WILEY | Hindawi

Research Article

Real-Time Vehicle Trajectory Prediction for Traffic Conflict Detection at Unsignalized Intersections

Qianxia Cao [ID],[1] **Zhongxing Zhao** [ID],[2] **Qiaoqiong Zeng** [ID],[2] **Zhengwu Wang** [ID],[2]
and Kejun Long [ID][3]

[1]*Key Laboratory of Highway Engineering of Ministry of Education, Changsha University of Science and Technology,*
 Changsha 410114, China
[2]*School of Traffic & Transportation Engineering, Changsha University of Science and Technology, Changsha 410114, China*
[3]*Hunan Key Laboratory of Smart Roadway and Cooperative Vehicle-Infrastructure Systems,*
 Changsha University of Science and Technology, Changsha 410114, China

Correspondence should be addressed to Qianxia Cao; qianxiacao@gmail.com

Real-time prediction of vehicle trajectory at unsignalized intersections is important for real-time traffic conflict detection and early warning to improve traffic safety at unsignalized intersections. In this study, we propose a robust real-time prediction method for turning movements and vehicle trajectories using deep neural networks. Firstly, a vision-based vehicle trajectory extraction system is developed to collect vehicle trajectories and their left-turn, go straight, and right-turn labels to train turning recognition models and multilayer LSTM deep neural networks for the prediction task. Then, when performing vehicle trajectory prediction, we propose the vehicle heading angle change trend method to recognize the future move of the target vehicle to turn left, go straight, and turn right based on the trajectory data characteristics of the target vehicle before passing the stop line. Finally, we use the trained multilayer LSTM models of turning left, going straight, and turning right to predict the trajectory of the target vehicle through the intersection. Based on the TensorFlow-GPU platform, we use Yolov5-DeepSort to automatically extract vehicle trajectory data at unsignalized intersections. The experimental results show that the proposed method performs well and has a good performance in both speed and accuracy evaluation.

1. Introduction

At unsignalized intersections, the traffic volume is small and there is no traffic signal control. Conflicts between traffic flows at unsignalized intersections cannot be effectively separated in time and space, leading to traffic safety issues that cannot be ignored. By judging the conflict points between vehicles in advance and prompting the driver to take measures to avoid risks, the safety level of unsignalized intersections can be effectively improved. Vehicle trajectory prediction is an important part of realizing conflict warning. Based on the predicted trajectory's arrival time to collision (TTC), postencroachment time (PET), gap time (GT), and other parameters, the position of the conflict point that exceeds the safety threshold can be extracted to determine the risk of the conflict point, and then the conflict warning can be carried out.

Currently, relevant scholars are mostly working on methods for predicting vehicle trajectories in autonomous driving scenarios. The main methods of vehicle trajectory prediction in the autonomous driving environment are divided into methods based on physical models and methods based on trajectory data. The methods based on physical models take motion as the starting point and construct dynamic or kinematic models based on expert knowledge [1–3]. In [1], the maximum curvature of the trajectory and the obstacle avoidance path planner based on the parameter cubic Bezier curve are defined. Pool et al. developed a motion mixture model through probabilistic filters for cyclist path prediction and used the local road topology to obtain a better prediction distribution [2]. Xie et al. used the lane line curvature as a constraint to predict the trajectory of the vehicle in the next few seconds by the constructed cubic Bezier curves while combining the vehicle state information

and applying the constant turn rate and acceleration model (CTRA model) to form a weighting function to filter the best-predicted trajectory [3].

The methods based on trajectory data use deep learning or nondeep learning to analyze large amounts of historical data and to make trajectory predictions. Nondeep learning algorithms include implicit Markov models, regression models, Kalman filters, and Gaussian processes. The extended Kalman filter (EKF) [4] and Monte Carlo are proven to have good accuracy in short-term trajectory prediction. For example, Kawasaki and Tasaki proposed an intersection turning vehicle trajectory prediction method [5]. The speed and intersection geometry are considered, and the vehicle speed is assumed to be minimized before crossing the crosswalk. Finally, the ideal speed model is combined with the extended Kalman filter to predict the future vehicle position in multiple steps.

The rapid development of deep learning in recent years has brought new ideas to trajectory prediction. Recurrent neural networks (RNNs), long short-term memory (LSTM), and gated recurrent units (GRUs) have been successfully applied in time-series data analysis. In terms of deep learning model selection, many scholars have proposed unique deep learning algorithms based on application scenarios. Some scholars use the RNN to predict vehicle trajectory data [6]. However, the RNN structure cannot remember the long-term information state, and the gradient disappears or the gradient explodes during reverse training, making the network lose its learning ability. LSTM can better avoid these problems, so it is more popular. For example, Chen et al. proposed a vehicle trajectory prediction model based on the LSTM encoder–decoder [7]. The model uses three layers of different LSTMs to capture the information of spatial, temporal, and trajectory data. The information is spliced into the entire context vector, and finally, the trajectory is predicted by the decoder. Alahi et al. proposed the "social-LSTM" structure [8], which allows LSTM that is adjacent in space to share each other's hidden state, thereby capturing the dependencies between multiple related sequences. Ji et al. proposed a vehicle trajectory prediction model based on LSTM [9]. The model first uses the softmax function to determine the driving intention and then uses LSTM to predict the vehicle trajectory. Luo et al. proposed a target-oriented lane attention trajectory prediction model [10]. Its trajectory coding module uses two standard LSTMs: one is used to encode historical position, and the other is used to encode historical speed. The two extracted features are connected to predict the motion feature of the vehicle. Besides LSTM, some scholars are also doing research on a two-stage trajectory prediction network (TPNet) [11]. In the first stage, they extract basic features from the trajectory data. In order to narrow the search range, they predict a rough end point, and this predicted end point is used to generate a recommended trajectory. In the second stage, they screen recommended trajectories based on historical trajectories and movable areas formed by high-precision maps. They find the most likely future trajectory from the recommended trajectories and then refine them to ensure the diversity of the final prediction. Moreover, Yao et al. proposed a bidirectional multimodal trajectory prediction method

(BiTrap) based on target estimation [12]. This model has achieved good results in predicting pedestrian trajectories from the first-person view (FPV) and bird's-eye view (BEV) scenarios.

In terms of deep learning data acquisition, there are many ways to acquire traffic data, such as video and loop detector. Feng et al. [13] proposed a cross-frame target association algorithm under the constraints of vehicle dynamics and trajectory confidence using Yolov5. Chen et al. [14] proposed a method to extract vehicle trajectories automatically and accurately from aerial video. This method uses wavelet transform to denoise the Frenet coordinate data and eliminate the deviation of the vehicle trajectory position. In addition, some scholars use OpenCV 2.3 [15] to collect turning motion trajectories to train DNNs and LSTM networks for early trajectory prediction in the next 2 s. In terms of deep learning data processing, the original traffic flow data may be polluted by noise during the data collection process. In this way, noise data will significantly affect the performance of traffic flow prediction. In this case, some scholars use noise reduction processing on the original traffic flow data to obtain better prediction results. Jiang et al. [16] proposed the Savitzky–Golay filter to filter the noise of the NGSIM (I-80) dataset. They used three deep neural networks, long short-term memory (LSTM), gated recurrent unit (GRU), and stacked autoencoder (SAE), to predict the position and speed of the advancing vehicle. In addition, empirical mode decomposition (EMD), ensemble empirical mode decomposition (EEMD), and wavelet (WL) have also been applied to remove the noise of traffic flow data [17]. According to the characteristics of different datasets, some scholars solve actual traffic problems from a unique perspective according to the scenarios. Kim et al. [18] divided the road environment on which the vehicle travels into an occupancy grid map, expressed the predicted trajectory of the vehicle as the occupancy probability on the occupancy grid map, and used the LSTM network structure to generate the future vehicle occupancy probability on the occupancy grid map. Mirus et al. [19] studied the influence of the composition of the training dataset on the neural network-based vehicle trajectory prediction model. The research results of this study show that the training effect of the LSTM model that combines driving scenarios with classification training is better than that of the LSTM model that does not distinguish between scenarios.

At present, there are few research studies on real-time prediction methods of the vehicle trajectory in actual manual driving scenarios. From the perspective of the domestic and the foreign traffic environment, autonomous driving technology has not been widely used in most areas. It is undeniable that autonomous driving is a development trend. But for now, the research on vehicle trajectory prediction in autonomous driving scenarios cannot be applied to the current vehicle conflict detection at unsignalized intersections. The study of vehicle trajectory prediction under manual driving scenarios and its application in the field of conflict warning can quickly apply the research results to practice and greatly improve the safety level of unsignalized intersections.

The problems faced are as follows: (1) existing vehicle trajectory data based on UAV, GPS, driving simulation, and other sources cannot realize real-time detection and prediction of the vehicle trajectory. Based on a fixed video surveillance system, real-time detection and prediction of vehicle trajectories can be achieved, but there is currently no corresponding vehicle trajectory dataset. (2) Multiobjective and long-term real-time vehicle trajectory prediction methods are still being explored, and there are many challenges in improving the prediction accuracy and speed.

The main goal of our research is to extract the vehicle trajectory data at the entrance lane of an unsignalized intersection in real time and then predict the turning and trajectory of the vehicle to further detect vehicle conflicts at the intersection. The main academic contributions of this research are as follows: (1) a real-time vehicle trajectory prediction framework for traffic conflict detection at unsignalized intersections based on road surveillance videos is proposed. (2) We propose a vehicle turning intention recognition method based on the change trend of the vehicle heading angle at the entrance of an unsignalized intersection and a vehicle trajectory prediction method based on a multilayer LSTM model. (3) We extract thousands of vehicle trajectory data from surveillance videos of unsignalized intersections and generate a vehicle trajectory dataset facing the road-monitoring perspective.

Our research is organized as follows: Section 2 describes the proposed real-time prediction method in detail, Section 3 explains the data source and experimental process and results, and Section 4 briefly summarizes the research results.

2. Materials and Methods

Vehicle trajectory prediction at unsignalized intersections is a multitarget trajectory prediction problem. The focus is on predicting the trajectory of each vehicle. The trajectory prediction of a single vehicle can be divided into two stages. First, according to the detected vehicle's trajectory characteristics at the entrance of the intersection, it is judged whether the turning intention of the vehicle entering the intersection is straight, left, or right. Second, the historical vehicle trajectory data are used to predict the track position of the vehicle passing the intersection.

For the first stage, accurately identifying the vehicle's turning intention can provide an important guarantee for the accuracy and reliability of the real-time prediction of the vehicle trajectory. In this paper, we consider the vehicle's heading angle at the entrance lane as the turning feature and use the vehicle heading angle change trend to recognize the vehicle's steering intention. For the second stage, the real-time vehicle trajectory position prediction is a time-series prediction, so we consider using LSTM to build a vehicle trajectory position prediction model. The overview of our method is shown in Figure 1.

The vehicle trajectory data are time-series data. We first extract the historical trajectory of vehicles passing through the intersection from the surveillance video. The trajectories can be divided into three types: left turn, going straight, and right turn. Using these three trajectory datasets to train the LSTM, we can get the left-turn LSTM, straight-going LSTM, and right-turn LSTM models. Then, according to the real-time detected trajectory of the target vehicle at the entrance of the intersection, the vehicle's heading angle change trend feature is used to identify the vehicle's turning intention. Then, according to the identified turning label of the target vehicle, the trained LSTM model corresponding to the turning label is used for further trajectory prediction.

2.1. Model Training. The model we use is a multilayer LSTM model (see Figure 2), and its hidden layer contains $N-1$ dropout layers, $N-1$ LSTM layers, and one dense layer. The historical vehicle trajectory data of the same entrance at the same intersection extracted from a video surveillance camera are divided into three types: left-turn dataset, straight-going dataset, and right-turn dataset. Through training separately, the left-turn LSTM, straight-going LSTM, and right-turn LSTM models of each entrance of the intersection can be obtained.

2.2. Turning Intention Prediction. First, we use the historical trajectory data of the vehicle at the entrance lane of the intersection to calculate the heading angle of each displacement of the vehicle. Then, we select the heading angle data of the appropriate length to generate a sliding pane and perform a univariate linear regression on the data in the sliding pane. Finally, according to the actual situation of the intersection, the classification conditions of the vehicle turning left, going straight, and turning right are adjusted according to the data characteristics such as the regression slope and change trend. When we perform vehicle turning intent recognition, we extract the vehicle heading angle feature based on the video trajectory data detected in the entrance lane in real time and combine the adjusted classification conditions to accurately and stably identify the turning intention of the vehicle. The heading angle calculation formula [20] is as follows:

$$\theta_t(a_t, a_{t+1}) = \arctan\left(\frac{y_a^{t+1} - y_a^t}{x_a^{t+1} - x_a^t}\right), \tag{1}$$

where θ_t is the heading angle and (x_a^t, y_a^t) is the coordinate of the vehicle "a" at time t.

2.3. Trajectory Prediction. After obtaining the turning label of the target vehicle trajectory, based on the real-time detection of the target vehicle trajectory data at the entrance of the intersection, we use the trained multilayer LSTM model consistent with the turning label of the target vehicle to predict the trajectory of the vehicle passing the intersection. The forecasting process is shown in Figure 3.

3. Results and Discussion

Based on the surveillance video of an unsignalized intersection, we use Yolov5-DeepSort to extract vehicle trajectories and obtain information such as the ID of each vehicle
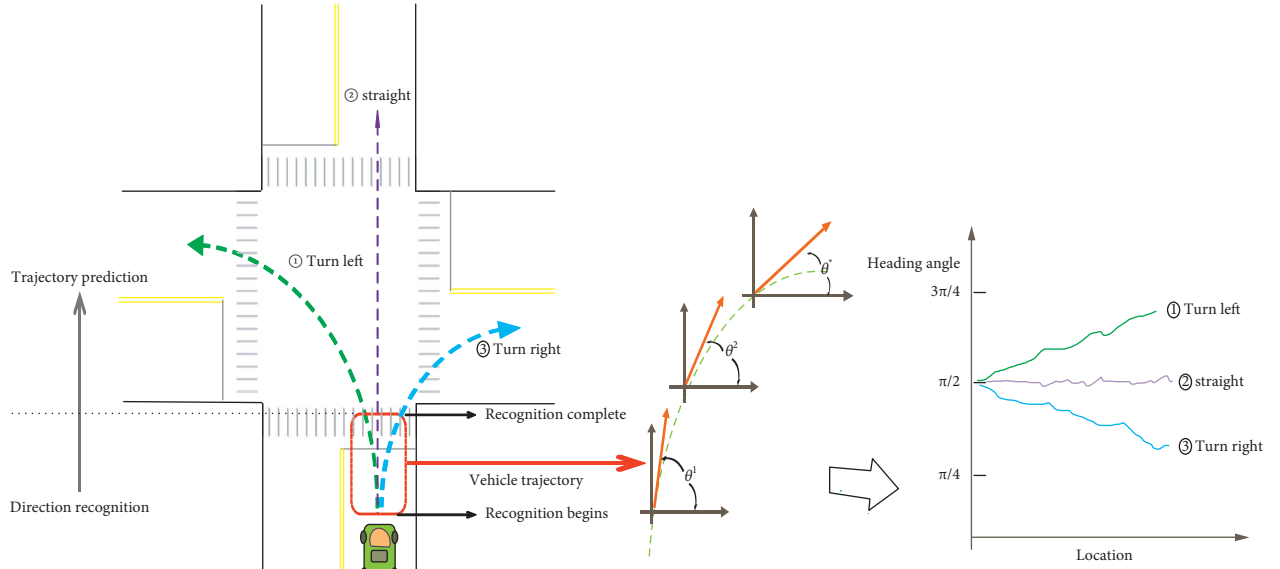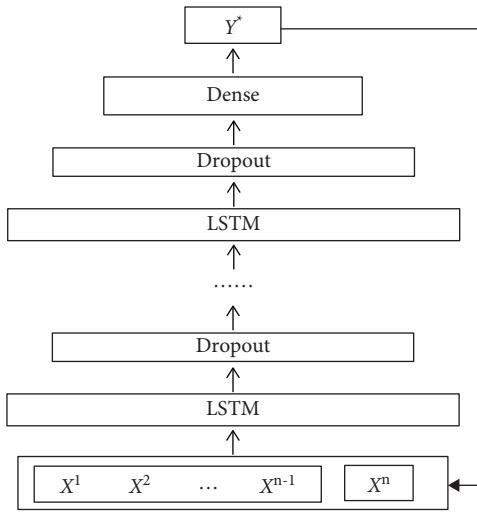
Figure 1: Overview of our method.

Figure 2: Multilayer LSTM model.

and the trajectory coordinates through the intersection. Then, after the video trajectory data are processed, the trajectory of the stopped vehicle and the abnormal trajectory of the vehicle are removed, and the trajectory data are automatically labeled with the turning category. Then, we use the processed trajectory data for training and prediction to demonstrate the effectiveness of the model proposed in the study. The model is implemented on one TensorFlow GPU.

3.1. Datasets

3.1.1. Data Acquisition. The vehicle trajectory data selected in the experiment come from the traffic flow videos of unsignalized intersections taken by video surveillance cameras. The total length of the captured video is about 4

hours, the resolution is $1920 * 1080$, and the video frame rate is 25 frames per second. We use Yolov5-DeepSort to detect and track vehicles at the intersection. Yolov5 [21] detects and recognizes vehicles in each frame of the video. DeepSort assigns a unique vehicle ID and tracks the same vehicle in real time, thereby obtaining the trajectory data of all vehicles passing through the intersection. The vehicle detection and tracking results are shown in Figure 4, and the extracted part of the trajectory is shown in Figure 5.

3.1.2. Data Processing. The trajectory data obtained through video detection are affected by the speed, driving path, and body size of each vehicle. They are also affected by objective factors such as weather, angle, and pixels at the time of video shooting. Therefore, the extracted trajectory data need further processing. We clean the data, eliminate abnormal trajectories, and extract complete and normal vehicle trajectory data for model verification. Examples of extracted trajectory data are shown in Table 1.

3.1.3. Direction Label. In order to obtain the turning label of each vehicle's trajectory data in the video, we draw the recognition area at each entrance and exit position of the intersection in the video and obtain the pixel coordinates of the recognition area. The rule of whether to enter and exit the recognition area is as follows: every time we obtain a new vehicle coordinate at the entrance of the intersection, the coordinate will be automatically corresponded to the corresponding position of the recognition area, and then we can judge which recognition area of the intersection the vehicle is passing through according to the vehicle coordinates. The matching rules for turning labels are as follows: when the target vehicle appears in the intersection recognition area M for the first time, and after a certain period, if it appears in the intersection recognition area N for the second time, then
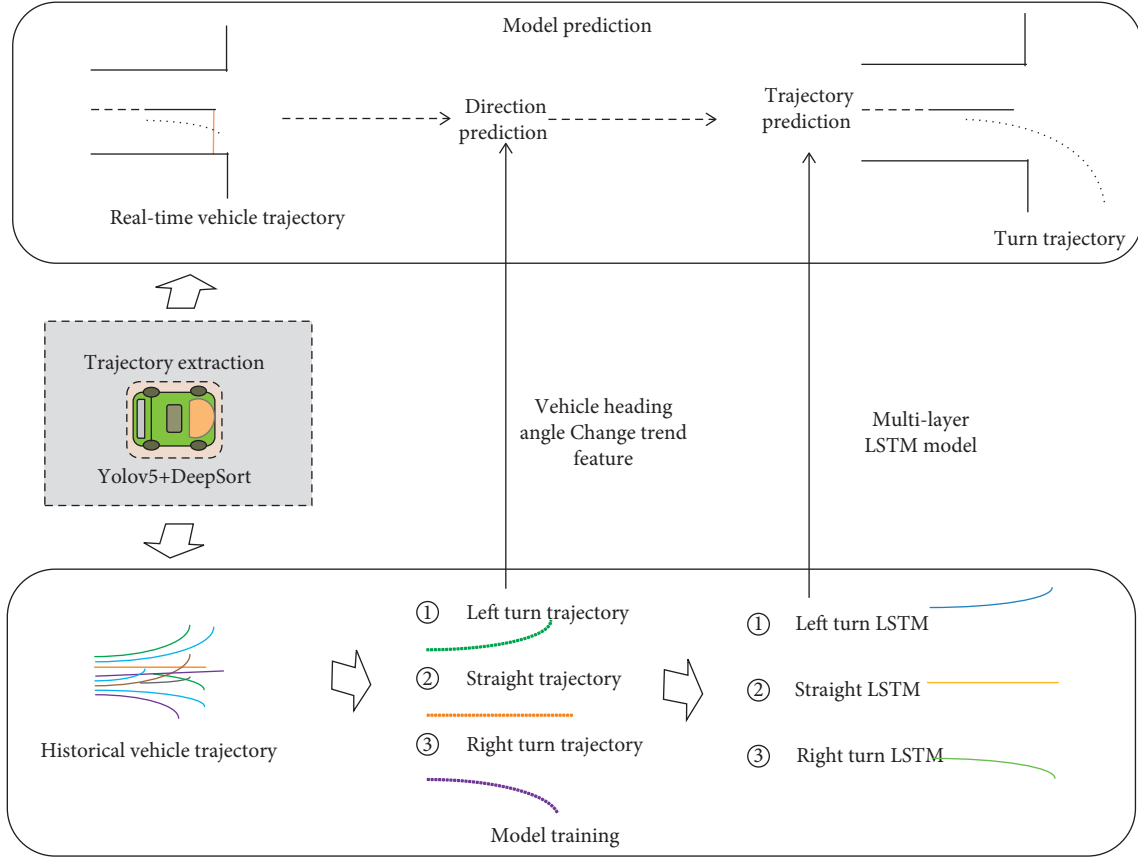
FIGURE 3: Schematic diagram of the forecasting method.

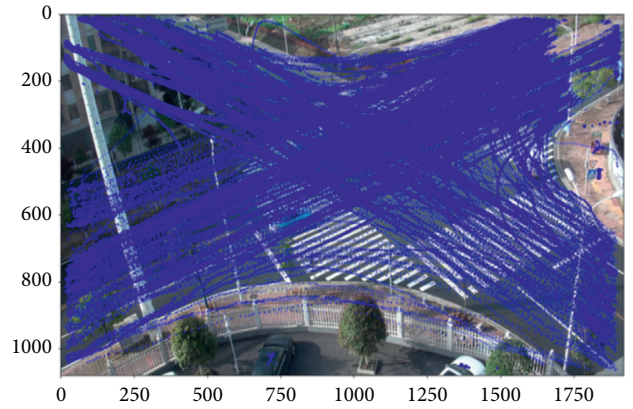FIGURE 4: Vehicle detection and tracking results.

FIGURE 5: Part of the extracted vehicle trajectory.

we will think that the vehicle enters the intersection from the M entrance and leaves from the N exit, and the turning label format is set to (M, N). After the turning recognition is completed, the trajectory data are divided into a straight-going dataset, a left-turn dataset, and a right-turn dataset according to the turn label. The recognition area is drawn as shown in Figure 6.

3.1.4. Experimental Dataset. To establish a trajectory dataset, we need to unify the number of trajectory points for each trajectory, so that the trajectory data form a matrix with a fixed size and no null values and meet the requirements for

vehicles to pass through the intersection completely. According to the actual situation of the intersection, we select 4 seconds after the vehicle passes the stop line as the trajectory coordinate prediction range, that is, the coordinate data of 100 trajectory points after the vehicle passes the stop line.

3.2. Model Training. In order to better verify the effectiveness of our model, the trajectory data of the same entrance at the same intersection are selected in the experiment to verify the model. There are 1030 vehicle trajectory data in the

TABLE 1: Examples of extracted trajectory data.

Vehicle ID	Turn direction	Coordinate	1	2	3	4	5
1	(1, 2)	x	385.43	397.58	409.47	420.45	431.65
1	(1, 2)	y	195.02	195.81	196.43	196.12	196.95
2	(4, 2)	x	378.77	393.54	407.71	421.81	435.95
2	(4, 2)	y	101.68	102.07	102.45	101.13	100.6
3	(3, 1)	x	165.43	180.47	195.27	210.39	224.41
3	(3, 1)	y	188.35	189.9	190.33	191.12	192.01
4	(1, 4)	x	252.1	264.86	277.81	290.98	303.35
4	(1, 4)	y	175.02	175.83	176.71	177.1	178.06
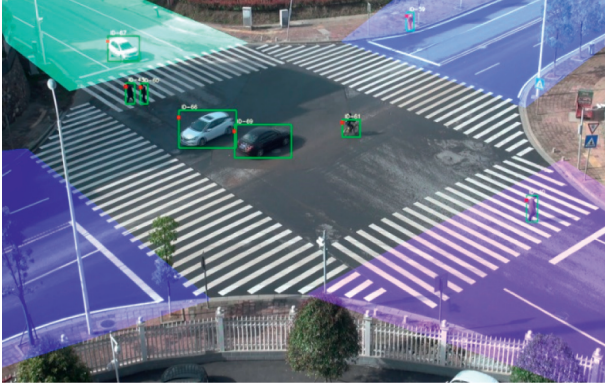
FIGURE 6: Recognition area of the entrance and exit of the intersection.

FIGURE 7: Scatter plot of the regression slope of the heading angle for each turn.

selected entrance lanes, including 301 left-turn, 406 straight-going, and 323 right-turn datasets. We first normalize the trajectory data and then use the dedimensionalized trajectory data to train the left-turn, straight-going, and right-turn LSTM models of the entrance lane. We use 80% of the trajectory data as the training set and 20% as the validation set.

3.2.1. Turn Recognition Training.

We adjust the parameters of the vehicle heading angle change trend recognition algorithm according to the terrain characteristics of the unsignalized intersection. The adjustable recognition algorithm parameters are the starting recognition position, the length of the recognition area, the size of the sliding pane, and the sliding step length.

After multiple rounds of training, we choose 12 meters before the stop line as the starting position for recognition and choose 50 trajectory coordinate points after the starting position for heading angle calculation. The sliding pane is 20, and the sliding step is 2. The judgment rule for turning recognition is as follows: when the regression slope of the heading angle exceeding 2/3 falls between −0.25 and 0.25, it is the intention to go straight; when the regression slope of the heading angle exceeding 2/3 falls between 0.1 and 2, it is the intention to turn right; and when the regression slope of the heading angle exceeding 6/10 falls between −2 and −0.2, it is the intention to turn left. Refer to Figure 7 for the regression slope scatter diagram of each steering heading angle. The scatter plot of the regression slope of each turning heading angle is shown in Figure 7.
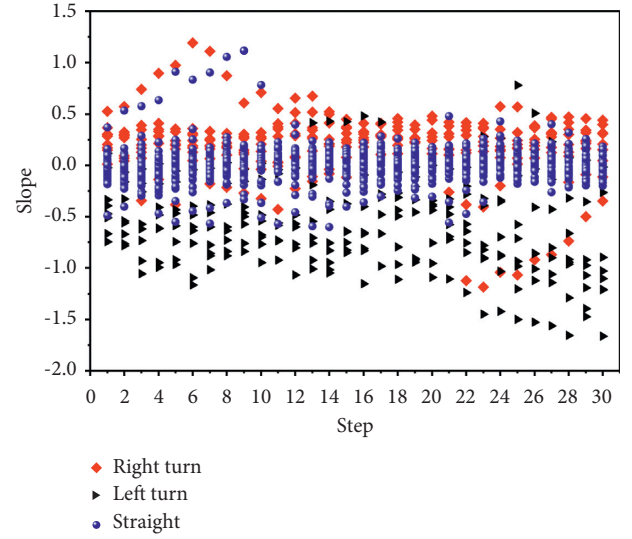
3.2.2. Multilayer LSTM Model Training.

After normalizing the vehicle trajectory data in the training set and eliminating dimensions, we select the appropriate number of LSTM model layers, sliding pane length, and prediction step length through training.

First, we input the trajectory data into the LSTM model of different numbers of layers for experiments and then use the LSTM model with the optimal number of layers to test sliding panes of different lengths and the prediction step size. The optimizer selected for each LSTM model is Adam, the error calculation method is MSE (mean square error), and the accuracy evaluation standard is ACC.

During the model training process, we find that adding a layer of LSTM will cause the training time to increase at a rate of 0.3 to 0.8 times. When we choose too few LSTM layers, we cannot learn the data completely, and when we choose too many LSTM layers, the training time will be too long. Therefore, if we want to achieve the purpose of improving training accuracy and shortening training time, we must select the appropriate number of layers for the multilayer LSTM according to the dataset. We also find that different sliding pane lengths will lead to different training errors. This is because when the sliding pane is too short, there will be too little input of known trajectory characteristic information, and when the sliding pane is too long, too much trajectory characteristic information will be input.

Finally, we find that the increase in the length of the predicted trajectory point has a slight impact on the training time, but it will cause an increase in the training error. This is because as the length of the predicted trajectory point increases, the trajectory feature information learned by the multilayer LSTM from the known trajectory is not enough to predict the long future trajectory point. According to the experimental results and considering the influence of various data, our parameter selection is shown in Table 2. The changes in the number of LSTM layers and the training accuracy are shown in Figure 8, the predicted coordinate length and the training mean square error are shown in Figure 9, and the predicted sliding pane length and the training mean square error are shown in Figure 10.

3.3. Model Prediction and Result Discussion

3.3.1. Turning Recognition Based on the Change Trend of Vehicle Heading Angle. We use the validation set data to test the vehicle heading angle change trend recognition algorithm and obtain the average recognition accuracy of each steering intention. Finally, we compare the vehicle heading angle change trend recognition algorithm with the KNN turning prediction algorithm, and the accuracy of the vehicle heading angle change trend recognition algorithm has been greatly improved. The vehicle heading angle change trend recognition algorithm makes full use of the geographic information of the intersection and recognizes the turning intention according to the heading angle change characteristics of each turning vehicle. The recognition result is only related to the characteristics of the intersection and the changing trend of the vehicle heading angle. It does not rely too much on the historical database with rich samples and has a good migration ability. Therefore, compared with the KNN algorithm, it has higher stability and practicability. The accuracy comparison between the vehicle heading angle change trend algorithm and the KNN algorithm is shown in Table 3.

3.3.2. Vehicle Trajectory Prediction Based on Four-Layer LSTM Model. After the target vehicle obtains the predicted turning label, we use the trained corresponding turning multilayer LSTM model to predict the future trajectory data of the target vehicle (100 trajectory points) based on the real-time detected target vehicle trajectory data at the entrance lane (30 trajectory points). Compared with the constant turn rate and acceleration (CTRA) vehicle model, the multilayer LSTM prediction model proposed in the thesis has obvious advantages in prediction accuracy and speed. The comparison of prediction accuracy and time-consuming data is shown in Table 4.

3.3.3. Result Discussion. According to the target vehicle's turning predicted by the vehicle heading angle change trend algorithm and the trajectory data of the target vehicle detected in real time at the entrance, the corresponding turning LSTM model is used to predict the future trajectory point.

The prediction result of going straight is shown in Figures 11–14. The prediction result of left turn is shown in Figures 15–18. The prediction result of right turn is shown in Figures 19–22.

According to Figure 11, the training accuracy of 97.22% of the straight-going LSTM models exceeds 94%, and the training accuracy of 63.89% of the straight-going LSTM models exceeds 97%. The training effect is good. According to Table 4 and Figures 12–14, in the pixel coordinate system, the absolute error between the predicted trajectory of the straight-going LSTM model and the actual trajectory is within 100, and the average absolute error is 45.784. The deviation between the trajectory predicted by the straight-going LSTM model and the actual trajectory is small. The prediction takes 1.87 seconds and has a real-time performance. The training accuracy, absolute prediction error, and prediction time of the straight-going LSTM model are all within acceptable ranges, and the experimental results are good.

According to Figure 15, the training accuracy of the left-turn LSTM model exceeds 97%, and the training effect is good. According to Table 4 and Figures 16 and 17, in the pixel coordinate system, the average absolute error between the predicted trajectory and the actual trajectory is 42.151. Among them, the prediction error of the first 50 steps does not exceed 40. According to Table 4 and Figure 18, 87.23% of the prediction errors are distributed within 100. The prediction effect in the first half is better. The prediction takes 2.02 seconds, and the prediction speed is slightly slow. By analyzing the trajectory, we can see that the driving distance for a left turn is longer, and there are more conflict points at the intersection. Moreover, the radius of the left turn is larger, and the trajectory direction has a certain degree of uncertainty. The prediction process is more complicated and requires higher computing power.

Figure 19 shows that the training accuracy of the right-turn LSTM model is more dispersed than that of the other two turning models. 60% of the models have a training accuracy of over 97%, and 90% of the models have a training accuracy of over 94%, which is slightly inferior to that of the other two turning models. According to Figures 20–22, we find that there is a certain deviation between the predicted trajectory of right turn and the actual trajectory. In the pixel coordinate system, the average absolute error between the predicted trajectory of the right-turn LSTM model and the actual trajectory is 73.21. 72.73% of the prediction deviations are distributed within 100. The prediction takes 1.96 seconds, which is better than that of the left-turn model. By analyzing the trajectory, we can see that the right-turn trajectory has a large turning amplitude and a small turning radius. Although there are fewer conflict points, the distance that the right-turning vehicle travels within the intersection is shorter, and fewer characteristic data are acquired.

In summary, by comparing the prediction models of left-turn, straight-going, and right-turn trajectories in terms of prediction error, prediction time consumption, and prediction model stability, we can see the following: (1) In terms of prediction errors, the average prediction error of the left-turn LSTM model is the smallest, the average prediction

TABLE 2: LSTM model parameters for each turn.

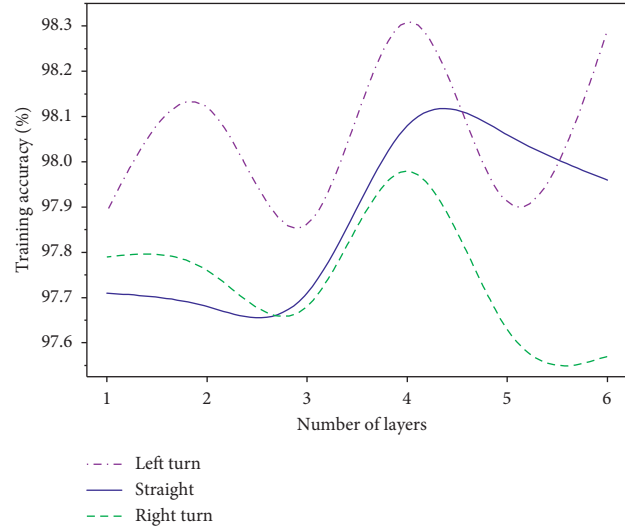| LSTM model	Number of LSTM layers	Sliding pane length	Prediction step
Going straight	4	6	1
Left turn	4	9	1
Right turn	4	12	1

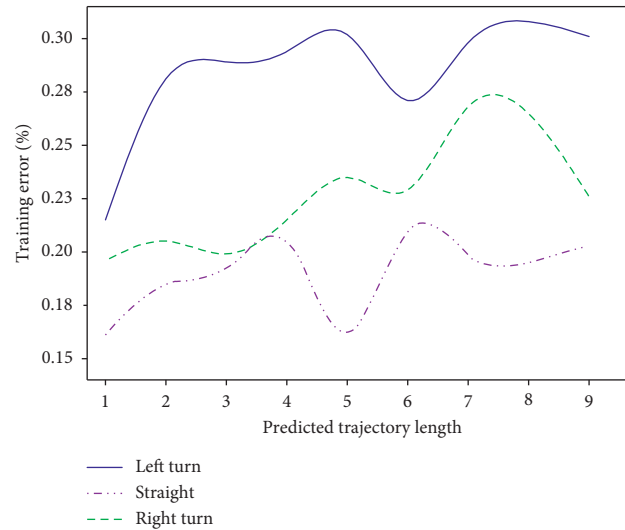FIGURE 8: Relationship between the number of LSTM layers and the training accuracy.

FIGURE 9: Relationship between the prediction step size and the training mean square error.

error of the straight-going LSTM model is not much different from that of the left-turn LSTM model, and the average prediction error of the right-turn LSTM model is slightly larger. (2) In terms of time-consuming prediction, the complexity of the straight-going trajectory is low, and the time-consuming prediction is the shortest. The trajectory of turning right and turning left is more complicated, the prediction time is about 2 s, and the prediction of turning left is slightly longer. (3) In terms of the stability of the prediction model, the standard deviation of the prediction error of the straight-going LSTM model is 21.233, the standard deviation of the prediction error of the left turn is 37.451, and the standard deviation of the prediction error of the right turn is 32.388. The result of straight-going trajectory prediction is relatively stable, and the prediction result of the left-turn trajectory is slightly worse than that of the right-turn trajectory.

The real-time vehicle trajectory prediction effect for traffic conflict detection at unsignalized intersections is shown in Figure 23.
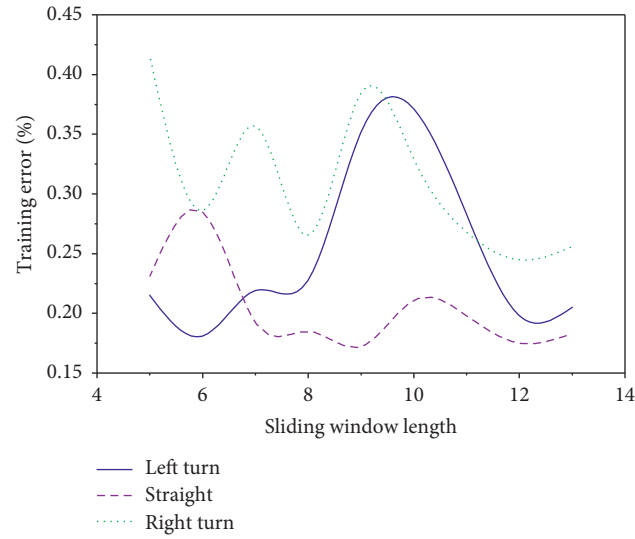
FIGURE 10: Relationship between the length of the sliding pane and the training mean square error.

TABLE 3: Comparison of accuracy of different turning recognition methods.

Methods	Accuracy		
	Going straight (%)	Left turn (%)	Right turn (%)
Ours	99.7	96.2	97.3
KNN	87.3	81.5	78.6

TABLE 4: Comparison of prediction algorithms.

Methods	Direction	MAE (axis)	Time (s)
Four-layer LSTM	Going straight	45.784	1.87
	Left turn	42.151	2.02
	Right turn	73.210	1.96
CTRA	Going straight	110.495	3.57
	Left turn	265.614	4.11
	Right turn	246.051	3.96
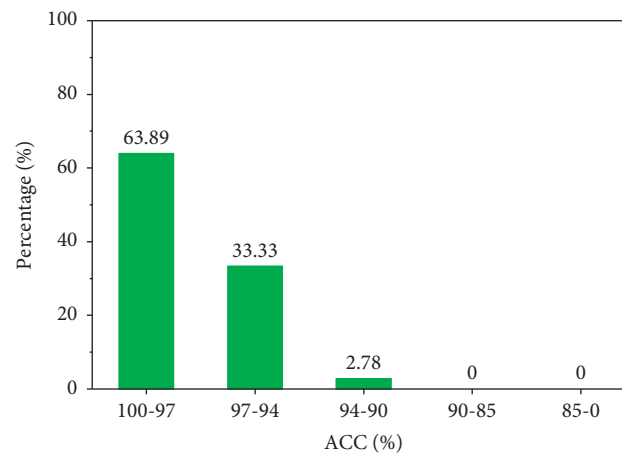
FIGURE 11: Training accuracy distribution for going straight.
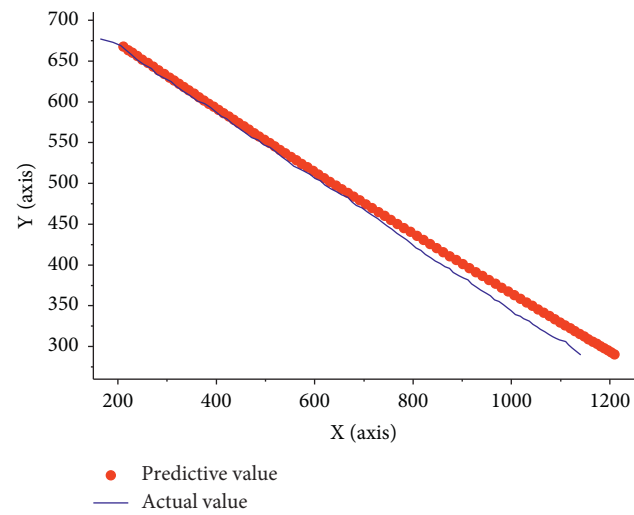
Figure 12: Prediction result of the straight trajectory.
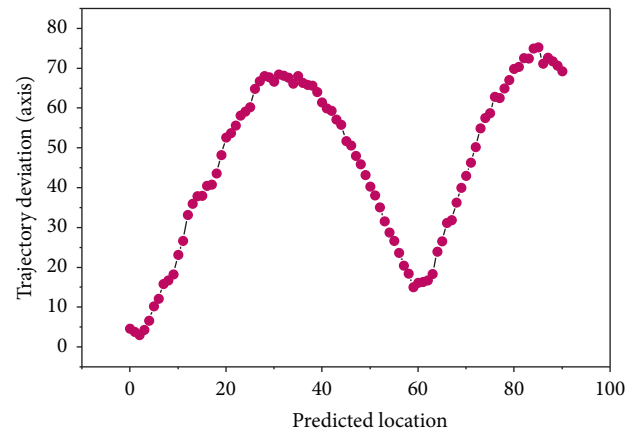
Figure 13: MAE scatter plot for straight trajectory prediction.
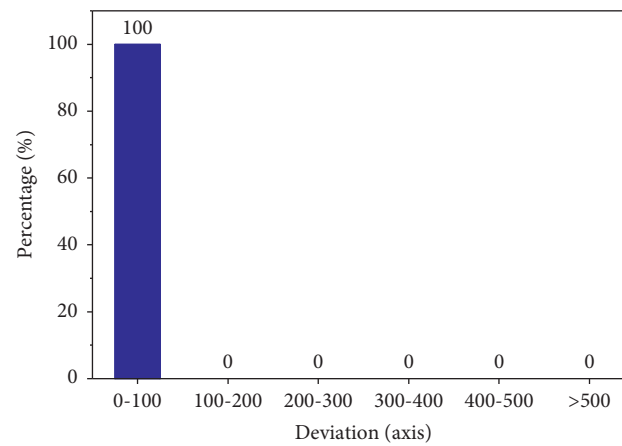
Figure 14: MAE histogram of straight trajectory prediction.
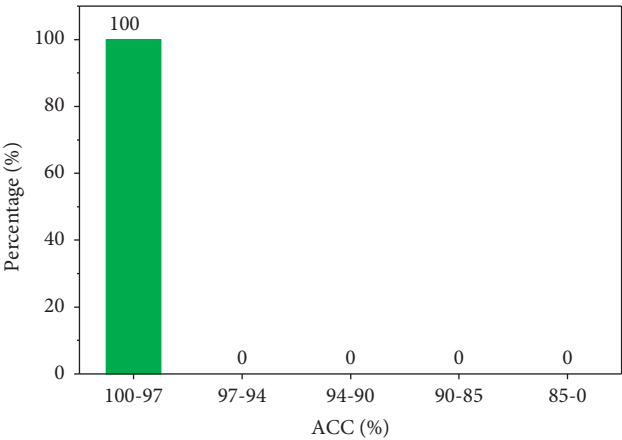
FIGURE 15: Training accuracy distribution for left turn.
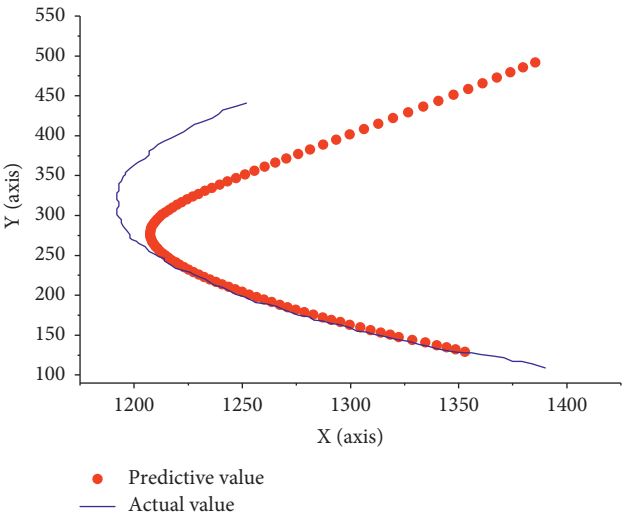

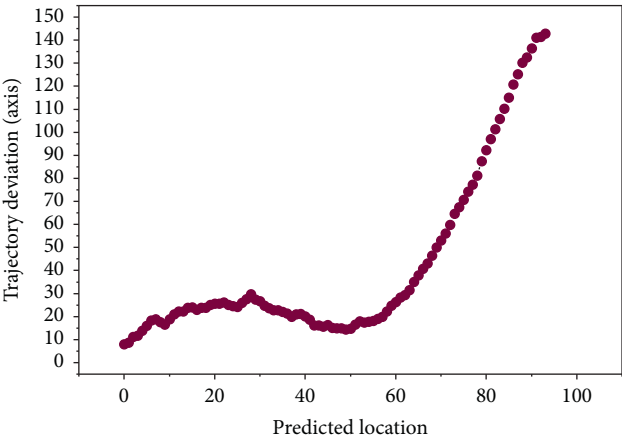FIGURE 16: Prediction result of the left-turn trajectory.

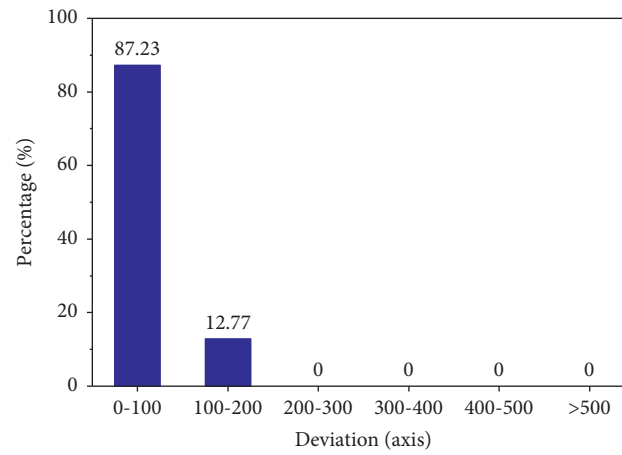FIGURE 17: MAE scatter plot for left-turn trajectory prediction.

FIGURE 18: MAE histogram of left-turn trajectory prediction.
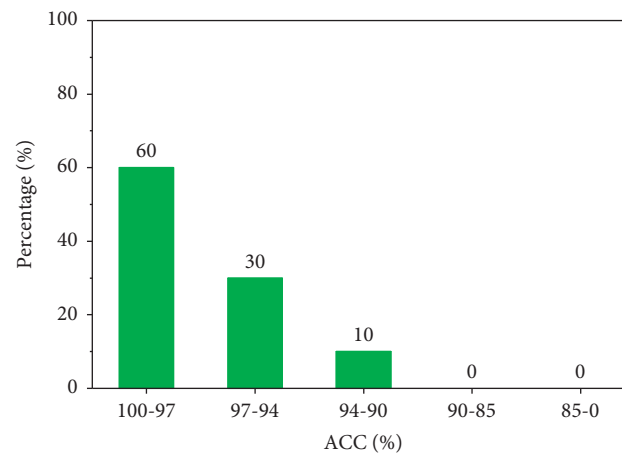

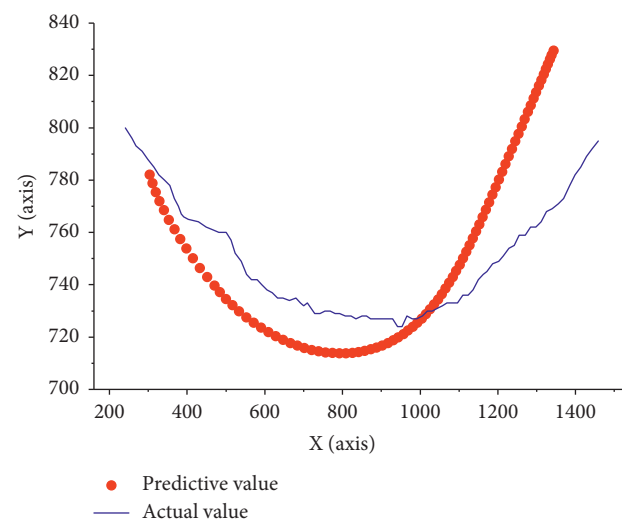FIGURE 19: Training accuracy distribution for right turn.

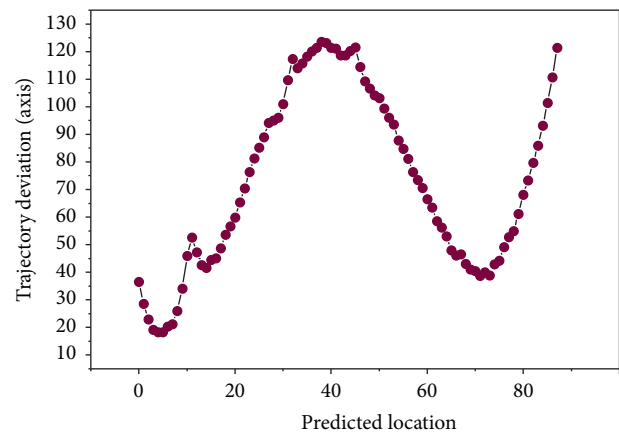FIGURE 20: Prediction result of the right-turn trajectory.

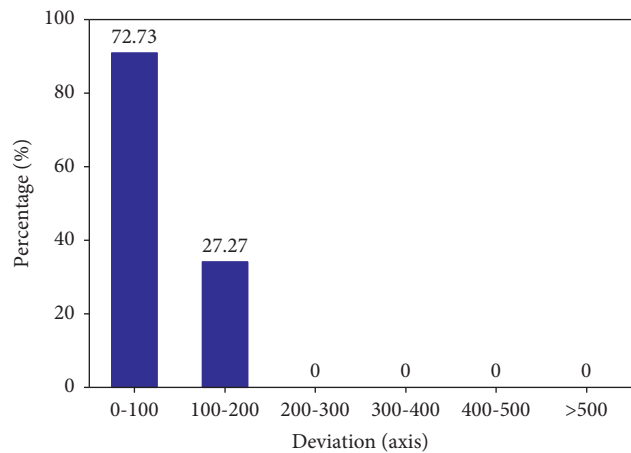FIGURE 21: MAE scatter plot for right-turn trajectory prediction.

FIGURE 22: MAE histogram of right-turn trajectory prediction.

FIGURE 23: Real-time vehicle trajectory prediction effect.

4. Conclusions

In this paper, a real-time vehicle trajectory prediction method based on the vehicle heading angle change trend recognition algorithm and the multilayer LSTM model is constructed. This method first extracts the vehicle trajectory data of the intersection through video detection and then trains three multilayer LSTM models for going straight, left turn, and right turn according to the direction categories. Then, we use the vehicle heading angle change trend recognition algorithm to recognize the turning intention of the target vehicle. Finally, we use the LSTM model corresponding to the turning category to predict the trajectory position in real time. The experimental results show that compared with other algorithms, the vehicle heading angle change trend recognition algorithm has better prediction accuracy and stability. The four-layer LSTM model is effective in predicting vehicle trajectories at unsignalized intersections. Compared with the constant turn rate and acceleration (CTRA) vehicle model, it has better prediction performance.

Our further work is to explore the improved algorithm of LSTM or GRU and at the same time increase the prediction speed to achieve further improvement in real-time performance. Then, by further studying the conflict discrimination algorithm, real-time conflict warning of vehicles at unsignalized intersections will be realized.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] K. Kwangjin Yang and S. Sukkarieh, "An analytical continuous-curvature path-smoothing algorithm," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 561–568, 2010.

[2] E. A. Pool, J. F. Kooij, and D. M. Gavrila, "Using road topology to improve cyclist path prediction," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pp. 289–296, Los Angeles, CA, USA, June 2017.

[3] F. Xie, J. T. Lou, and K. Zhao, "A research on vehicle trajectory prediction method based on behavior recognition and curvature constraint," *Automotive Engineering*, vol. 41, no. 9, pp. 1036–1042, 2019.

[4] A. Barth and U. Franke, "Where will the oncoming vehicle be the next second?" in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 1068–1073, Eindhoven, Netherlands, June 2008.

[5] A. Kawasaki and T. Tasaki, "Trajectory prediction of turning vehicles based on intersection geometry and observed velocities," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pp. 511–516, Changshu, China, June 2018.

[6] S. Choi, H. Yeo, and J. Kim, "Network-wide vehicle trajectory prediction in urban traffic networks using deep learning," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2672, no. 45, pp. 173–184, 2018.

[7] G. Chen, L. Hu, Q. Zhang, Z. Ren, X. Gao, and J. Cheng, "ST-LSTM:Spatio-Temporal graph based long short-term memory network for vehicle trajectory prediction," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 608–612, Abu Dhabi, UAE, October 2020.

[8] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. F. Fei, and S. Savarese, "Social lstm: human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 961–971, Las Vegas, NV, USA, june 2016.

[9] X. W. Ji, C. Fei, and X. K. He, "Intention recognition and trajectory prediction for vehicles using LSTM network," *China Journal of Highway and Transport*, vol. 32, no. 6, pp. 34–43, 2019.

[10] C. Luo, L. Sun, D. Dabiri, and A. Yuille, "Probabilistic multimodal trajectory prediction with lane attention for autonomous vehicles," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2370–2376, Las Vegas, NV, USA, October 2020.

[11] L. Fang, Q. Jiang, J. Shi, and B. Zhou, "Tpnet: trajectory proposal network for motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6797–6806, Seattle, WA, USA, June 2020.

[12] Y. Yao, E. Atkins, M. J. Roberson, and R. Vasudevan, "Bitrap: Bi-directional pedestrian trajectory prediction with multimodal goal estimation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1463–1470, 2021.

[13] R. Y. Feng, Z. B. Li, and Q. F. Wu, "Association of vehicle object detection and the time-space trajectory matching from aerial videos," *Journal of Transport Information and Safety*, vol. 39, no. 2, pp. 61–69+77, 2021.

[14] X. Chen, Z. Li, Y. Yang, L. Qi, and R. Ke, "High-resolution vehicle trajectory extraction and denoising from aerial videos," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 3190–3202, 2020.

[15] M. S. Shirazi and B. T. Morris, "Trajectory prediction of vehicles turning at intersections using deep neural networks," *Machine Vision and Applications*, vol. 30, no. 6, pp. 1097–1109, 2019.

[16] H. Jiang, L. Chang, Q. Li, and D. Chen, "Trajectory prediction of vehicles based on deep learning," in *Proceedings of the 4th International Conference on Intelligent Transportation Engineering (ICITE)*, pp. 190–195, Singapore, September 2019.

[17] X. Chen, H. Chen, Y. Yang et al., "Traffic flow prediction by an ensemble framework with data denoising and deep learning model," *Physica A: Statistical Mechanics and Its Applications*, vol. 565, Article ID 125574, 2021.

[18] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi, "Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network," in *Proceedings of the IEEE 20th International Conference on*

Intelligent Transportation Systems (ITSC), pp. 399–404, Yokohama, Japan, October 2017.

[19] F. Mirus, T. C. Stewart, and J. Conradt, "The importance of balanced data sets: analyzing a vehicle trajectory prediction model based on neural networks and distributed representations," in *Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Glasgow, UK, July 2020.

[20] H. Saleem, F. Riaz, L. Mostarda, M. A. Niazi, A. Rafiq, and S. Saeed, "Steering angle prediction techniques for autonomous ground vehicles: a review," *IEEE Access*, vol. 9, pp. 78567–78585, 2021.

[21] T. H. Wu, T. W. Wang, and Y. Q. Liu, "Real-time vehicle and distance detection based on improved yolo v5 network," in *Proceedings of the 3rd World Symposium on Artificial Intelligence (WSAI)*, pp. 24–28, Guangzhou, China, June 2021.