This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

Vehicle Trajectory Prediction with Interaction Regions and Spatial-Temporal Attention

Dengyang Cheng[1], Xiang Gu[1], Cong Qian[1], Chaonan Du[1], and Jin Wang[1],(Member, IEEE)
[1]School of Infomation Science and Technology, Nantong University, Nantong, Jiangsu, China

Corresponding author: Xiang Gu (e-mail: gu.x@ntu.edu.cn).

"This work is supported by the National Natural Science Foundation of China(62002179) and the Science Found-ation of Nantong of Jiangsu Grants (JC22022038)."

ABSTRACT Vehicle trajectory prediction is a key technology in autonomous driving, one of the aims of which is to ensure the safety of vehicle travelling and lane changing. Especially in complex traffic scenarios, it is critical to accurately predict the vehicle trajectory so that the automated driving system can make appropriate response actions to help the vehicle driver avoid accidents to the greatest extent possible. Some studies on vehicle trajectory prediction have used recurrent networks (LSTMs) to extract temporal correlations and additional convolutional neural networks (CNNs) to capture spatial correlations. This hybrid model can be added more operations, so we propose a new CNN-LSTM hybrid model containing multi-module, interaction regions and spatial-temporal attention mechanisms (IA-CSTM). In this paper, the traffic scene is divided into multiple interaction regions, and vehicles in different regions have different impacts on the predicted vehicles. The model has multiple modules that deal with vehicle information in different interaction regions. The model will cross-fuse vehicle information in important interaction regions to enhance the model's perception of vehicle information. Multiple modules are linked through a spatial-temporal attention mechanism and capture the relevance of the vehicle in the interaction region to predicted vehicle.

INDEX TERMS Vehicle trajectory prediction, interaction regions, multi-module, spatial-temporal attention mechanism, IA-CSTM.

I. INTRODUCTION

THE emergence of autonomous driving technology holds the promise of enhancing driver and passenger safety while curbing economic losses attributed to human error-induced car crashes. Nevertheless, accomplishing a fully human-replacing autonomous driving system poses challenges due to the intricate and dynamic road environment, encompassing interactions among diverse vehicles. For autonomous driving technology to navigate safely and efficiently, it necessitates the capability to detect and identify other objects, and promptly predict and respond to their actions, akin to human behavior. Consequently, accurately predicting the trajectories of other road objects becomes pivotal in autonomous driving technology.

Trajectory prediction is a challenging problem for the following reasons. Vehicles are interdependent and the behavior of one vehicle has an impact on other vehicles [1]. For instance, when a vehicle ahead slows down, human drivers tend to adjust their speed accordingly. To achieve accurate trajectory prediction, the model must consider not only the predicted vehicle's trajectory but also the trajectories of surrounding vehicles and speculate on their potential future interactions. The accumulation of errors further compounds the issue. Typically, a trajectory prediction model relies on the vehicle's current position and historical trajectory to anticipate its next position, leading to gradual error accumulation and deviation from the actual situation. Additionally, the highly nonlinear nature of vehicle trajectories over time, influenced by drivers' decisions, poses a significant challenge for traditional physical and machine learning models.

Previous research has suggested predicting objects' future positions by identifying their maneuvers (e.g., lane changes, braking, or continuing driving) [2], [3]. However, when these methods inaccurately inferred the maneuver type, they failed to achieve precise object location prediction. This kind of misjudgment often occurs in scenarios where predictions solely rely on sensors like GPS while disregarding visual cues, such as steering signals. Subsequently, Bhattacharyya et al. [4] attempted to deduce the bounding boxes of objects in RGB camera frames by predicting future vehicle odometer

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

TABLE 1: Summary of related works.

Reference	Analysis Type	Application	Models	Techniques	Datasets
Lin et al. 1995 [5]	Simulated data	Vehicle Trajectory Prediction	Kalman Filter Dynamic Model	Adaptive to Kalman filter gain	Simulated Data
Lefèvre et al. 2011 [8]	Real data	Vehicle Trajectory Prediction	Bayesian Network	A Bayesian Network with uncertain evidence	Real traffic at two four-way-stop intersections
Zhang et al. 2020 [10]	Real data	Vehicle Trajectory Prediction	Gaussian Mixture and Hidden Markov Model	Calculate front running space and comfort loss under each scenario	NGSIM
Zhang et al. 2018 [12]	Simulated data	Flight Trajectory Prediction	ALO-LSTM Model	Use ALO algorithm to optimize the initial weights of LSTM network	Simulated Data
Fujii et al. 2021 [13]	Real data	Human Trajectory Prediction	RNN Model	Use two-block RNN model	ETH, UCY
Yang et al. 2020 [15]	Real data	Vehicle Trajectory Prediction	ConvNets and LSTMs	Use an attention mechanism and a social map	BLVD
Li et al. 2019 [16]	Real data	Vehicle Trajectory Prediction	GRIP model	Use several graph convolutional blocks and LSTMs	NGSIM
Alahi et al. 2016 [19]	Real data	Human Trajectory Prediction	Social-LSTM model	Use the social-pooling strategy	ETH, UCY
Deo et al. 2018 [20]	Real data	Vehicle Trajectory Prediction	CS-LSTM model	Use the convolutional social pooling	NGSIM
Miguel et al. 2022 [23]	Real data	Vehicle Trajectory Prediction	Recurrent VAE Network	Recurrent layers and Variational Auto Encoder model architecture	HighD, NGSIM
Seff et al. 2023 [24]	Real data	Vehicle Trajectory Prediction	MotionLM model	Cast multi-agent motion prediction as a language modeling	WOMD
(our model)	Real data	Vehicle Trajectory Prediction	IA-CSTM model	Use multiple interaction regions and spatial-temporal attention mechanism	NGSIM

sequences. Nevertheless, for the self-driving car to respond accurately to these predicted locations, the projected bounding boxes in the RGB frames still need to be mapped onto the self-driving car's coordinate system.

During vehicle travelling, drivers encounter various types of traffic agents such as cars, pedestrians, bicycles and buses. These traffic agents have different shapes, dynamic characteristics and movement patterns. To ensure the safe operation of self-driving vehicles, their perception and navigation systems should be able to analyse the motion patterns of the surrounding traffic agents and predict their future locations in order to make more accurate driving decisions. To simulate the interaction between vehicles, some academics [19] represent human trajectory location information as a social tensor map structure and use a pooling layer to capture inter-vehicle correlations. Some academics [20] do convolutional operations on the tensor graph to improve the model's perception of spatial correlations between vehicles. Some academics [16] construct the targets within the traffic scene as an undirected graph and do convolution operations on the undirected graph to extract spatial temporal correlations. There are also academics [24] used model to produce joint distributions over interactive agent futures in a single autoregressive decoding process. These methods capture the spatial-temporal correlations between target vehicles during the encoding or decoding phase. In this paper, multiple interaction regions and attention are adopted to capture spatial temporal correlations between vehicles. Multiple interaction regions and modules can help the model capture inter-vehicle spatial-temporal correlations.

This paper introduces a multi-module CNN-LSTM hybrid model with Interaction and Attention (IA-CSTM). The IA-CSTM model comprises two distinct interaction regions, namely the vision interaction region and the neighbourhood interaction region, where vehicles within these regions exert varying influences on the ego vehicle. The vision module within the IA-CSTM model is founded on the Vision interaction region. To enrich vehicle information, we propose the employment of a cross fusion block, facilitating feature cross fusion and tensor map construction. On the other hand, the Neighbourhood module is devised based on the neighbourhood interaction region, enabling the extraction of interaction information between the ego vehicle and its surrounding neighbors. Furthermore, the Ego module is developed specifically for the predicted vehicle. In this context, we introduce spatial-temporal attention mechanisms, each catering to distinct aspects: one for capturing temporal dependencies in vehicle trajectories and the other for grasping spatial interactions between the home vehicle and its adjacent counterparts. In traffic scenarios, interactions between vehicles involve a variety of driving behaviours such as overtaking, lane changing, approaching and slowing down. These behaviours reflect the dynamics between the vehicles and are mapped to the vehicle trajectories. By utilizing the spatial-temporal attention mechanism, our method effectively captures diverse types of traffic interactions within the vision and neighbourhood interaction regions, culminating in improved predictive accuracy of the model.

In this paper, the IA-CSTM model is validated on the public Next Generation Simulation (NGSIM) U.S. Highway 101 (US-101) and Interstate 80 (I-80) datasets. The experimental findings distinctly demonstrate the superiority of our method over other baselines, showcasing remarkable performance improvement.

The main contribution of this work are summarized as

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

follows:

1) A model with interaction regions and multiple modules is proposed, which effectively captures the correlations between vehicles. Each module is designed to address specific interaction regions.
2) A spatial-temporal attention method is proposed to capture the temporal correlation of vehicle trajectories and the spatial correlation between vehicles, and to help the model link multiple modules.

II. RELATED WORK

A. PHYSICS AND MACHINE LEARNING BASED TRAJECTORY PREDICTION METHODS

There exist numerous conventional trajectory prediction techniques comprising both physics-based and machine learning methods. The physical models consist of kinetic models [5], Kalman filter models [6] and similar approaches. Lytrivis et al. [7] presented a cooperative path prediction algorithm that takes into account data coming from other vehicles via wireless communications. There are also traditional machine learning trajectory prediction methods, such as Bayesian networks [8] and Hidden Markov Models [9], [10], are utilized.

In some complex scenarios, traditional physics-based approaches and machine learning models may be limited because they fail to effectively capture the diversity and interaction in vehicular traffic scenarios. Single-trajectory CTRA models may assume that vehicles have similar motion patterns, ignoring the differences between different types of vehicles. Dynamics models may focus only on the previous trajectories of vehicles and fail to adequately consider potential coordination and competition between vehicles. Hidden Markov predictions of future trajectories rely only on the current state, but vehicle trajectories may receive influences over longer historical time horizons.

B. DEEP-LEARNING BASED TRAJECTORY PREDICTION METHODS

Deep learning is also divided into two categories, one for sequence prediction models and one for spatial interaction models. Among the sequence prediction models, GRU [11] and LSTM [12] have been successful in the task of predicting vehicle trajectory sequences. Both LSTM and GRU are able to deal with long term dependencies in the trajectory sequences, where the vehicle behaviour may be influenced by factors over a long history time. RNNs are also widely used in modelling the historical motion trajectories of vehicles [13]. RNN has good effect on the processing of sequence data, which is very important for vehicle trajectory prediction, the trajectory data has strong sequence. RNN has the ability to capture the temporal information in the historical trajectory by taking into account the historical position and speed of the vehicle. Bhujel et al. [14] proposed RNNs with an encoder-decoder architecture, and encoder decoder is also a common architecture for RNNs. RNN is mainly designed to deal with time series data and their prediction of vehicle trajectories in simple traffic scenarios gives better results. When there are

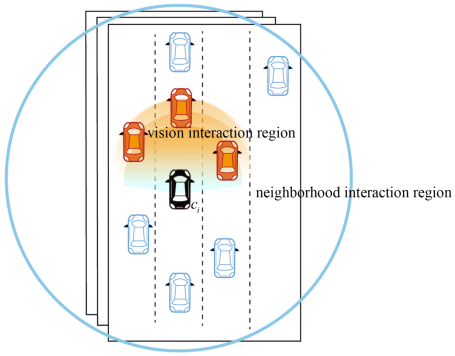


FIGURE 1: **Vision and Neighborhood interaction regions:** We highlight the various interactions of the ego vehicles (the black vehicle). The vision interaction region is the orange semi-circular region, and the vehicles in the region are labelled in orange. The neighbourhood interaction region is the blue circular region and the vehicles in the region are labelled blue. The model modelled these interaction regions with several modules.

complex traffic scenarios, the vehicle trajectory data will also be spatial, which is not an area of expertise for RNNs. To capture spatial features, RNN is usually used in combination with other models. Examples include CNNs, attention mechanisms [15] and graph convolutional networks [16].

Systems based on target detection and graph convolutional networks based on attention mechanisms are beginning to be used in traffic scenarios and are also performing well in vehicle trajectory prediction. Güney et al. [17], [18] proposed and implemented a real-time advanced driver assistance systems (ADAS system) with low power consumption and high-speed mobile GPU platforms. This system can accurately detect vehicles, traffic signs and pedestrians to obtain their node information. The results of these target detections can be constructed as nodes of an undirected graph, where each node represents a target. Miguel et al. [23] proposed method uses the encoder part of the Variational Auto Encoder model (VAE) to represent the vehicle's surroundings agents behavior in time, taking advantage on the fact that VAE encodes similar situations or states close in the latent space and the generative properties of the VAE decoder, that is used to generate naturalistic driving trajectories. Seff et al. [24] represented continuous trajectories as sequences of discrete motion markers and cast multi-intelligent motion prediction as a language modelling task. Temporal causality can be achieved through sequential factorisation of this approach.

Alahi et al. [19] connected neighbouring LSTMs through a social-pooling strategy, which enables spatially neighbouring lstm to share information with each other. This allows for better vehicle interaction in traffic scenarios. Deo et al. [20] represented neighbouring objects in terms of the social tensor and proposed a convolutional social pooling method. Vehicle interactions in a traffic scenario are modelled with a convolutional social pool, and the hidden information of the tensor map is extracted with convolutional operations. This method improves on the social pooling method proposed by

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

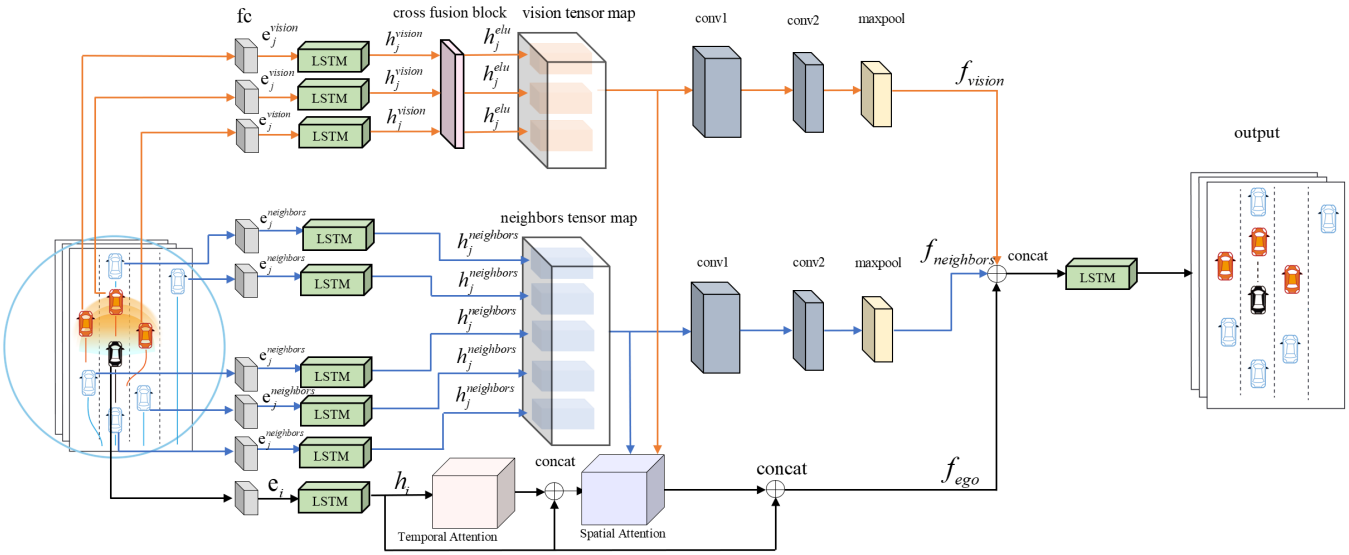


FIGURE 2: **IA-CSTM Model Architecture:** The ego vehicle is represented in black. The orange semi-circular region in front of it is its vision interaction region and The blue circular region around it is its neighborhood interaction region. The vision module processes vehicle information within the vision interaction region, while the neighborhood module handles information within the neighborhood interaction region. We generate input embeddings for all vehicles based on trajectory information. These embeddings are passed through LSTMs and eventually used to construct the vision tensor map, the neighbors tensor map and the ego tensor. The vision and neighbors tensor map are passed through separate convolutional network, and do spatial-temporal attention manipulation with the ego tensor in the ego module. Finally, the outputs of the three modules are concatenated and fed into an LSTM to generate a trajectory prediction for the ego vehicle.

Alahi. Compared to the pooling method, the novel attention mechanism can calculate the importance of different neighbourhoods to the study object and can capture more hidden information.

In this paper, we improve Deo method and propose a multi-module network that includes multiple modules and spatial-temporal attention mechanism.

III. PROBLEM FORMULATION

Before describing the scheme of this paper, this paper defines the trajectory prediction problem as the problem of predicting the future location of an object based on its trajectory history. As shown in Fig 1, for all vehicles in the blue circular region there is set $A = \{c_i\}_{i=1,2...n}$, where n indicates the number of vehicles in the region. Each vehicle's historical trajectory has a total of t time steps. The input X to the model in this paper is the trajectory coordinates of vehicles in all time steps, denoted as:

$$X = \left[p^1, p^2, p^3, ..., p^{t_h}\right] \tag{1}$$

where

$$p^t = \left[\left(x^t_{c_1}, y^t_{c_1}\right), \left(x^t_{c_2}, y^t_{c_2}\right), ..., \left(x^t_{c_n}, y^t_{c_n}\right)\right]_{t=1,2...t_h} \tag{2}$$

represents the coordinates (x, y) of all vehicles in the observed scene at time step t; c_i is one of the set A. The outputs Y of the model are the predicted future trajectory of observed vehicles from time step t_h+1 to t_h+t_f, and t_f is the predicted horizon:

$$X = \left[q^{t_h+1}, q^{t_h+2}, q^{t_h+3}, ..., q^{t_h+t_f}\right] \tag{3}$$

where

$$q^t = \left[\left(x^t_{c_1}, y^t_{c_1}\right), \left(x^t_{c_2}, y^t_{c_2}\right), ..., \left(x^t_{c_n}, y^t_{c_n}\right)\right]_{t=t_h+1,t_h+2...t_h+t_f} \tag{4}$$

IV. PROPOSED SCHEME

In the context of multi-objective traffic, the goal of this paper is to predict vehicle trajectories, i.e., future spatial coordinate sequences of vehicles. To achieve time series prediction, a model that captures temporal correlations in the data, such as LSTM, is required. However, LSTM is unable to learn dependencies or associations between heterogeneous vehicles. In this paper, several modules are constructed to incorporate convolutional neural networks and spatial-temporal attention mechanisms. Convolutional neural networks can learn local dependencies from features. And the spatial-temporal attention mechanism allows the model to amplify the temporal correlation of the vehicle's own historical trajectory and extract the spatial correlation between the vehicle and the vehicles when processing heterogeneous vehicle data. With different modules, interaction region information and spatial-temporal attention mechanisms, the model proposed in this paper is able to learn the interaction information between different vehicles in space and time and better capture the temporal and spatial correlations in the data. This allows us to predict vehicle trajectories more accurately. We visualise the model structure in Fig 2. This hybrid model approach promises to improve the performance of trajectory prediction in heterogeneous traffic data. It is able to take into account both local features and relationships between vehicles on

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

different interaction regions, thus better capturing complex relationships in the data.

A. INTERACTION REGIONS

In order to better predict the vehicle trajectory, this paper proposes to add different training weights to different vehicles in the traffic scene. In this paper, we set two different interaction regions for the ego vehicle (the black one), because during the driving process, the driver will focus most of his attention on the observable vehicles in front of the vehicle, and if the vehicle in front of the vehicle shows slowing down, changing direction and braking sharply, then the vehicle will be largely affected by it. We define the semi-circular region in front of the vehicle as the vision interaction region (as shown in the orange region in Fig 1). However, the surrounding vehicles also have different effects on the ego vehicle, and we define the circular region around the ego vehicle as the neighbourhood interaction region (shown as the blue circular region in Fig 1). Vehicles are classified into three categories, the first being the ego vehicle (the black one), the second being the driver-observable vehicle in the orange semi-circular region, and the third being the neighbouring vehicles in the blue circular region. Where vehicles in the neighbourhood interaction region do not include vehicles in the vision interaction region.

Based on the vision interaction region and neighborhood interaction region set in this paper, the number of vehicles in the two interaction regions is represented as the set $V = \{c_i\}_{i=1.2...m}$ and $N = \{c_i\}_{i=1.2...n}$. Then all the vehicles within interaction regions $c_i \in V \cup N$, and input their historical trajectories into the model.

B. VISION MODULE

The model proposed in this paper consists of several modules, and the inputs X_j^{vision} to the Vision Module (represented by the orange line in Fig 2) are the historical trajectories of all the vehicles in the vision interaction region. The data first goes through the fully connected layer to get e_j^{vision}.

$$e_j^{vision} = FC\left(W_{FC}X_j^{vision} + b_{FC}\right) \tag{5}$$

where W_{FC} and b_{FC} are conventional symbols denoting the weight matrix and bias vector respectively, of the fully connected layer. Then it goes through a single LSTM and outputs the hidden state $h_j^{vision}(t)$, t is the time step.

$$h_j^{vision}(t) = LSTM\left(h_j^{vision}(t-1), e_j, W_{LSTM}, b_{LSTM}\right) \tag{6}$$

where $h_j^{vision}(t-1)$ refers to the corresponding vehicle's hidden state vector from the previous time step $t-1$. W_{LSTM} and b_{LSTM} are conventional symbols denoting the weight matrix and bias vector respectively, of the layer LSTM.

Vehicles in the vision interaction region have more potential impact on the ego vehicle travelling, in order to be able to enhance the correlation between vehicles in this region and the ego vehicle. In this paper, we propose a feature cross-fusion block to cross-fusion the hidden states of vehicles in the vision interaction region with features, thus allowing the model to pay more attention to the information about vehicles in the region. As shown in Fig 3, in this block, the first vehicle's hidden state h_1^{vision} is concatenated with the second vehicle's hidden state h_2^{vision} , and the second vehicle's hidden state h_2^{vision} is concatenated with the third vehicle's hidden state h_3^{vision}. And so on, until the $(n-1)$-th vehicle's hidden state h_{n-1}^{vision} is concatenated with the n-th vehicle's hidden state h_n^{vision}.

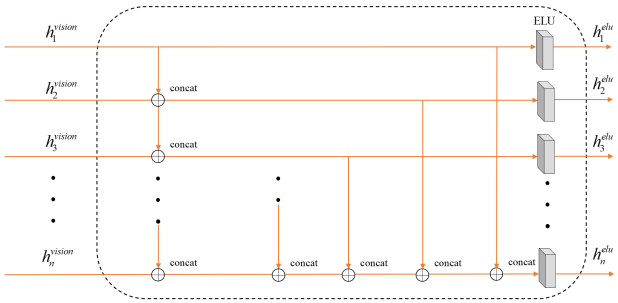


FIGURE 3: **Cross Fusion Block:** In order to enhance the correlation between the observed vehicles and the ego vehicle, we do a cross fusion operation on the hidden states of the vehicles in the vision interaction region.

$$h_2^{vision} = concat\left(h_1^{vision}, h_2^{vision}\right) ... h_n^{vision} = concat\left(h_{n-1}^{vision}, h_n^{vision}\right) \tag{7}$$

Then, all the hidden states of the vehicles after completing the operation are simultaneously concatenated with the hidden state of the n-th vehicle.

$$h_n^{vision} = concat\left(h_1^{vision}, h_2^{vision}, ..., h_{n-1}^{vision}\right) \tag{8}$$

All the hidden states h_j^{vision} of the interacting vehicles are passed through the fully connected layer of the Exponential Linear Units (ELUs) non-linearity [21] to obtain h_j^{elu}. All h_j^{elu} are then pooled together in a "vision tensor map", $Vs_j = \left\{h_j^{elu} | c_j \in V\right\}$. Finally, the vision tensor map is convolved with two different layers of convolution kernels and one maxpool layer to get the output f_{vision}.

C. NEIGHBORHOOD MODULE

The inputs to the neighborhood module (indicated by the blue line in Fig 2) are the historical trajectory coordinates of all vehicles in the neighborhood interaction region (excluding vehicles in the vision interaction region). Similar to the vision module, the input data first goes through the fully connected layer to get $e_j^{neighbors}$. Then, after a single LSTM to output the vehicle hidden states $h_j^{neighbors}$, all $h_j^{neighbors}$ are pooled together in a "neighbors tensor map", $Nb_j = \left\{h_j^{neighbors} | c_j \in N\right\}$. Finally, the neighbors tensor map is convolved with two different layers of convolution kernels and one maxpool layer to get the output $f_{neighbors}$.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

D. EGO MODULE

The LSTM model can capture the temporal correlation between the vehicles themselves and their historical trajectories, but cannot capture the spatial correlation between the vehicles and each other. In order to further extract the temporal correlation of vehicles themselves and the spatial dependence between vehicles, this paper proposes to combine spatial-temporal attention and LSTM to construct the ego module (indicated by the black line in Fig 2). The ego vehicle data first goes through the fully connected layer and single LSTM to get the hidden state h_i of the vehicle history data. Then, temporal attention operation is done on h_i to extract more temporal correlation of the ego vehicle trajectory. Finally, the spatial attention operation is done on h_i, h_j^{vision} and $h_j^{neighbors}$ (for ease of illustration, h_j^{vision} and $h_j^{neighbors}$ in this section are referred to as h_j), and extract spatial correlations between vehicles.

Temporal attention: As shown in Fig 4, in the temporal attention module, all the information of the past time step of the ego vehicle will do message passing with the information of the present time step. The module firstly computes the query matrices $\{Q_i^{ego}\}$, key matrices $\{K_i^{ego}\}$ and the value matrices $\{V_i^{ego}\}$. For the i-th ego vehicle, we calculate

$$Q_t^{ego}(t) = W_{Q^{ego}} h_i^t, K_t^{ego}(t) = W_{K^{ego}} h_i^t, V_t^{ego}(t) = W_{V^{ego}} h_i^t \tag{9}$$

where $W_{Q^{ego}}$, $W_{K^{ego}}$ and $W_{V^{ego}}$ are the corresponding query, key and value matrix of the i-th ego vehicle. For the trajectory of the i-th ego vehicle, as shown in Fig 4(a), we define the message passing from time step s to t as:

$$m^{s \to t} = (Q_i^{ego}(t))^T K_i^{ego}(s) \tag{10}$$

Then, we compute the temporal attention masks for the i-th ego vehicle at time step t as follows:

$$TAttention(t) = \frac{Softmax([m^{s \to t}])}{\sqrt{d_k}} [V_i^{ego}(s)]_{s=1\ldots t-1} \tag{11}$$

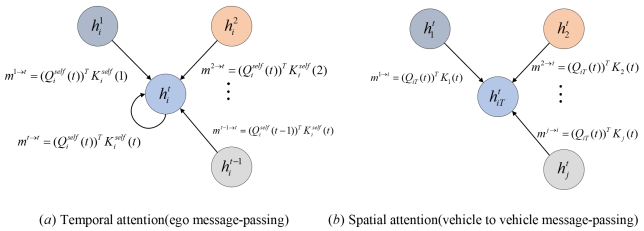
where $[m^{s \to t}]_{s=1\ldots t-1}$ shows that the current time step information can only interact with the time step information preceding it. Similarly, we can get the multi-head attention (k heads) of the i-th ego vehicle at the time step t.

$$T_i(t) = f_o([head_1, \ldots, head_k]) | head_j = TAttention_j(t), j \in [1, k] \tag{12}$$

where f_o is a fully connected layer that merges information from k headers. After calculating the multiple attention T_i for each time step $t \in [1, t_h]$ of the i-th vehicle, and concatenate T_i with the original input h_i to get h_{iT}.

$$h_{it} = concat(T_i, h_i) \tag{13}$$

Spatial attention: In order to capture the spatial correlation between the ego vehicle and the surrounding vehicles, the h_{iT} obtained from temporal attention block and the hidden state h_j of the vehicles in the interaction region do spatial attention operation. The module firstly calculates the ego vehicle's query vector $q_{iT}^t = W_{q_{iT}}(t) h_{iT}^t$, surrounding vehicles' key vector $k_j^t = W_{k_j}(t) h_j^t$ and value vector $v_j^t = W_{v_j}(t) h_j^t$. We compute



(*a*) Temporal attention(ego message-passing) (*b*) Spatial attention(vehicle to vehicle message-passing)

FIGURE 4: **(a) Temporal Attention:** the hidden state of the ego vehicle at time step t, i.e., h_i^t, can only access its previous steps' hidden states $h_i^1, h_i^2, \ldots, h_i^{t-1}$;**(b) Spatial Attention:** each step allows the ego vehicle to message with all vehicles in the interaction region.

$$Q_{iT}(t) = \{q_{iT}^t\}, K_j(t) = \{k_j^t\}, V_j(t) = v_j^t | j \in V \cup N \tag{14}$$

and the information transfer from the surrounding vehicle j to the ego vehicle as follows:

$$m^{j \to i} = (Q_{iT}(t))^T K_j(t) \tag{15}$$

then the spatial attention masks at time step t can be calculated as:

$$SAttention(t) = \frac{Softmax([m^{j \to i}])}{\sqrt{d_k}} [V_j(t)]_{j \in V \cup N} \tag{16}$$

We can obtain the muti-head attention (k heads) of the ego vehicle for time step t:

$$S_i(t) = f_o([head_1, \ldots, head_k]) | head_j = SAttention_j(t), j \in [1, k] \tag{17}$$

where f_0 is a fully connected layer that merges the k heads' information. After calculating the multi-head attention S_i of the ego vehicle at each time step t. S_i is concatenated with the hidden state h_i of the ego vehicle that has previously gone through the single LSTM to obtain h_{iS}.

$$h_{iS} = concat(S_i, h_i) \tag{18}$$

h_{iS} is then passed through the fully connected layer to get f_{ego}. Finally, the outputs of the three modules are concatenated to obtain z.

$$z = concat(f_{vision}, f_{neighbors}, f_{ego}) \tag{19}$$

We input z into a LSTM decoder to predict future the ego vehicle future trajectory.

E. IMPLEMENTATION DETAILS

In this paper, two interaction regions are set for the ego vehicle, one is vision interaction region and the other is neighborhood interaction region. The vision interaction region is a semi-circular region (orange semi-circular region in Fig 1) with radius $R1$ in front of the ego vehicle centred on the vehicle. The neighbourhood interaction region is a circular region centred on the ego vehicle with a radius of $R2$ (the blue circular region in Fig 1).The final values of $R1$ and

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

$R2$ will determine the optimal value based on the ablation experiments in the next section.

We train our model as a regression task. The overall loss can be calculated as:

$$Loss = \frac{1}{t_h} \sum_{t=t_h+1}^{t_h+t_f} \left\| Y_t^{pred} - Y_t^{true} \right\|^2 \tag{20}$$

where t_f is predicted horizon, Y_t^{pred} and Y_t^{true} are predicted positions and ground truth at time step t, respectively.

In this paper, we use single-layer LSTMs as encoders and decoders with hidden state dimensions of 64 and 128, respectively. The size of the convolution kernel for the first convolution layer is 3×3. The size of the convolution kernel for the second convolution layer is 3×1. The size of the convolution kernel for the maxpool layer is 2×1. We using the Adaptive Moment Estimation (Adam) optimizer [22] with a batch size of 64 and learning rate of 0.001.

V. EXPERIMENT

A. EXPERIMENT ENVIRONMENT

In this paper, in order to make a fair comparison with other methods, the models in this paper are trained and evaluated on publicly available datasets. The experiments in this paper are conducted on a server running ubuntu 20.04 with the arithmetic model V100-SXM2, 32GB of GPU and 12-core CPU.

Dataset: The proposed model was trained and evaluated using the public NGSIM US-101 and I-80 datasets. Both datasets were captured at 10 Hz over 45 min and split into three periods of 15 min. These periods represent mild, moderate, and congested traffic conditions. These two datasets consist of vehicles' trajectories on real freeway traffic. Each vehicle's trajectory was divided into segments of 8 s, where the first 3 s are used as observed track history and the remaining 5 s are the prediction horizon. Following Deo et al. [20], the trajectory data were down-sampled for 10 Hz to 5 Hz, i.e., five frames per second. The two datasets above are merged into one dataset, which is randomly shuffled and divided into the training set, validation set, and test set at a ratio of 7:1:2. The following experimental evaluations are conducted on the test set.

Evaluation Metrics: In this paper, we use the same evaluation metrics as other methods [20] and report our results in terms of the root mean square error (RMSE) of the predicted future trajectory at each time step within the 5 seconds prediction horizon. At the time step t, the RMSE can be calculated as follows:

$$RMSE_t = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (Y_t^{pred}[i] - Y_t^{true}[i])^2} \tag{21}$$

where m is the number of vehicles in the test datasets, Y_t^{pred} and Y_t^{true} are predicted positions and ground turth at time step t, respectively.

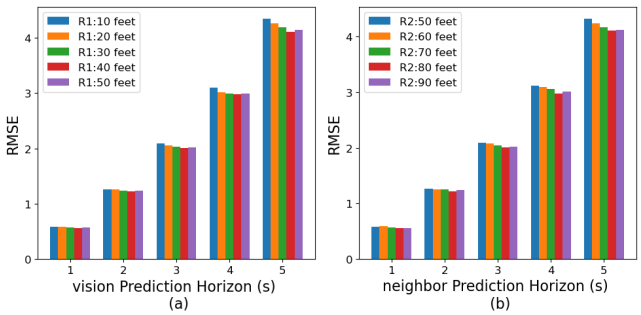


FIGURE 5: **(a):** fixing the radius $R2$ of the neighborhood interaction region and comparing different vision interaction region radius $R1$;**(b):** fixing the radius $R1$ of the vision interaction region and comparing different neighborhood interaction region radius $R2$.

B. ABLATION EXPERIMENT

In this paper, two interaction regions are constructed for the ego vehicle. The model includes several modules. To verify whether interaction regions and modules give a boost to the model, we performed ablation experiments for these interaction and modules.

Ablation Experiment on interaction regions: In this section the values of $R1$ and $R2$ will be varied and the effect of the model will be observed. The values of $R2$ in the experiment range from $[50, 60, 70, 80, 90]$ feet, and the values of $R1$ are $[10, 20, 30, 40, 50]$ in feet. As shown in Fig 5(a), when $R2 = 80$ feet, it can be seen that the model performance at $R1 = 40$ feet outperforms the other values. This is also very much in line with the driver's field of vision distance in real life situations. As shown in Fig 5(b), when $R1 = 40$ feet, it can be seen that the model performance at $R2 = 80$ feet outperforms the other values.

Ablation Experiment on several modules: The model in this paper consists of three main modules, the vision module, the neighbourhood module and the ego module. The first ablation experiments are targeted at the feature cross-fusion block inside the vision module. The purpose of the first ablation experiment is to test whether the feature cross-fusion block in this paper can amplify the effect of the observed vehicle on the ego vehicle and improve the model performance. The second ablation experiment targets the temporal attention block of the ego module, with the aim of observing whether this attention mechanism can amplify the temporal correlation of vehicle trajectories and improve model performance. The third ablation experiment targets the spatial attention block of the ego module, with the aim of observing whether this attention mechanism can capture spatial correlations between vehicles and improve model performance. As shown in Table 2, the first ablation experiment shows that the cross-fusion block of the vision module can help the model to add more weights to the observed vehicles in the vision interaction region, and better learn the influence of the observed vehicles in the region on the future trajectory of the ego vehicle. The second ablation experiment shows that the temporal attention of the ego module can help the model to

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

TABLE 2: Comparison of RMSE values for ablation studies of the proposed method. These values are converted into the meter unit.

Model	$RMSE_{1s}$	$RMSE_{2s}$	$RMSE_{3s}$	$RMSE_{4s}$	$RMSE_{5s}$
CS-LSTM [20]	0.62	1.29	2.13	3.20	4.52
IA-CSTM(without spatial attention)	0.59	1.26	2.09	3.16	4.48
IA-CSTM(without temporal attention)	0.58	1.24	2.03	3.08	4.36
IA-CSTM(without cross-fusion)	0.58	1.23	2.02	3.01	4.15
IA-CSTM	0.56	1.22	2.01	2.98	4.11

TABLE 3: Comparison of RMSE values with other baseline models. These values are converted into the meter unit.

Model	$RMSE_{1s}$	$RMSE_{2s}$	$RMSE_{3s}$	$RMSE_{4s}$	$RMSE_{5s}$
CV [20]	0.73	1.78	3.13	4.78	6.68
V-LSTM [20]	0.68	1.65	2.91	4.46	6.27
S-LSTM [19]	0.65	1.31	2.16	3.25	4.55
CS-LSTM [20]	0.61	1.27	2.09	3.10	4.37
CS-LSTM(M) [20]	0.62	1.29	2.13	3.20	4.52
IA-CSTM(our)	0.56	1.22	2.01	2.98	4.11

enhance the temporal correlation information between the ego vehicle and its own historical trajectory, helping the model to improve the prediction accuracy. The third ablation experiment shows that the spatial attention of the ego module can help the model to extract the spatial correlation between the surrounding vehicles and the ego vehicle, which can greatly improve the accuracy of the model. Based on the comparison of the method proposed in the three ablation experiments with the CS-LSTM(M) model, it can be seen the importance of constructing two interaction regions in this paper, as well as the feasibility of incorporating the spatial-temporal attention mechanism.

Vehicles in the vision interaction region more strongly with the ego vehicle and vehicles in the neighborhood interaction region less strongly with the ego vehicle. Grouping the vehicles around the ego vehicle into two interaction regions is more effective than a single region because it enhances the model's perception of the vehicles in the vision region. The model augments the trajectory features of vehicles in the vision region with cross-fussion block, which is equivalent to augmenting the interaction information between vehicles. Meanwhile, the attention mechanism in the Ego module connects all the vehicles in the two interaction regions, preserves the interaction effects of all the interacting vehicles, and extracts the effective information to improve the accuracy of the model.

C. COMPARED MODELS

In this paper, we compare the IA-CSTM model to the following baselines:

- *Constant Velocity (CV):* This method simply uses a constant velocity Kalman filter to predict the trajectory.
- *Vanilla LSTM (V-LSTM):* This method does not consider interactions, but uses an LSTM-based encoder-decoder structure for prediction.
- *LSTM with fully connected social pooling (S-LSTM):* The method fuses the historical trajectories of vehicles around the target and uses a fully connected layer in the decoding to fuse the coded representations of the target vehicle and its surrounding vehicles.
- *LSTM with convolutional social pooling (CS-LSTM):* This method applies a convolutional social pooling layer that considers the interaction between the target and its surrounding vehicles based on a spatial grid. Its output is a unimodal trajectory distribution.
- *CS-LSTM(M):* The model outputs a strategy-based multimodal trajectory distribution. The modes used for evaluation are those with the highest probability.

Table 3 presents the *RMSE* values of the methods compared. We can observe that the *RMSE* values of CV and V-LSTM are much higher than the other models. These two models use only the trajectory history information of the target vehicle, while the other models utilise the motion information of the surrounding vehicles. This result suggests that taking into account the interactions between vehicles is crucial for trajectory prediction. The RMSE value of this paper's model is lower compared to both CS-LSTM and CS-LSTM(M) models. This result indicates that the hierarchical consideration of the degree of influence of surrounding vehicles on the vehicle is necessary in this paper, and that the enhancement of the temporal correlation of the vehicle's historical trajectories as well as the spatial correlation between the vehicles with the attentional mechanism is necessary.

D. VISUALIZATION OF PREDICTION RESULTS

In this paper, we visualise the prediction results of the same car in the same scene using the method proposed by the ablation experiment and the CS-LSTM. As can be seen from the visualisation results in Fig 6(a) and Fig 6(c), the spatial attention operation in the model has greatly improved the accuracy of the model predictions, as evidenced by the experimental results in Table 2. Similarly, as can be seen from the visualisation results in Fig 6(a) and Fig 6(b), the cross-fusion block in the model channel one also has a certain enhancement effect on the prediction results of the model, which is also proved by the experimental results in Table 2. From the visualisation results in Fig 6(a) and Fig 6(d), it can be seen that the results predicted by the model in this paper

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

are better than those predicted by CS-LSTM [20].

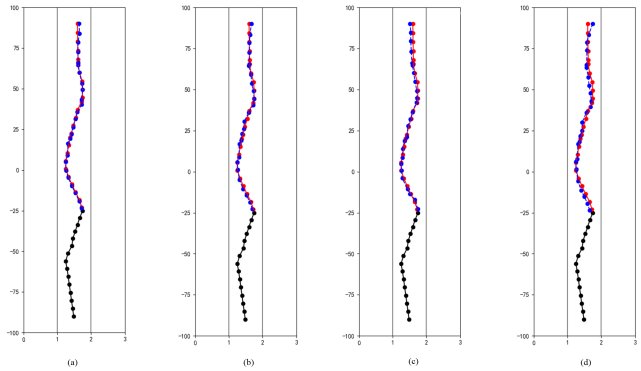


FIGURE 6: Black dots and lines indicate historical vehicle trajectories, red dots and lines indicate real future vehicle trajectories, and blue dots and lines indicate predicted future vehicle trajectories.**(a):** IA-CSTM prediction result for the ego vehicle; **(b):** IA-CSTM (without cross-fusion block) prediction result for the ego vehicle; **(c):** IA-CSTM (without spatial attention) prediction result for the ego vehicle; **(d):** CS-LSTM prediction result for the ego vehicle

VI. CONCLUSION AND FUTURE WORK

In highly dynamic traffic scenes, the vehicle's subsequent movements are affected by the interactions of its surrounding vehicles. This paper proposes a multi-module model. In this paper, two interaction regions are constructed for a moving vehicle, the influence of the surrounding vehicles on the ego vehicle is stratified, and the vehicle information in different interaction regions is processed by different modules. The model uses a spatial-temporal attention mechanism to enhance and capture the temporal relevance of the vehicle's historical trajectory and the spatial relevance of the surrounding vehicles, respectively. The model focuses on vehicle-to-vehicle interactions. Our proposed model outperforms many benchmark models on two large public datasets.

The datasets used for model training consists only of highway sections, which is simpler than urban traffic. Therefore, the model has to be improved on complex traffic scenarios. We will later transform the model inputs into multimodal data types. All the agents in the urban traffic scenario are constructed into an undirected graph, which can reflect the interaction information between each traffic agent more intuitively. The inputs to the model can have both undirected graph and vehicle trajectory coordinates, and the model is used to output the nature of the interaction between each road agent. At the same time, the inputs of multimodal data types can provide the model with a more comprehensive perspective, which helps to improve the model's understanding of the problem and its ability to solve it.

REFERENCES

[1] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings and A. Mouzakitis, "Deep Learning-Based Vehicle Behavior Prediction for Autonomous Driving Applications: A Review," in IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 1, pp. 33-47, Jan. 2022, doi: 10.1109/TITS.2020.3012034.

[2] A. Houenou, P. Bonnifait, V. Cherfaoui and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 2013, pp. 4363-4369, doi: 10.1109/IROS.2013.6696982.

[3] M. Schreier, V. Willert and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 2014, pp. 334-341, doi: 10.1109/ITSC.2014.6957713.

[4] A. Bhattacharyya, M. Fritz and B. Schiele, "Long-Term On-board Prediction of People in Traffic Scenes Under Uncertainty," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 4194-4202, doi: 10.1109/CVPR.2018.00441.

[5] Chiu-Feng Lin and A. G. Ulsoy, "Vehicle dynamics and external disturbance estimation for future vehicle path prediction," Proceedings of 1995 American Control Conference - ACC'95, Seattle, WA, USA, 1995, pp. 155-159 vol.1, doi: 10.1109/ACC.1995.529227.

[6] V. Lefkopoulos, M. Menner, A. Domahidi and M. N. Zeilinger, "Interaction-Aware Motion Prediction for Autonomous Driving: A Multiple Model Kalman Filtering Scheme," in IEEE Robotics and Automation Letters, vol. 6, no. 1, pp. 80-87, Jan. 2021, doi: 10.1109/LRA.2020.3032079.

[7] P. Lytrivis, G. Thomaidis and A. Amditis, "Cooperative Path Prediction in Vehicular Environments," 2008 11th International IEEE Conference on Intelligent Transportation Systems, Beijing, China, 2008, pp. 803-808, doi: 10.1109/ITSC.2008.4732629.

[8] S. Lefèvre, C. Laugier and J. Ibañez-Guzmán, "Exploiting map information for driver intention estimation at road intersections," 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany, 2011, pp. 583-588, doi: 10.1109/IVS.2011.5940452.

[9] S. Qiao, D. Shen, X. Wang, N. Han and W. Zhu, "A Self-Adaptive Parameter Selection Trajectory Prediction Approach via Hidden Markov Models," in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 1, pp. 284-296, Feb. 2015, doi: 10.1109/TITS.2014.2331758.

[10] S. Zhang, Y. Zhi, R. He and J. Li, "Research on Traffic Vehicle Behavior Prediction Method Based on Game Theory and HMM," in IEEE Access, vol. 8, pp. 30210-30222, 2020, doi: 10.1109/ACCESS.2020.2971705.

[11] Z. Hao, X. Huang, K. Wang, M. Cui and Y. Tian, "Attention -Based GRU for Driver Intention Recognition and Vehicle Trajectory Prediction," 2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI), Hangzhou, China, 2020, pp. 86-91, doi: 10.1109/CVCI51460.2020.9338510.

[12] Z. Zhang, R. Yang and Y. Fang, "LSTM Network Based on on Antlion Optimization and its Application in Flight Trajectory Prediction," 2018 2nd IEEE Advanced Information Management,Communicates,Electronic and Automation Control Conference (IMCEC), Xi'an, China, 2018, pp. 1658-1662, doi: 10.1109/IMCEC.2018.8469476.

[13] R. Fujii, J. Vongkulbhisal, R. Hachiuma and H. Saito, "A Two-Block RNN-Based Trajectory Prediction From Incomplete Trajectory," in IEEE Access, vol. 9, pp. 56140-56151, 2021, doi: 10.1109/ACCESS.2021.3072135.

[14] N. Bhujel, E. K. Teoh and W. -Y. Yau, "Pedestrian Trajectory Prediction Using RNN Encoder-Decoder with Spatio-Temporal Attentions," 2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR), Singapore, 2019, pp. 110-114, doi: 10.1109/ICMSR.2019.8835478.

[15] T. Yang, Z. Nan, H. Zhang, S. Chen and N. Zheng, "Traffic Agent Trajectory Prediction Using Social Convolution and Attention Mechanism," 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 2020, pp. 278-283, doi: 10.1109/IV47402.2020.9304645.

[16] Li, Xin et al. "GRIP: Graph-based Interaction-aware Trajectory Prediction." 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (2019): 3960-3966.

[17] E. Güney, C. Bayilmiş and B. Çakan, "An Implementation of Real-Time Traffic Signs and Road Objects Detection Based on Mobile GPU Platforms," in IEEE Access, vol. 10, pp. 86191-86203, 2022, doi: 10.1109/ACCESS.2022.3198954.

[18] Güney, Emine Dilay and Cuneyt Bayilmis. "An Implementation of Traffic Signs and Road Objects Detection Using Faster R-CNN." Sakarya University Journal of Computer and Information Sciences (2022): n. pag.

[19] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei and S. Savarese, "Social LSTM: Human Trajectory Prediction in Crowded Spaces," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 961-971, doi: 10.1109/CVPR.2016.110.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3335091

[20] N. Deo and M. M. Trivedi, "Convolutional Social Pooling for Vehicle Trajectory Prediction," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 2018, pp. 1549-15498, doi: 10.1109/CVPRW.2018.00196.
[21] Clevert, Djork-Arné et al. "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)." arXiv: Learning (2015): n. pag.
[22] Kingma, Diederik P. and Jimmy Ba. "Adam: A Method for Stochastic Optimization." CoRR abs/1412.6980 (2014): n. pag.
[23] M. Á. D. Miguel, J. M. Armingol and F. García, "Vehicles Trajectory Prediction Using Recurrent VAE Network," in IEEE Access, vol. 10, pp. 32742-32749, 2022, doi: 10.1109/ACCESS.2022.3161661.
[24] Seff, Ari et al. "MotionLM: Multi-Agent Motion Forecasting as Language Modeling." (2023), arXiv:2309.16534.



DENGYANG CHENG born in October 1999, is from Nantong City, Jiangsu Province. He graduated from Huaiyin Institute of Technology and am currently a graduate student at Nantong University. As a student of Professor Gu Xiang, his research direction is Internet of Vehicles, which is subdivided into vehicle trajectory prediction.



XIANG GU born in June 1973, professor, doctor, master supervisor. He graduated from Nanjing University with a bachelor's degree in Electronics and Information System in July 1995. In April 2001, he graduated from Nanjing University of Posts and Telecommunications with a master's degree in engineering. In July 2004, he graduated from University of Science and Technology of China with a master's degree in Computer science and technology. From June 2013 to August 2013, short-term study in University of Colorado Springs in the United States; May 2014, Postdoctoral Postdoctoral Station of Information and Communication Engineering, Nanjing University of Posts and Telecommunications; April 2015 to March 2016 Visiting scholar at Arizona State University.



CONG QIAN born in July 1997, male, Han nationality, from Wuhu City, Anhui Province. He studied at Bengbu University for my undergraduate degree and am currently pursuing a master's degree at Nantong University under the guidance of Professor Gu Xiang. Currently, his research direction is mainly the Internet of Vehicles, and He mainly study vehicle driving behavior.



CHAONAN DU born in April 1998, female, Han nationality, from Heze City, Shandong Province. She studied in Shandong Jianzhu University as an undergraduate, and now she is studying for a master's degree in Nantong University under the supervision of Professor Gu Xiang. At present, her research direction is mainly the Internet of vehicles, and I mainly study lane detection.



JIN WANG , male, PhD, Associate professor. He is a member of the Association for Computing Machinery (ACM), a member of the IEEE Computer Society, and a senior member of the Chinese Computer Society (CCF). From September 2000 to July 2004, he graduated from Nanjing University of Science and Technology with a bachelor's degree in Computer Science and Technology. From September 2004 to July 2009, I studied pattern recognition and Intelligent System in Nanjing University of Science and Technology, and received a Doctor's degree in engineering. From December 2013 to December 2014, I went to Arizona State University for further study for one year.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4