory data, excluding the first frame, to form a new input. This iterative process continued to predict trajectory data for future time steps.

This method allows the model to continuously update its predictions as new data becomes available, making it possible to generate a more accurate trajectory prediction over time by leveraging recent historical information and driving style influences.

4.2. Model Evaluation Metrics

To accurately assess the performance of the LSTM model in the trajectory prediction, the mean absolute error (MAE) and root-mean-square error (RMSE) are used as evaluation metrics. The MAE measures the average magnitude of errors in a set of predictions, without considering their direction. It is a linear score, meaning that all individual differences are weighted equally. The RMSE measures the square root of the average of the squared differences between the predicted and actual values. Unlike MAE, it penalizes larger errors more heavily due to the squaring of differences, making it more sensitive to outliers. The mathematical expression for MAE and RMSE are as shown in Equations (13) and (14).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{13}$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{14}$$

In these equations, n represents the number of samples, x_i and y_i are the actual values, and $\hat{x}_i$ and $\hat{y}_i$ are the predicted values. The smaller the MAE and RMSE values, the closer the model's predictions are to the true values, indicating a better predictive performance of the model.

4.3. Model Parameter Settings

The Adam algorithm is a commonly used optimization algorithm that is widely applied in the field of deep learning. Compared to other traditional gradient descent methods, the Adam algorithm has the characteristics of adaptive learning rate and second-order moment estimation, which can adaptively adjust the learning rate according to the gradient characteristics of different parameters, thereby accelerating the convergence of the model and improving the training effect. Based on the characteristics of the vehicle's driving trajectory, the Adam algorithm is used as the optimization algorithm, with root-mean-square error (RMSE) as the loss function for model training and optimization. The commonly used optimization steps of the Adam algorithm are as follows: calculation of the momentum estimate of the gradient (first moment estimate), calculation of the momentum estimate of the squared gradient (second moment estimate), bias correction, and updating model parameters. Through these steps, the Adam algorithm dynamically adjusts the learning rate for each parameter during training, effectively balancing the model's convergence speed and stability, thereby improving training efficiency and performance. Additionally, the selection of hyperparameters significantly influences the network's performance, convergence speed, and generalization ability. The key hyperparameters in the model are set as follows:

Learning rate: In an LSTM network, the learning rate is a critical parameter that controls the step size of parameter updates during training. To ensure a good learning performance, an adaptive-decay learning rate is introduced during training. The initial learning rate is set to 0.001. If there is no improvement in the training set over 10 epochs, the learning rate is reduced by a decay factor of 0.2. This strategy helps ensure the stable convergence of the model.

Training epochs: A training epoch refers to the process of passing all the training data through the network once and performing a single parameter update. The number of training epochs affects the model's ability to learn data features. With too few epochs, the model may not fully learn the data's features, leading to underfitting. Conversely, too many epochs can cause the model to overfit the training data. Considering the number of trajectory samples used in this study, the maximum number of iterations is set to 300 epochs.

The above parameter settings are for optimizing the model's learning process, ensuring a balance between adequate training and the prevention of overfitting, thus leading to a robust and generalizable model.

5. Experimental Results and Discussion

To study the impact of driving style on trajectory prediction results, two models were compared: one that does not consider driving style (the control group) and one that does (the experimental group). The selected evaluation metrics were used to assess the performance of these models in vehicle-trajectory prediction. Meanwhile, the model proposed in this study was compared to two traditional LSTM models to validate their predictive performance.
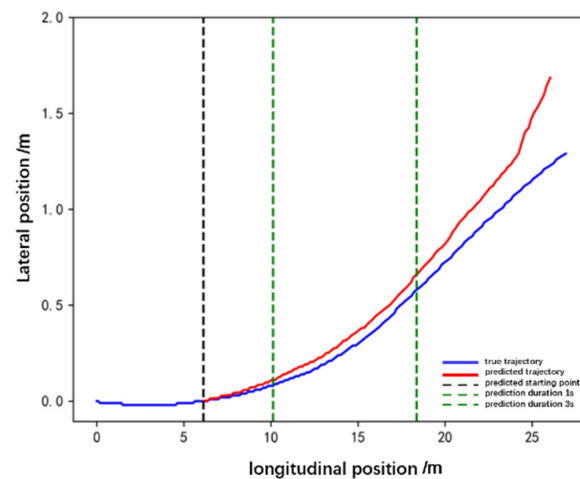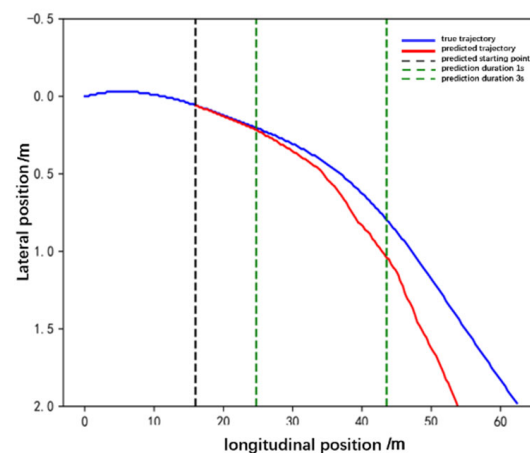
5.1. Prediction Results without Considering Driving Style

When the impact of driving style on trajectory prediction was not considered, only driving intention and historical trajectory data were used as inputs to the LSTM network. After completing the model training, the model was used to predict vehicle-trajectory data for the next 5 s. The prediction performance was evaluated using the lateral and longitudinal MAE and RMSE metrics, with the results shown in Table 1.

Table 1. Results of evaluation indicators without considering driving style.

Prediction Period	1 s	3 s	5 s
Lateral MAE	0.0179	0.1587	0.8966
Longitudinal MAE	0.1097	1.0588	2.5366
RMSE	0.1252	1.1762	4.5331

Due to the high similarity of trajectory data of the same type, in order to test the predictive performance of the prediction model, one trajectory was randomly selected from each of the three types of trajectory data for prediction and compared to the real trajectory data, as shown in Figures 7–9. It can be seen from these three figures that the predicted trajectories closely follow the general trend of the actual trajectories, with the model performing the best in predicting the car-following trajectory, indicating that the LSTM network can effectively predict the vehicle's future trajectory. Especially when the prediction period is within 3 s, the prediction effect is better. From the data in Table 1, it can be seen that the trajectory prediction errors from the model tend to accumulate and increase as the prediction time increases. Comparing the magnitude of the horizontal and vertical errors, the results show that the longitudinal MAE is much larger than the lateral MAE, which may be due to the difference in vehicle speed between the horizontal and vertical directions.

Figure 7. Prediction results of left lane-changing.

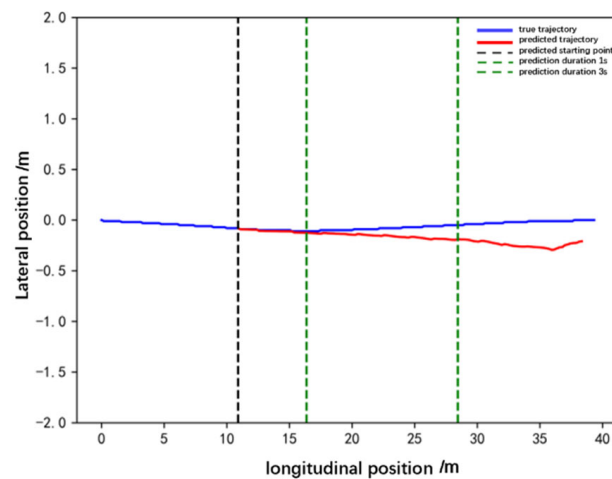**Figure 8.** Prediction results of right lane-changing.

Figure 9. Prediction results of car following.

5.2. Prediction Results Considering Driving Style

When the impact of driving style on vehicle-trajectory prediction was considered, driving style, driving intention, and historical trajectory data were all used as inputs to the LSTM network. After completing the model training, the model was used to predict vehicle-trajectory data for the next 5 s. The prediction performance was evaluated using the lateral and longitudinal MAE and RMSE metrics, with the results shown in Table 2.

Table 2. Results of evaluation indicators considering driving style.

Prediction Period	1 s	3 s	5 s
Lateral MAE	0.0114	0.1218	0.6201
Longitudinal MAE	0.1050	0.9666	2.2808
RMSE	0.1186	1.1245	4.2425

Similarly, one predicted trajectory from each category was randomly selected and compared to the actual trajectory, as shown in Figures 10–12. These three figures demonstrate that the predicted trajectories closely follow the general trend of the actual trajectories, and the predicted results from the model that considers driving style are closer to the actual trajectories. The prediction performance is improved compared to prediction models that do not consider driving style when the prediction period is greater than 3 s. The model also performs better in predicting car-following trajectories compared to lane-changing trajectory. Comparing the data in Tables 1 and 2, it can be seen that the prediction model considering driving style has smaller lateral MAE and longitudinal MAE and RMSE in the prediction results in different prediction periods, indicating that the prediction model considering driving style has a better prediction performance. However, the data in both Tables 1 and 2 indicate that, regardless of whether driving style is considered, the trajectory prediction errors from the model tend to accumulate and increase as the prediction time horizon extends. Significant prediction deviations are observed after a predicting period greater than 5 s.
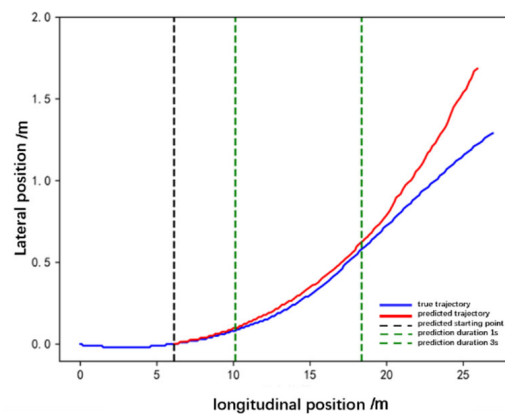
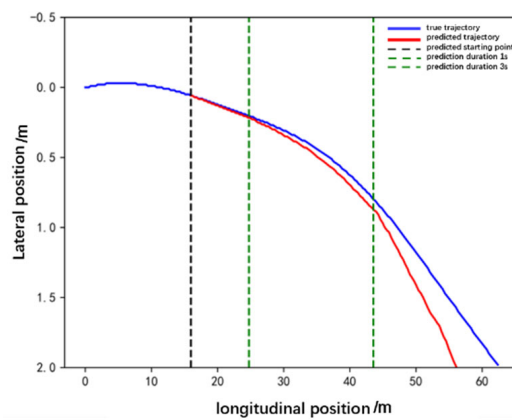**Figure 10.** Prediction results of left lane-changing.

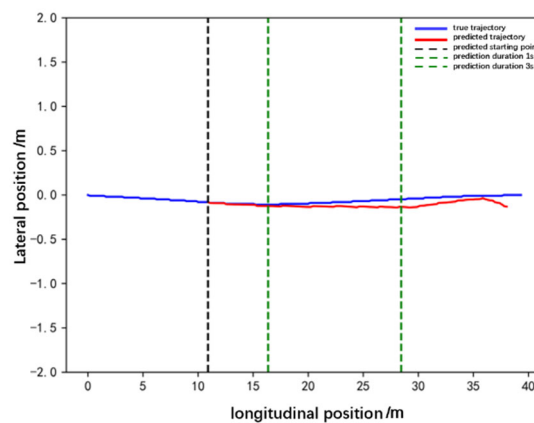**Figure 11.** Prediction results of right lane-changing.

Figure 12. Prediction results of car following.

5.3. Prediction Results with Other Similar Models

In order to further verify the predictive performance of the model established in this study compared to traditional models, the prediction model considering driving style with a better predictive performance was selected and compared to the LSTM models in references [4,15].

The "CTRA" is a prediction method based on the CTRA model in reference [2], the "ST_LSTM" is the LSTM prediction model considering spatiotemporal factors in reference [15], and the "IS_LSTM" is the prediction model considering driving style established in this study. RMSE is used as a comparative indicator, the predictive performance of the three models is compared, as shown in Table 3.

Table 3. RMSE comparison of models.

Prediction Period	1 s	2 s	3 s	4 s	5 s
CTRA	0.2371	0.8336	2.2733	5.0723	9.6216
ST_LSTM	0.1634	0.6712	1.5282	2.9184	4.3549
IS_LSTM	0.1186	0.5308	1.1245	2.8613	4.2425

According to Table 3, the trajectory prediction method based on kinematic models (CTRA) can ensure a high prediction accuracy in a shorter prediction period. However, the prediction error rapidly increases as the prediction period increases, which shows that the reliability of a long prediction period is poor. Comparing two similar LSTM models, the advantage of the prediction model of this study is more obvious when the prediction period is less than 5 s. However, after more than 5 s, the prediction error of both models also increases rapidly, and the advantage of our model gradually weakens.

6. Conclusions

The accurate prediction of vehicle-driving trajectories can support driving risk assessment and assistive driving systems. In this study, an LSTM network based on the Keras framework was constructed using the TensorFlow deep-learning platform. The impact of driving intention on trajectory prediction was considered by incorporating it along with historical trajectory data as inputs to the LSTM network. Additionally, to study the influence of driving style on trajectory prediction results, the lateral and longitudinal MAE and RMSE metrics were selected to evaluate the predictive performance of the model, a comparative study on the predictive performance of different models was conducted. The following conclusions were drawn:

(1) By comparing the values of the MAE and RMSE, it can be concluded that the vehicle-driving trajectory prediction model based on the LSTM in this study has a high prediction performance; the predicted trajectory is basically consistent with the actual trajectory.

(2) By adding the driving style as a new parameter in the LSTM model, it was found that trajectory prediction results made with the consideration of driving style exhibit better accuracy. It indicates that the driving style has a significant impact on the driving behavior and trajectory of vehicles. However, as the prediction time horizon increases, the accuracy of the predictions decreases, with noticeable errors emerging after 5 s.

(3) By comparing to the CTRA model based on kinematic models, it was found that the LSTM-based prediction model has a better predictive performance in a long prediction period. However, compared to the trajectory prediction model based on spatiotemporal LSTM, it was found that the model proposed in this study has significant advantages in a short prediction period (less than 3 s). However, as the prediction period increases, the prediction errors of both models rapidly increase.

The future predicted vehicle in this study can enhance the drivers' ability to judge and make decisions about danger. By predicting driving trajectory and analyzing the motion status and position of the target vehicle and surrounding vehicles during the predicted period, the possible collision risks can be judged, which can help drivers to improve driving safety and comfort. At the same time, trajectory prediction is also beneficial for optimizing advanced driving assistance systems and promoting the development of higher-level autonomous driving technology. However, the model in this study is trained based on vehicle-driving trajectories from highways, and the performance of the long-term prediction still needs to be improved. In future research, the following aspects of research can be considered to improve the predictive performance of trajectory prediction models: further improving the accuracy of the original data and exploring more complex models, such as deep-learning models (GRU, etc.) or ensemble learning methods (gradient boosting trees, etc.). Additionally, further attempts can be made to expand the application scope of the model, such as urban roads, intersections, and ramps.

References

1. Jin, Z.L.; Ran, L.L. The total number of motor vehicles in China has reached 435 million in 2023. *Beijing Business Daily*, 12 January 2024; p. 1.
2. Qu, D.Y.; Wang, S.J.; Liu, H.M.; Meng, Y.M. A car-following model based on trajectory data for connected and automated vehicles to predict trajectory of human-driven vehicles. *Sustainability* **2022**, *14*, 7045. [CrossRef]
3. Chen, X.Q.; Wei, C.X.; Yang, Y.; Luo, L.J.; Biancardo, S.A.; Mei, X.J. Personnel trajectory extraction from port-like videos under varied rainy interferences. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 6567–6579. [CrossRef]
4. Toledo-moreo, R.; Zamora-izquierdo, M.A. IMM-based lane-change prediction in highways with low-cost GPS/INS. *IEEE Trans. Intell. Transp. Syst.* **2009**, *10*, 180–185. [CrossRef]
5. Abbas, M.T.; Jibran, M.A.; Afaq, M.; Song, W.C. An adaptive approach to vehicle trajectory prediction using multimodel Kalman filter. *Trans. Emerg. Telecommun. Technol.* **2020**, *31*, e3734. [CrossRef]
6. Schubert, R.; Adam, C.; Obst, M.; Mattern, N.; Leonhardt, V.; Wanielik, G. Empirical evaluation of vehicular models for ego motion estimation. In Proceedings of the IEEE Intelligent Vehicles Symposium, Baden, Germany, 5–7 June 2011.
7. Zyner, A.; Worrall, S.; Nebot, E. A recurrent neural network solution for predicting driver intention at unsignalized intersections. *IEEE Robot. Automat. Lett.* **2018**, *3*, 1759–1764. [CrossRef]
8. Xin., L.; Wang, P.; Chan, C.Y.; Li, S.E.; Cheng, B. Intention-aware long horizon trajectory prediction of surrounding vehicles using dual LSTM networks. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018.
9. Nishiwaki, Y.; Miyajima, C.; Kitaoka, N.; Terashima, R.; Wakita, T.; Takeda, K. Generating lane-change trajectories of individual drivers. In Proceedings of the 2008 International Conference on Vehicular Electronics and Safety, Columbus, OH, USA, 22–24 September 2008.
10. Xie, G.; Gao, H.B.; Qian, L.J.; Huang, B.; Li, K.Q.; Wang, J.Q. Vehicle trajectory prediction by integrating physics- and maneuver-based approaches using interactive multiple models. *IEEE Trans. Ind. Electron.* **2017**, *65*, 5999–6008. [CrossRef]
11. Qiao, S.J.; Han, N.; Zhu, X.W.; Shu, H.P.; Zheng, J.L.; Yuan, C.A. A dynamic trajectory prediction algorithm based on Kalman filter. *Acta Electron. Sinica* **2018**, *2*, 418–423.
12. Zhang, R.F.; Cao, L.B.; Bao, S.; Tan, J.J. A method for connected vehicle trajectory prediction and collision warning algorithm based on V2V communication. *Int. J. Crashworthiness* **2017**, *22*, 15–25. [CrossRef]
13. Xiao, W.; Zhang, L.; Meng, D. Vehicle trajectory prediction based on motion model and maneuver recognition model and maneuver model fusion with interactive multiple models. *SAE Int. J. Adv. Curr. Pract. Mobil.* **2020**, *2*, 3060–3071. [CrossRef]
14. Xu, G.Q.; Liu, L.; Ou, Y.S.; Song, Z.J. Dynamic modeling of driver control strategy of lane-change behavior and trajectory planning for collision prediction. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 1138–1155. [CrossRef]
15. Xie, F.; Lou, J.T.; Zhao, K.; Qi, Y. A research on vehicle trajectory prediction method based on behavior recognition and curvature constraints. *Auto. Eng.* **2019**, *41*, 1036–1042.
16. Xing, R.R.; Zhang, Y.H.; Cai, X.Y.; Lu, J.P.; Peng, B.; Yang, T. Vehicle-trajectory prediction method for an extra-long tunnel based on section traffic data. *Sustainability* **2023**, *15*, 6732. [CrossRef]
17. Li, G.; Zhao, L.; Wu, L.; Jiang, C.M.; Tang, W.Y. Investigating gap acceptance behavior based on correlated random parameter survival model with heterogeneity in means. *Transp. Lett.* **2024**, *16*, 2335742. [CrossRef]
18. Chen, X.Q.; Dou, S.T.; Song, T.Q.; Wu, H.F.; Sun, Y.; Xian, J.F. Spatial-temporal ship pollution distribution exploitation and harbor environmental impact analysis via large-scale AIS data. *J. Mari. Sci. Eng.* **2023**, *12*, 960. [CrossRef]

19. Xiao, G.N.; Wang, Y.Q.; Wu, R.J.; Li, J.P.; Cai, Z.Y. Sustainable maritime transport: A review of intelligent shipping technology and green port construction applications. *J. Mar. Sci. Eng.* **2024**, *12*, 1752. [CrossRef]

20. Dai, S.; Li, L.; Li, Z. Modeling vehicle interactions via modified LSTM models for trajectory prediction. *IEEE Access* **2019**, *7*, 38287–38296. [CrossRef]

21. Wen, H.Y.; Zhang, W.G.; Zhao, S. Vehicle lane-change trajectory prediction model based on generative adversarial networks. *J. South China Univ. of Technol.* **2020**, *48*, 32–40.

22. Yang, D.; Liu, J.W.; Zheng, B.; Sun, F. A vehicle lane-changing trajectory prediction model based on temporal convolutional networks and attention mechanism. *J. Trans. Syst. EIT* **2024**, *24*, 114–126.

23. Xue, Q.W.; Xing, Y.Y.; Lu, J. An integrated lane change prediction model incorporating traffic context based on trajectory data. *Transp. Res. Part C Emerg. Technol.* **2022**, *141*, 103738. [CrossRef]

24. Guo, H.Y.; Keyvan-Ekbatani, M.; Xie, K. Lane change detection and prediction using real-world connected vehicle data. *Transp. Res. Part C Emerg. Technol.* **2022**, *142*, 103785.

25. Deo, N.; Trivedi, M.M. Convolutional social pooling for vehicle trajectory prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018.

26. Lin, L.; Li, W.Z.; Bi, H.K.; Qin, L.Q. Vehicle trajectory prediction using LSTMs with spatial-temporal attention mechanisms. *IEEE Intell. Transp. Syst. Mag.* **2022**, *14*, 197–208. [CrossRef]

27. Liang, M.; Yang, B.; Hu, R.; Chen, Y.; Liao, R.J.; Feng, S.; Urtasun, R. Learning lane graph representations for motion forecasting. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020.

28. Messaoud, K.; Yahiaoui, I.; Verroust-Blondet, A.; Nashashibi, F. Attention based vehicle trajectory prediction. *IEEE Trans. Intell. Vehicles.* **2021**, *6*, 175–185. [CrossRef]

29. Xie, D.F.; Fang, Z.Z.; Jia, B.; He, Z.B. A data-driven lane-changing model based on deep learning. *Transp. Res. Part C Emerg. Technol.* **2019**, *106*, 41–60. [CrossRef]

30. Ji, X.W.; Fei, C.; He, X.K.; Liu, Y.L.; Liu, Y.H. Intention recognition and trajectory prediction for vehicles using LSTM network. *China J. Highw. Transp.* **2019**, *32*, 34–42.

31. Shi, K.S.; Wu, Y.K.; Shi, H.T.; Zhou, Y.; Ran, B. An integrated car-following and lane changing vehicle trajectory prediction algorithm based on a deep neural network. *Phys. A Stat. Mech. Appl.* **2022**, *599*, 127303. [CrossRef]

32. Zhao, B.; Lin, Y.L.; Hao, H.J.; Yao, Z.H. Fuel consumption and traffic emissions evaluation of mixed traffic flow with connected automated vehicles at multiple traffic scenarios. *J. Adv. Transp.* **2022**, 6345404. [CrossRef]

33. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Serdyuk, D.; Chambliss, J.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014.

34. Shen, Q.R.; Xie, X.; Li, G.; Wu, L.Z.; Yang, Z. Study on ring-road incident duration based on latent class accelerated hazard model. *PLoS ONE* **2024**, *19*, e0308473. [CrossRef] [PubMed]

35. Huang, H.; Zen, Z.N.; Yao, D.Y.; Pei, X.; Zhang, Y. Spatial-temporal convLSTM for vehicle driving intention prediction. *Tsinghua Sci. Technol.* **2022**, *27*, 599–609. [CrossRef]

36. Li, G.; Yang, Z.; Pan, Y.Y.; Ma, J.X. Analysing and modelling of discretionary lane change duration considering driver heterogeneity. *Transp. B-Transp. Dyn.* **2023**, *11*, 343–360.

37. Luo, Q.; Liu, S.L.; Luo, S.Q.; Guo, X.Y.; Rong, J.; Li, J.H. Research on driving intention recognition method based on guassian mixed model and hidden markov model. *Transp. Eng.* **2024**, *24*, 29–33.