A Hierarchical Framework for Interactive Behaviour Prediction of Heterogeneous Traffic Participants Based on Graph Neural Network

Zirui Li, Chao Lu, *Member, IEEE*, Yangtian Yi, and Jianwei Gong, *Member, IEEE*

***Abstract*— In complex and dynamic urban traffic scenarios, the accurate prediction of trajectories of surrounding traffic participants (vehicles, pedestrians, etc) with interactive behaviours plays an important role in the navigation and the motion planning of the ego vehicle. In this paper, based on the graph neural network (GNN), we propose a hierarchical GNN framework to model interactions of heterogeneous traffic participants (vehicles, pedestrians and riders) combined with LSTM to predict their trajectories. The proposed framework consists of two modules with two GNNs for interactive events recognition (IER) and trajectory prediction (TP). The IER module is used to recognise interactive events between traffic participants and the ego vehicle. With the recognised results as the input, the TP module is built for interactive trajectory prediction. In addition, to realise the multi-step prediction, a long short-term memory network (LSTM) is combined with GNN in the TP module. The proposed hierarchical framework is verified by the naturalistic driving data collected from the urban traffic environment. Comparative results with state-of-the-art methods indicate that the hierarchical GNN framework obtains an outstanding performance in the recognition of interactive events and the prediction of interactive behaviours.**

***Index Terms*— Trajectory prediction, interactive behaviours, graph neural network, heterogeneous traffic participants.**

I. Introduction

SAFELY driving in the urban environment is a great challenge for autonomous driving systems due to dynamic and complicated traffic situations involving heterogeneous traffic participants (vehicles, pedestrians, etc). Accurately modelling, understanding and predicting behaviours of traffic participants have a significant effect on the motion planning and the control for autonomous vehicles [1]–[4]. With this in mind, many researchers have made efforts to model the behaviours of traffic participants and predict their trajectories. According to the specific and detailed modelling process, previous works are divided into three categories: physical-model-based methods, manoeuvre-based methods and interaction-aware methods [3].

The physical-model-based methods focus on physical characteristics of vehicles and predict the trajectories of vehicles by kinematic and dynamic models, which are built based on control inputs (e.g. acceleration, velocity and steering), vehicle properties (e.g. mass and wheel base) and environmental conditions (e.g. speed limit and road type) [3], [5], [6]. However, riders and pedestrians reflect higher uncertainty in motion prediction compared to vehicles and do not have explicit kinematic and dynamic models. Therefore, the physical models cannot accurately predict trajectories of riders and pedestrians. In addition, physical-model-based methods are unable to model the interactions and influence among traffic participants, which is important for the trajectory prediction and the behaviour understanding [3].

Due to the disadvantages of the physical models, intentions of traffic participants are considered in the manoeuvre-based methods which model the future motion of traffic participants based on their current and historical manoeuvres [7]. The manoeuvre-based methods usually consist of two steps: manoeuvre recognition and trajectory prediction. The manoeuvres are defined as a series of movement patterns or driver intentions. The step of manoeuvre recognition firstly recognises intentions or patterns whose types are pre-defined. For example, in the non-signalised intersection, the manoeuvres of vehicles can be defined as: “turn left”, “turn right” and “go straight” [8]. According to recognised results, the trajectory prediction step outputs the future trajectories based on the pre-trained model in each manoeuvre group [7]. In the manoeuvre recognition step, statistical-learning-based methods, such as SVM [9], [10], HMM [9], [11], GMM, Bayesian Network [12], random forest classifiers [13], manifold alignment [14]–[16] and transfer learning [17]–[19], are developed and applied to recognise manoeuvres of traffic participants. Except for statistical learning, artificial neural network, such as recurrent neural networks (RNN) are other choices for manoeuvres recognition [20]. In the step of trajectory prediction, widely-used methods include rapidly-exploring random tree (RRT) [21], Gaussian process (GP) [22] and cluster-based models [23], [11]. However, these methods are unable to predict long-term trajectories because the above methods cannot model the relationship in time series to process sequences of inputs. To overcome this limitation, RNN-based methods are developed in recent studies [24], [25]. Although the manoeuvre-based methods can take intentions of traffic participants into account and make the predicted trajectories interpretable, these approaches have their own limitations. In dynamic and complicated traffic scenarios, each traffic

Manuscript received 5 February 2020; revised 26 July 2020, 13 November 2020, and 26 February 2021; accepted 2 June 2021. Date of publication 29 June 2021; date of current version 8 July 2022. This work was supported by the National Natural Science Foundation of China under Grant 61703041 and Grant U19A2083. The Associate Editor for this article was P. Ye. *(Zirui Li and Chao Lu contributed equally to this work.) (Corresponding authors: Jianwei Gong; Chao Lu.)*

The authors are with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China (e-mail: 3120195255@bit.edu.cn; chaolu@bit.edu.cn; 1120160838@bit.edu.cn; gongjianwei@bit.edu.cn).

Digital Object Identifier 10.1109/TITS.2021.3090851

participant in manoeuvre-based studies is modelled individually and the relationships among different participants are not considered.

The interaction-aware methods provide a better understanding of the influence and dependencies among traffic participants, which is important for accurate prediction of trajectories [26]–[30]. In these researches, trajectories predicted by the interaction-aware methods are named as interactive trajectories. In [26], an extension of RNN, social LSTM (long short-term memory) network is proposed to model the interactions among pedestrians as social behaviours. Here, "social" represents the impact for one pedestrian from their neighbours, which is realised by encoding the features of the pedestrians with the social pooling mechanism. LSTM is applied to model the interactions between different pedestrians in time series. Based on social LSTM, social GAN (generative adversarial network) are developed to predict interactive behaviours successively [31]. However, these "social" aware methods describe interactions between traffic participants implicitly, which are unable to quantitatively interpret the degrees of interactions. To solve this problem, some researchers proposed to model the interactions between traffic participants by the graph structure with nodes and edges [32]–[34]. In [33], each node in the graph structure can be modelled by an LSTM network to predict long-term trajectories of vehicles in the highway. The graph structure is also applied in the prediction of the traffic flow for intelligent transportation systems (ITS) [35]–[38]. Considering the interactions between different types of traffic participants, [39] proposed a graph neural network (GNN) named *Trafficpredict* to predict trajectories of heterogeneous participants. However, the explicit relationship with the semantic definition among different traffic participants are not considered in *Trafficpredict*. In this research, the definition above is named as interactive events [40]. Besides, the features of interactive trajectories, interactive events between two traffic participants can be applied to model interactive behaviours [40]. Following [40], the interactive event considered in this paper refer to the events involving at least two traffic participants when interactions happen between them. Typical interactive events include overtaking from left (right), driving away from left (right), parallel driving in left (right), etc. And these can cover most of the situations for vehicles, riders and pedestrians. The detailed descriptions of these events can be found from [40]. Specific interactive events of traffic participants, as a main component of interactive behaviours, are neglected in [33] and [39], which reduce the potential of GNN for the interaction-aware trajectory prediction. In this paper, a hierarchical GNN-based framework for the interactive behaviours prediction of heterogeneous traffic participants is proposed. The proposed framework consists of two modules, IER and TP, which combines the advantages of manoeuvre-based and interaction-aware methods.

Main contributions of this paper are as follows:

1. A novel hierarchical GNN framework is proposed for the interactive behaviours prediction of heterogeneous traffic participants. The framework firstly recognises interactive events among traffic participants and the ego vehicle. Secondly, recognised results are combined with historical trajectories to predict future trajectories.
2. Considering that different types of traffic participants have different characteristics of motion patterns, a new layer representing the feature of the participants with the same type is applied in GNN to model the similarities of traffic participants.
3. The proposed framework combines manoeuvre-based and interaction-aware methods within a hierarchical structure, which obtains the advantages from both sides.

The rest of this paper is arranged as follows. The description of proposed hierarchical GNN framework is presented in section II. Section III details the problem formulation and the methodology. Experimental settings and comparative results are shown in section IV. Conclusion and future works are summarised in section V.

II. Description of the Hierarchical Framework

To predict interactive behaviours of heterogeneous traffic participants in the dynamic urban environment, a hierarchical GNN-based framework is proposed in this paper. The spatial interactive behaviours are modelled by graph structures in the proposed framework. It represents the mutual influence among traffic participants at the same time, which is measured by the spatial distance between two participants. And interactions in time series are captured and modelled by LSTMs, which describe the correlation and dependency of sequential inputs. LSTMs are applied on nodes and edges in the graph, which construct the GNN structure. In this research, interactive behaviours are modelled and predicted from the two perspectives: interactive events and trajectories, which are realised by interactive events recognition (IER) and trajectory prediction (TP) modules in the hierarchical framework, respectively. The illustration of the framework is shown in Fig.1.

In the IER module, an X-to-1 GNN is trained on the basis of the historical trajectory information and is used to recognise interactive events among traffic participants. The "X-to-1" indicates that interactions in time series are modelled by X-to-1 LSTM with sequential inputs and single output (recognition results of interactive events). In the X-to-1 GNN, each traffic participant is modelled as a node in the graph structure, and the interaction between two participants is represented by the spatial edge connecting two spatially distributed nodes. An X-to-1 LSTM is applied to each node and edge to describe the correlation in time series. The IER module can be represented by the function ϕ_{IER}:

$$\mathbf{O}_{\mathrm{IER}} = \phi_{\mathrm{IER}}(\mathbf{s}_{t-n:t}) \tag{1}$$

where $\mathbf{s}_{t-n:t} = [\mathbf{s}^{t-n}, \ldots, \mathbf{s}^{t-1}, \mathbf{s}^{t}]$ is a set of inputs indexed from time $t-n$ to t and $\mathbf{O}_{\mathrm{IER}}$ is a set containing recognised results for interactive events which are labelled according to different traffic participants. At time t, the feature of input $\mathbf{s}^t$ is defined as follow:

$$\mathbf{s}^t = (x^t, y^t, c^t) \tag{2}$$

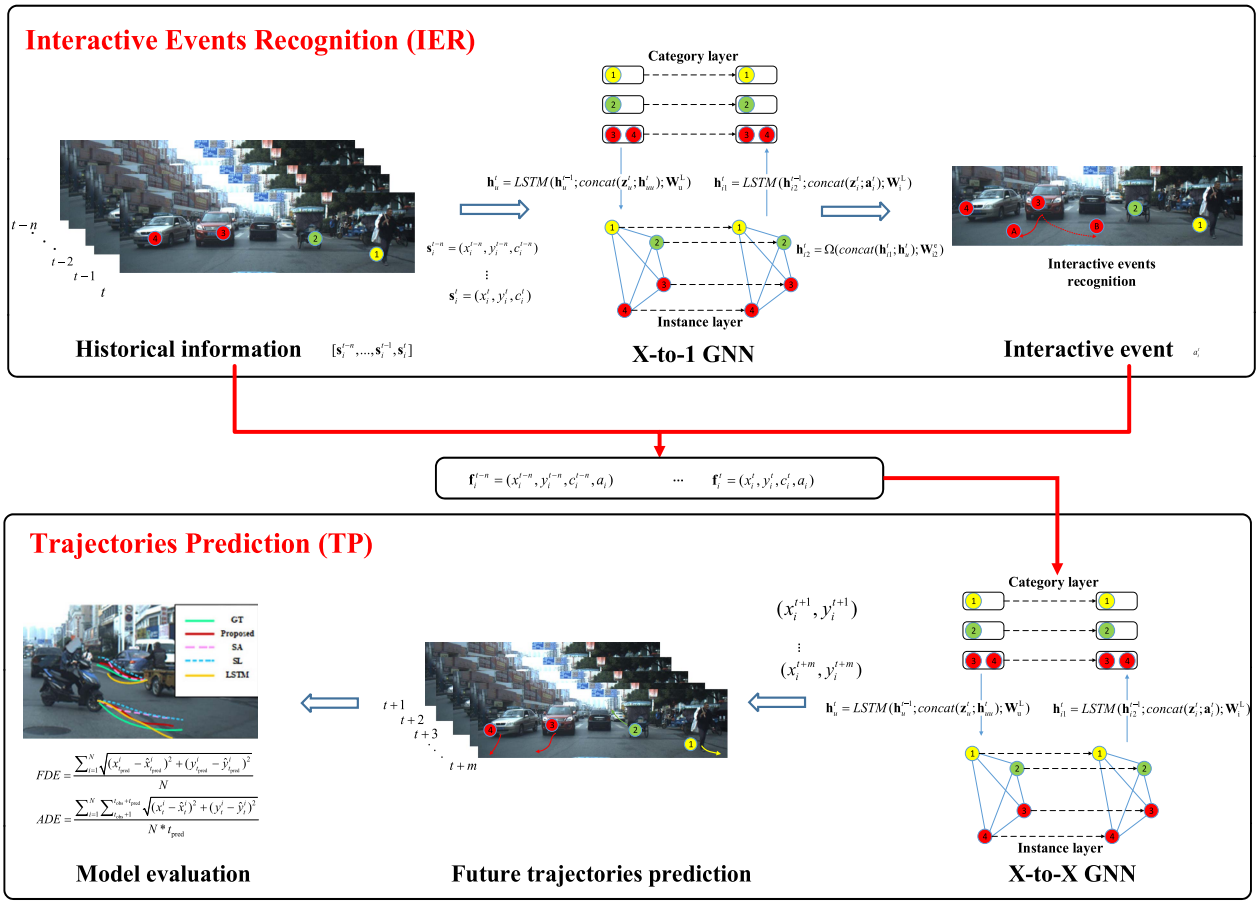

Fig. 1. The proposed hierarchical GNN-based framework.

In the module of TP, historical trajectories and interactive events recognised by the IER module are integrated as the input of an X-to-X GNN. The "X-to-X" means that an X-to-X LSTM is applied to model interactions in time series with sequential inputs/outputs. Different from the X-to-1 LSTM in the IER module, LSTMs in the TP module output sequential trajectories. To guarantee the consistence of the representation of traffic participants, the same graph structure used for IER is applied to TP. The function of TP module can be illustrated by the following equation.

$$\mathbf{O}_{\mathrm{TP}} = \phi_{\mathrm{TP}}(\mathbf{O}_{\mathrm{IER}}, \mathbf{s}_{t-n:t}) \tag{3}$$

where $\mathbf{O}_{TP} = [x^{t+1}, y^{t+1}; \ldots; x^{t+m-1}, y^{t+m-1}; x^{t+m}, y^{t+m}]$ are predicted trajectories for the instance nodes from time $t+1$ to $t+m$. For both X-to-1 and X-to-X GNN models, a similar instance layer proposed in [33] is constructed for different traffic participants as a graph structure. Considering that the traffic participants with different types have different movement patterns [39], a parallel layer named the category layer, representing features of different participants, is built.

The hierarchical framework is proposed to predict trajectories of traffic participants by modelling interactions among traffic participants in dynamic urban scenarios. At time t, the feature of i^{th} traffic participant is denoted as:

$$\mathbf{f}_i^t = [\mathbf{s}_i^t, a_i^t] = [x_i^t, y_i^t, c_i^t, a_i^t] \tag{4}$$

where x and y are coordinates in x-axis and y-axis. a is the labelled interactive event, and c is the type of i^{th} traffic participant. The proposed framework is a general framework, which can be applied in the scenario with arbitrary number of instances. In this work, three types of traffic participants, vehicle, pedestrian and rider, are considered.

In the proposed framework, to guarantee the consistence of the representation for traffic participants, IER and TP modules are developed based on the same graph structure $G = (A_{\mathrm{instance}}, A_{\mathrm{category}}, E_{\mathrm{Spatial}}, E_{\mathrm{Temporal}})$. Here, the instance node A_{instance} represents the traffic participant in the instance layer with feature $\mathbf{f}_i$. Traffic participants with labelled interactive events are modelled as nodes with the form $A_{\mathrm{instance}} = [x, y, c, a]$. The super node A_{category} is constructed to model the similarity of instance nodes with the same type. Two kinds of edges, the spatial edge E_{Spatial} and the temporal edge E_{Temporal} are used to represent spatial and temporal characteristics of interactive behaviours. Specifically, the interaction between two traffic participants i and j at time t are modelled as the spatial edge $E_{\mathrm{Spatial}}^{t,ij} = (A_i^t, A_j^t)$. The spatial edge $E_{\mathrm{Spatial}}^{t,ij}$ for A_i^t is calculated as $\mathbf{f}_{ij}^t = (x_{ij}^t, y_{ij}^t, c_{ij}^t, a_{ij}^t)$, where $x_{ij}^t = x_i^t - x_j^t$ and stands for relative positions from A_j^t to A_i^t, respectively. The unique encoder is applied to represent c_{ij}^t and a_{ij}^t, which are unique encoding of the spatial edge $E_{\mathrm{Spatial}}^{t,ij}$ for types of traffic participants and interactive events. The spatial edge from A_j^t to A_i^t is donated as

$E^{t,ji}_{\text{Spatial}} = (A^t_j, A^t_i)$. Similarly, the correlation and dependency of same traffic-agent in adjacent frames is defined as temporal edges $E^{t,ii}_{\text{Temporal}} = (A^t_i, A^{t+1}_i)$, which is used to share the historical information in the temporal aspect. In this research, frames represent the combination of pictures and features at each timestep. The feature of temporal edge $E^{t,ii}_{\text{Temporal}}$ can be described in the same way by substituting A^t_j with A^{t+1}_i.

The definition above illustrates the problem formulation of the instance layer. In the category layer, a super node A_{category} is used to describe similar movement patterns for each type of traffic participants. All instance nodes with the same type c_i in the instance layer are integrated in one group. To transfer the information between instance and category layers, each instance node has an edge oriented toward super node A^i_{category}. In addition, after modelling the similarity of movement patterns, the super node A^i_{category} passes back the information by the directed edge oriented toward the group of instance nodes. Similarly, the temporal edges are also constructed among super nodes. Temporal edges are used to represent the environmental information and model the interactions between heterogeneous traffic participants. The relationship between LSTMs, nodes and edges in two modules are illustrated in Fig.2. IER and TP modules are presented from top to bottom. Sequential inputs are fed into the IER module through $\mathbf{f}^t_{21}$, $\mathbf{f}^t_{31}$ and $\mathbf{f}^t_{11}$, which are used to construct the graph edge E, calculate hidden states $\mathbf{h}$ of LSTMs and generate recognised interactive events a. The historical information and recognised results in the IER module are combined as the input to the TP module, which has a similar GNN structure of the IER module. Finally, the TP module outputs predicted trajectories $[x_i^{t+1:t+\text{pred}}, y_i^{t+1:t+\text{pred}}]$.

III. The Formulation of GNN

In the proposed hierarchical GNN-based framework, IER and TP modules are developed based on GNN which consists of two layers, the instance layer and the category layer. The instance layer is used to learn motion characteristics of traffic participants, and the category layer is built to model movement patterns of traffic participants with the same type. Two layers are detailed as follows. Statements of primary parameters of the instance layer and the category layer are detailed in Table I and Table II, respectively. In Fig.3, a schematic diagram is presented to explain the formulation of the instance and the category layers. The temporal edge LSTM, node LSTM, spatial edge LSTM and information transfer are represented by four types of arrows. $E^{t,31}_{Spatial}$ from node #3 and $E^{t,21}_{Spatial}$ from node #2 are combined as $\mathbf{H}^t_1$ at time t. The information transferred between the super node #1 and the instance node #1 are presented by $\mathbf{d}^t_1$ and h^t_u.

A. Instance Layer

Characteristics of instances (or traffic participants) in traffic are captured by the instance layer. For each instance node A_{instance}, an LSTM is assigned to predict the node changes. Considering that traffic participants with different types have different characteristics and motion patterns, instance nodes in the same category have the same parameters. In this research,

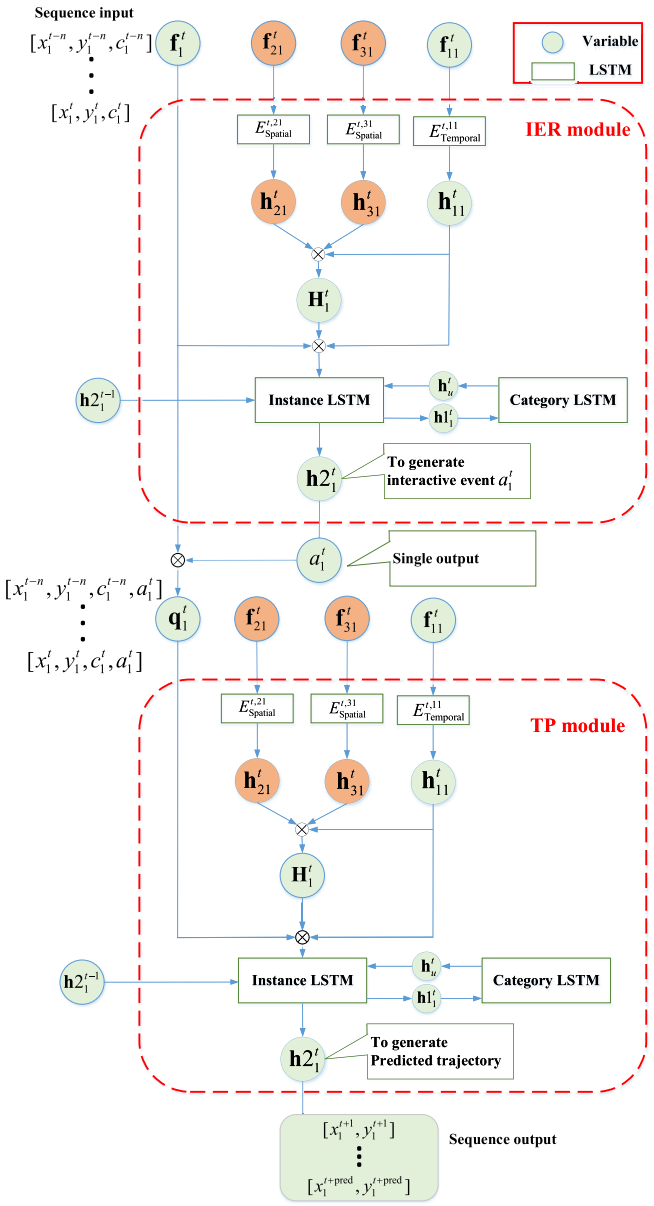

Fig. 2. The detailed illustration of proposed framework.

three LSTMs are trained in the instance layer for pedestrians, vehicles and riders. Features of E_{Spatial} and E_{Temporal} are sent to spatial edge LSTMs and temporal edge LSTMs, which are applied to model spatial edges and temporal edges, respectively. All spatial edges share the same parameters for LSTMs and all temporal edges are divided into three categories according to the corresponding node type. Primary variables in the instance layer are detailed in Table I.

At time t, the feature $\mathbf{f}^t_{ij}$ of spatial edge $E^{t,ij}_{\text{Spatial}} = (A^t_i, A^t_j)$ is embedded into a fixed vector $\mathbf{z}^t_{ij}$, which is the input to the edge LSTM L_{ij}:

$$\mathbf{z}^t_{ij} = \Omega(\mathbf{f}^t_{ij}, \mathbf{W}^{\text{e}}_{\text{spa}}) \tag{5}$$

where $\Omega(\cdot,\cdot)$ is a linear embedding function, and $\mathbf{W}^{\text{e}}_{\text{spa}}$ are weights of the embedding layer. Then, $\mathbf{z}^t_{ij}$ will be sent to

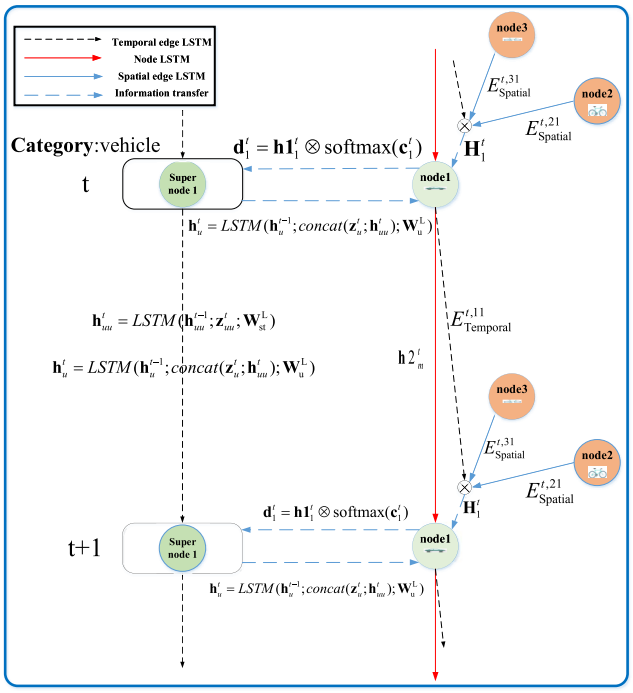

Fig. 3. The relationship and information transfer between variables in GNN.

TABLE I
STATEMENTS OF PRIMARY VARIABLES IN THE INSTANCE LAYER

Parameters	Annotation
A_{instance}	Instance nodes
E_{Spatial}	Spatial edges
E_{Temporal}	Temporal edges
L_{ij}	The spatial edge LSTM
L_{ii}	The temporal edge LSTM
L_i	The instance LSTM
$\mathbf{h}_{ij}^t$	The hidden state in spatial edge LSTM
$\mathbf{h}_{ii}^t$	The hidden state in spatial edge LSTM
$\mathbf{H}_i^t$	The weighted sum of $\mathbf{h}_{ij}^t$
$\mathbf{h}1_i^t$	The first hidden state of the instance node LSTM
$\mathbf{h}2_i^{t-1}$	The final hidden state of the instance node LSTM

LSTM L_{ij} and generate the hidden state $\mathbf{h}_{ij}^t$:

$$\mathbf{h}_{ij}^t = LSTM(\mathbf{h}_{ij}^{t-1}, \mathbf{z}_{ij}^t, \mathbf{W}_{\text{spa}}^{\text{L}}) \tag{6}$$

where $\mathbf{W}_{\text{spa}}^{\text{L}}$ are weights of the spatial edge LSTM. $\mathbf{h}_{ij}^t$ contains the information of spatial relations. The temporal edge $E_{\text{Temporal}}^{t,ij} = (A_i^t, A_i^{t+1})$ is defined in the same way as the spatial edge. Similarly, the output $\mathbf{h}_{ii}^t$ of the temporal edge LSTM L_{ii} contains the information in time series.

In the urban environment, each traffic participant may interact with several surrounding participants, and the importance of which may not be the same. To quantify the importance of surrounding traffic participants, the proposed GNN framework uses the soft attention mechanism mentioned in [41] to assign different weight w to different spatial edges of an instance node:

$$w(\mathbf{h}_{ij}^t) = softamx(\frac{k}{\sqrt{d_e}} Dot(\mathbf{W}_{ii}\mathbf{h}_{ii}^t, \mathbf{W}_{ij}\mathbf{h}_{ij}^t)) \tag{7}$$

where $\mathbf{W}_{ii}$ and $\mathbf{W}_{ij}$ are embedding weights for spatial and temporal edges, respectively. $Dot(\cdot,\cdot)$ is the dot product, and $k/\sqrt{d_e}$ is the scaling factor. Finally, impacts of the surrounding traffic participants in the spatial view are calculated by the weighted sum of $\mathbf{h}_{ij}^t$, which is denoted by $\mathbf{H}_i^t$. And the temporal influence on trajectories of i^{th} traffic participant is denoted as $\mathbf{h}_{ii}^t$. Thus, $\mathbf{H}_i^t$ and $\mathbf{h}_{ii}^t$ are concatenated and embedded into the fixed vector $\mathbf{a}_i^t$, which will be concatenated with the feature of instance node $\mathbf{f}_i^t$ as the input to instance LSTM L_i:

$$\mathbf{z}_i^t = \Omega(\mathbf{f}_i^t, \mathbf{W}_{\text{instance}}^{\text{node}}) \tag{8}$$

$$\mathbf{a}_i^t = \Omega(concat(\mathbf{h}_{ii}^t; \mathbf{H}_i^t); \mathbf{W}_{\text{instance}}^{\text{edge}}) \tag{9}$$

$$\mathbf{h}1_i^t = LSTM(\mathbf{h}2_i^{t-1}; concat(\mathbf{z}_i^t; \mathbf{a}_i^t); \mathbf{W}_{\text{instance}}^{\text{L}}) \tag{10}$$

where $\mathbf{W}_{\text{instance}}^{\text{node}}$ and $\mathbf{W}_{\text{instance}}^{\text{edge}}$ are embedding weights, and $\mathbf{W}_{\text{instance}}^{\text{L}}$ is the weight of i^{th} instance node LSTM cell. $\mathbf{h}1_i^t$ and $\mathbf{h}2_i^{t-1}$ are the first and the final hidden states of the instance node LSTM, which are illustrated in Section III. (B).

B. Category Layer

By distinguishing different categories of the participants and setting parameters for each type, the model can make an accurate prediction. Considering that different traffic participants have different characteristics, the category layer is applied in the proposed framework to capture category properties.

Each frame consists of one image and labels for trajectories. Every category of participants passes information about their characteristics by frame in time series. Therefore, for each type of participants, super nodes A_{category}^u, $u \in \{1, 2, 3\}$ are set with LSTM. Similar to the instance nodes, super nodes also have temporal edges in time series, which are shown by dash lines in Fig.1. The category layer consists of the four parts: super nodes, temporal edges for super nodes, directed edges from instance nodes in the same group to super nodes and directed edges from super nodes to instance nodes.

Each traffic participant produces the hidden state $\mathbf{h}1_i^t$ and the state of instance node $\mathbf{c}_i^t$ at time t,which are combined as the movement feature $\mathbf{d}_m^t$ for m^{th} instance node in the category u.

$$\mathbf{d}_m^t = \mathbf{h}1_i^t \otimes \text{softmax}(\mathbf{c}_i^t) \tag{11}$$

The feature $\mathbf{F}_u^t$ of the corresponding super node A_{category}^u, $u \in \{1, 2, 3\}$ will be obtained by computing the average of all instance features $d = \{d_m^t\}_{m=1}^n$ belonging to the same category u:

$$\mathbf{F}_u^t = \frac{1}{n}\sum_{m=1}^{n} d_m^t \tag{12}$$

Equation (12) shows that the feature of u^{th} super node $\mathbf{F}_u^t$ takes each instance into account, and captures the characteristics for participants with the same type, which will transmit

TABLE II
STATEMENTS OF PRIMARY PARAMETERS IN THE CATEGORY LAYER

Parameters	Annotation
A_{Category}	Super nodes in the category layer
$\mathbf{d}_m^t$	The movement feature for m^{th} instance node
$\mathbf{F}_u^t$	The feature of the corresponding super node
$\mathbf{F}_{uu}^t$	The feature of temporal edge in the category layer
$\mathbf{h}_{uu}^t$	The hidden state of temporal edge LSTM
$\mathbf{h}_u^t$	The hidden state of category LSTM (super node)
$\mathbf{h}1_m^t$	The hidden state in spatial edge LSTM
$\mathbf{h}2_m^t$	The final output of instance node

the information by the edge from the instance node to the super node.

In the category layer, the feature of the temporal edge $\mathbf{F}_{uu}^t$ is calculated by $\mathbf{F}_u^t - \mathbf{F}_u^{t-1}$, which will be combined with the hidden state $\mathbf{h}_{uu}^t$ to calculate the temporal edge between the same super node in adjacent frames:

$$\mathbf{z}_{uu}^t = \Omega(\mathbf{F}_{uu}^t, \mathbf{W}_{\text{st}}^{\text{e}}) \tag{13}$$

$$\mathbf{h}_{uu}^t = LSTM(\mathbf{h}_{uu}^{t-1}; \mathbf{z}_{uu}^t; \mathbf{W}_{\text{st}}^{\text{L}}) \tag{14}$$

where $\mathbf{W}_{\text{st}}^{\text{e}}$ and $\mathbf{W}_{\text{st}}^{\text{L}}$ are weights of the embedding layer and temporal LSTM cells.

Then, features from the group of instance feature $\mathbf{F}_u^t$ and temporal feature $\mathbf{h}_{uu}^t$ are integrated with the hidden state $\mathbf{h}_u^t$ of the super node as follows:

$$\mathbf{z}_u^t = \Omega(\mathbf{F}_u^t, \mathbf{W}_{\text{u}}^{\text{node}}) \tag{15}$$

$$\mathbf{h}_u^t = LSTM(\mathbf{h}_u^{t-1}; concat(\mathbf{z}_u^t; \mathbf{h}_{uu}^t); \mathbf{W}_{\text{u}}^{\text{L}}) \tag{16}$$

where $\mathbf{W}_{\text{u}}^{\text{e}}$ and $\mathbf{W}_{\text{u}}^{\text{L}}$ are embedding weights and super node LSTM cells.

Finally, $\mathbf{h}_u^t$ will be concatenated with $\mathbf{h}1_m^t$ and sent back to each instance node. The second hidden state $\mathbf{h}2_m^t$ is the final output of the instance node:

$$\mathbf{h}2_m^t = \Omega(concat(\mathbf{h}1_m^t; \mathbf{h}_u^t); \mathbf{W}_m^{\text{e}}) \tag{17}$$

where $\mathbf{W}_{\text{i2}}^{\text{e}}$ are embedding weights and $\mathbf{h}2_m^t$ is the final output of the m^{th} instance node. Primary variables in the category layer are detailed in Table II.

C. Interactive Events Recognition and Trajectory Prediction

The proposed hierarchical framework firstly recognises the interactive events in the IER module and secondly sends results of the recognition into the TP module, which assumes that the positions of the traffic participants in the next frame meet the bivariate Gaussian distribution with mean $\boldsymbol{\mu}_i^t = (\mu_x, \mu_y)_i^t$, standard deviation $\boldsymbol{\sigma}_i^t = (\sigma_x, \sigma_y)_i^t$ and correlation coefficient ρ_i^t. Corresponding positions can be represented by

$$(x_i^t, y_i^t) \sim [\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t] \tag{18}$$

The second hidden state of the instance node is used to predict these parameters using the linear function $\Omega(\cdot, \cdot)$:

$$[\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t] = \Omega(\mathbf{h}_{i2}^{t-1}, \mathbf{W}_f) \tag{19}$$

The loss function in the IER module L_{IER} is defined by the cross-entropy function, because of its advantage in the classification problem [42].

$$L_{\text{IER}}(\mathbf{W}_{\text{spa}}, \mathbf{W}_{\text{tem}}, \mathbf{W}_{\text{ins}}, \mathbf{W}_{\text{st}}, \mathbf{W}_{\text{sup}}, \mathbf{W}_m, \mathbf{W}_{\text{f}}) = cross\_entropy(\mathbf{I}_i^t; s_i^{t+1}) \tag{20}$$

where s_i^{t+1} is the ground truth label for i^{th} traffic participant at time $t+1$. $\mathbf{I}_i^t = \Omega(\mathbf{h}2_i^t, \mathbf{W}_{\text{i}})$ is the embedded vector generated by embedding weights $\mathbf{W}_{\text{i}}$ and the final hidden output of L_i.

According to [39], the loss function in the TP module L_{TP} is defined by the negative log Likelihood:

$$L_{\text{TP}}(\mathbf{W}_{\text{spa}}, \mathbf{W}_{\text{tem}}, \mathbf{W}_{\text{ins}}, \mathbf{W}_{\text{st}}, \mathbf{W}_{\text{sup}}, \mathbf{W}_m, \mathbf{W}_{\text{f}}) = -\sum_{t=T_{\text{obs}}+1}^{T_{\text{pred}}} \log(P(x_i^t, y_i^t \,|\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t)) \tag{21}$$

In the model training process, the target is to minimize loss functions L_{IER} and L_{TP}. At each time step, the proposed hierarchical framework back-propagate the error through instance nodes, category nodes, spatial edges and temporal edges to update parameters ($\mathbf{W}_{\text{spa}}, \mathbf{W}_{\text{tem}}, \mathbf{W}_{\text{ins}}, \mathbf{W}_{\text{st}}, \mathbf{W}_{\text{sup}}, \mathbf{W}_m, \mathbf{W}_{\text{f}}$).

IV. EXPERIMENTS

In this work, the proposed framework is verified in the urban environment with heterogeneous traffic participants. [40] **b**uilt a **l**arge-scale **5**D semantics **b**enchmark (BLVD), a dataset with labels for interactive events, which is selected as testing data. Training algorithms and relevant tests are implemented in Pytorch[1] on a PC with an Intel Core i5-6300HQ at 2.3GHz, 8GB RAM and 960M GPU. Details of the implementation, evaluation metrics, baseline methods and experimental results are presented in the following subsections.

A. Dataset

Previous studies presented in [33], [43]–[45] verify their methods using NGSIM, which have the following limitations: 1) The highway scenario is simpler with similar road conditions compared to the urban scenario. 2) Only trajectories of vehicles are collected, but the proposed framework needs heterogeneous traffic participants to model the complex and dynamic scenario. 3) No label for manoeuvres is provided, which is the foundation of proposed framework. Considering the three drawbacks above, the proposed hierarchical framework is verified by BLVD dataset and compared with state-of-the-art methods [40].

In order to verify the performance of proposed framework, the 5D dataset BLVD with explicit labels for interactions is selected in experiments. Different from other public datasets, BLVD provides a dynamic 5D semantic benchmark (3D+temporal+interactive), including 654 calibrated video clips for three kinds of participants: vehicles, pedestrians and riders (cyclists and motorbikes). And four types of scene conditions are collected in BLVD: daytime & low densities, night time & low densities, daytime & high densities, and night time & high densities. Moreover, 13, 8 and 7 interactive events

[1] https://pytorch.org/

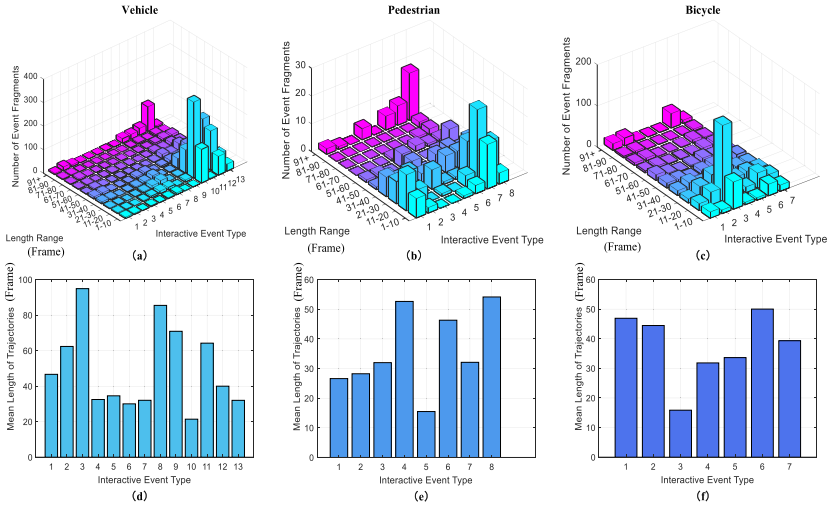

Fig. 4. Detailed characteristics of trajectories.

TABLE III
THE MAIN PARAMETERS IN THE EXPERIMENTAL VALIDATION

Parameter	Value
Temporal edge cell	128
Spatial edge cell	128
Node cell	64
Embedding layer	64
Learning rate	0.001
Epoch	20

between the ego vehicle and traffic participants are labelled in BLVD for vehicles, pedestrians and riders, respectively [40]. Statistics of the number of event fragments with respect to event type and length range of fragments are detailed in Fig.4.

B. Details of Implementation

In the urban traffic, interactive scenarios involving pedestrians are relatively less compared to those involving vehicles and riders. In most cases, the ego vehicle interacts with vehicles and riders, while pedestrians walk on the sidewalk. Therefore, in the BLVD dataset, the pedestrian trajectories are much fewer than trajectories of vehicles and riders. According to statistical results of BLVD, the ratio for trajectories of pedestrians, riders and vehicles collected in the BLVD dataset is 317:1114:3471 (1:3.51:10.94). Considering that the length of some trajectories for pedestrians and vehicles is too short and unbefitting in the training process, finally, 187, 1,039 and 2,222 trajectories of pedestrians, riders and vehicles are selected, respectively. Although only 187 trajectories of pedestrians are used for model training, the model performance can be guaranteed with a high accuracy of prediction and recognition in our test (as shown in Figs.8, 9 and 10 and in Tables IV, V and VII). We randomly sample 3103 trajectories (90%) for training and the rest (10%) for testing. The main parameters in the training process are detailed in Table III.

To evaluate the performance of the proposed hierarchical framework in different conditions, three groups with different length of observed frames and predicted frames (Observation→Prediction) are set in two experiments. The lengths of observations are set as 10, 20 and 30 frames (1s, 2s and 3s), while the lengths of the multi-step prediction are set as 5, 10, 15 and 20 frames (0.5s, 1s, 1.5s and 2s). All pairs of observation and prediction with limited horizon are randomly selected from the full length of trajectories.

C. Evaluation Metrics and Baselines

According to [26], the following metrics are selected to measure the performance and four baseline methods are chosen for comparison.

1) Metrics: In the IER module, the accuracy of the recognition for interactive events is calculated to measure the performance of our model:

$$Accuracy = \frac{\sum_{i=1}^{N} a_i}{N} \tag{22}$$

where a_t^i represents whether the recognition of interactive events for i^{th} instance is accurate at time t. $a_t^i = 0$ stands for a correct recognition and a_t^i=1 stands for a wrong one. N is the total number of samples. In the training process, N is the number of all training samples, while N is the number of test samples in the process of validation. Samples for training and testing are selected by cross validation (CV).

Following [26], in the TP module, the average displacement error (ADE) and the final displacement error (FDE) are chosen as evaluation metrics.

Average Displacement Errors (ADE): ADE is the average Euclidean distance errors between all predicted positions and ground truth positions, which is formulated as follow.

$$ADE = \frac{\sum_{i=1}^{N} \sum_{t_{\text{obs}}+1}^{t_{\text{obs}}+t_{\text{pred}}} \sqrt{(x_t^i - \hat{x}_t^i)^2 + (y_t^i - \hat{y}_t^i)^2}}{N * t_{\text{pred}}} \tag{23}$$

Final Displacement Errors (FDE): FDE is the mean Euclidean distance errors between final predicted positions and ground truth locations, which can be formulated as:

$$FDE = \frac{\sum_{i=1}^{N} \sqrt{(x_{t_{\text{pred}}}^i - \hat{x}_{t_{\text{pred}}}^i)^2 + (y_{t_{\text{pred}}}^i - \hat{y}_{t_{\text{pred}}}^i)^2}}{N} \tag{24}$$

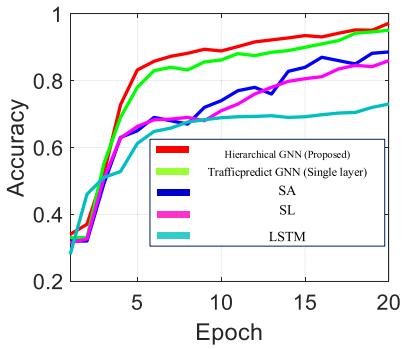

Fig. 5. The variation of recognition accuracy in the training process.

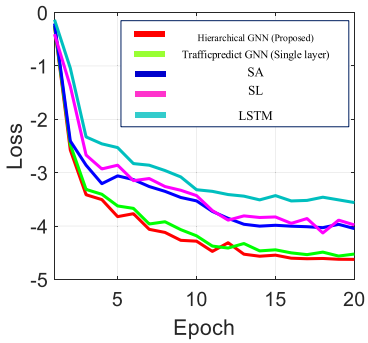

Fig. 7. The variation of loss in multi-step prediction of trajectories.

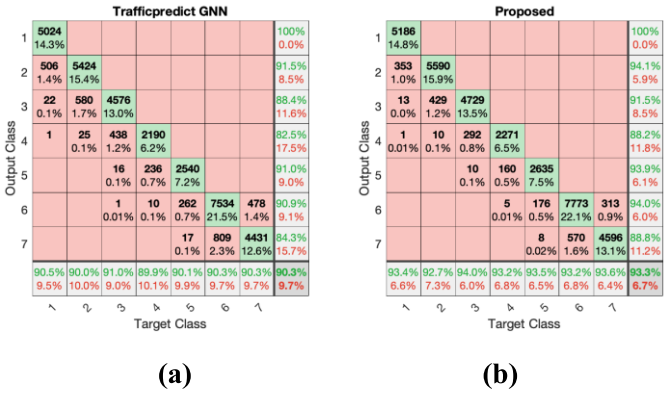

Fig. 6. Comparative results of riders for IER (a) The baseline method: *Trafficpredict* GNN; (b) The proposed method: Hierarchical GNN.

where (x_t^i, y_t^i) and $(\hat{x}_t^i, \hat{y}_t^i)$ are ground truth and predicted locations of the i^{th} observed instance at time t. N is the total number of samples.

2) Baseline Methods: In order to show the improvement of the recognition and the prediction, our framework is compared with these models below:

Social LSTM (SL): The SL proposes the social pooling method, which accounts for neighbouring crowds within certain regions [26].

Social Attention (SA): It is a model predicting trajectories for crowds, considering spatial relations of pedestrians with attention mechanism [46].

Basic LSTM: A LSTM with an input and an output embedding layer [47].

TrafficPredict GNN: A graph neural network considering different types of traffic participants [39].

D. The Performance of IER Module

The proposed framework in this paper is composed of two modules. The output of IER module influence the performance of framework. Before the framework is verified by the output of TP module, we firstly test the performance of the IER module in the recognition of interactive events.

1) Results: Fig.5 and Fig.7 present the variation of loss in the training process. The results in two figures can reflect the convergence of the model with the increase of epochs. If the value of loss increases slightly after several epochs, the model will be considered as a convergent condition. In this research, the number of epochs is set to 20 for the comparative study in the training process. After 20 epochs, the proposed framework reaches the condition of convergence and obtains the best results in comparative experiments. In other interactive scenarios or datasets, the number of training epochs for proposed framework depends on the variation of training loss and the requirement in the performance, which may differ from selected parameters in this experiment. Fig.5 presents the variation of recognition accuracy with epoch number increasing from 1 to 20 in the training process. At epoch 5, the accuracy of five methods are 83.2%, 78.1%, 65.3%, 62.8% and 61.2% for the proposed framework, GNN, SA, SL and basic LSTM, respectively. After 20 epochs, the proposed framework obtains the best performance and increases steadily, which reaches the condition of convergence. It indicates that the IER module of the proposed framework outperforms the baseline methods in the accuracy.

As for the application on intelligent vehicles, the framework can online generate results of the recognition and the prediction with one offline training process. In the testing process, the proposed framework can apply the trained parameters to generate predicted results and do not need to be trained for the next sample. The time consumption of the trajectory prediction in the process of the test and validation are detailed in experiments (Table VI and Table VII).

Table IV shows comparative results with the four baseline methods. The IER module of the proposed framework obtains the best performance with different lengths of input trajectories. The accuracy of SA and SL are slightly lower than that of IER module while basic LSTM gets the lowest accuracy in the comparative experiment. The difference between *Trafficpredict* GNN and the IER module in the proposed framework is the selection of the predicted length. For the recognition of interactive events, one-step prediction is realised by the X-to-1 LSTM in the proposed framework, while *Trafficpredict* GNN generates the sequential result by the multi-step prediction. The results in Table IV indicate that the one-step in the IER module of the proposed framework outperforms *Trafficpredict* GNN in the recognition of interactive events.

Confusion matrices for the proposed framework and *Trafficpredict* GNN are presented in Fig.6, which illustrates the detailed recognised results of riders. The result is generated from 35132 samples in the training set, including 7 different interactive events for riders (cyclists and motorbikes). The numbers of class 1∼7 stand for interactive events: riding away

TABLE IV
THE COMPARATIVE RESULTS FOR RECOGNITION OF INTERACTIVE EVENTS

Metric	Participant	Observation	LSTM	SL	SA	*Trafficpredict* GNN	**Hierarchical GNN (Proposed)**
Accuracy	Pedestrian	10	0.95	0.94	0.95	0.97	**0.99**
		20	0.66	0.89	0.91	0.96	**0.98**
		30	0.58	0.85	0.89	0.94	**0.98**
	Rider	10	0.82	0.93	0.92	0.96	**0.97**
		20	0.64	0.87	0.89	0.95	**0.98**
		30	0.53	0.81	0.83	0.95	**0.96**
	Vehicle	10	0.92	0.93	0.92	0.95	**0.97**
		20	0.62	0.84	0.86	0.93	**0.97**
		30	0.59	0.80	0.82	0.93	**0.94**
	Total	10	0.89	0.93	0.93	0.96	**0.98**
		20	0.63	0.86	0.88	0.96	**0.98**
		30	0.61	0.83	0.84	0.93	**0.96**
	Average		0.71	0.87	0.88	0.95	**0.97**

TABLE V
THE COMPARATIVE RESULTS FOR MULTI-STEP PREDICTION OF INTERACTIVE TRAJECTORIES

Metrics	Agents	LSTM	SL	SA	*Trafficpredict* GNN	**Hierarchical GNN (Proposed)**
ADE[m]	Pedestrian	0.27	0.14	0.13	0.12	**0.06**
	Rider	0.38	0.14	0.13	0.14	**0.09**
	Vehicle	0.38	0.14	0.12	**0.09**	**0.09**
	Total	0.48	0.16	0.15	0.12	**0.07**
	Average	0.38	0.15	0.13	0.12	**0.07**
FDE[m]	Pedestrian	0.39	0.25	0.24	0.21	**0.09**
	Rider	0.57	0.22	0.21	0.20	**0.13**
	Vehicle	0.56	0.18	0.17	0.18	**0.13**
	Total	0.61	0.24	0.22	0.20	**0.09**
	Average	0.44	0.22	0.21	0.20	**0.11**

and getting closer, riding up and getting closer, riding away and getting farther, crossing quickly from left, crossing quickly from right, central separation area of the road (lines or physical materials) and stopping. Interactive events provided by BLVD dataset cover most of interactions between riders and ego vehicle. As for class 1∼7, compared to *Trafficpredict* GNN, the proposed hierarchical framework improves the accuracy of the recognition by 2.9%, 2.7%, 3.0%, 3.3%, 3.4%, 2.9%, and 3.3%, respectively. The proposed framework obtains the highest improvement in the recognition of class 5 (crossing quickly from left). The main reason is that interactive event (class 5, crossing quickly from left) is a common and familiar manoeuvre in the urban traffic scenario and easier to be recognised by the X-to-1 LSTM. According to the comparative result, the proposed hierarchical GNN obtains a higher accuracy for each class compared to *Trafficpredict* GNN, which shows the best performance among the four baseline methods.

2) Analysis: As mentioned above, the proposed framework achieves the best performance in the recognition of interactive events. In baseline methods above, basic LSTM is a general neural network without social pooling and attention mechanism, which cannot consider the information of the surrounding traffic participants and influence of interaction. Although SL can encode interactions between the ego vehicle and surrounding vehicles by social pooling and attention mechanism, the relationship between traffic participants is not defined explicitly and different types of traffic participants are not considered individually. Considering the two issues above, in the X-to-1 GNN-based recognised model of the IER module, the instance layer and category layer are developed based on the graph structure. The explicit definition of nodes and edges and individual models of traffic participants with different types improves the performance of IER module.

E. The Performance of the Proposed Hierarchical Framework

After the comparative study and comprehensive analysis of the IER module, in this part the integral framework is verified in the multi-step prediction of trajectories. Similarly, in Fig.7, the variation of loss value is presented with the number of epochs increasing from 1 to 20. The lower loss value indicates the better performance in the training process, because the loss is valued by the negative log Likelihood. Compared to three baseline methods at epoch 5,10,15 and 20, the loss of the proposed hierarchical framework decreases rapidly and obtain the lowest loss value, which verifies the same conclusion in Fig.5. At epoch 20, the proposed framework reaches the convergence and loss values of other baseline methods are listed as follows: −4.64 (proposed framework), −4.55 (GNN), −4.09 (SA), −3.98 (SL) and −3.51 (basic LSTM). The variation and final results of loss values indicate that the proposed framework outperforms baseline methods in the convergent speed and accuracy. The variation of 20 epochs

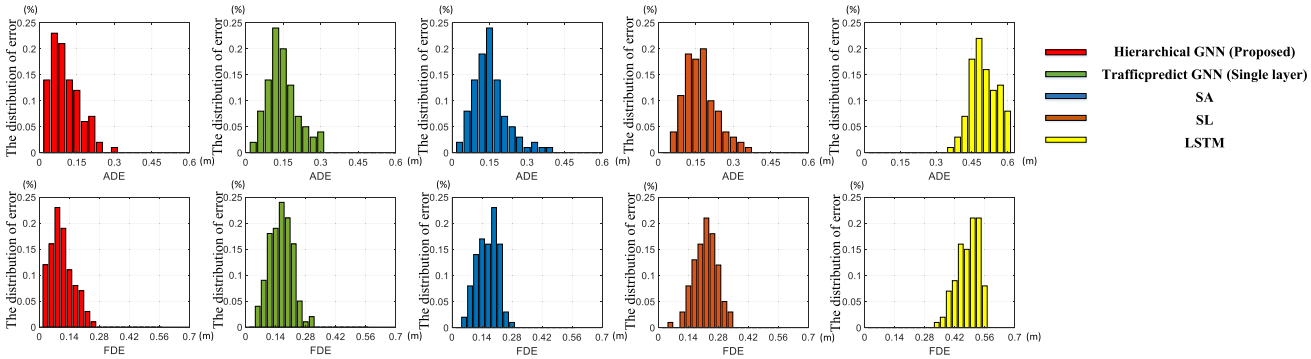

Fig. 8. Comparative results of error distributions. (Top: ADE; Bottom: FDE) The unit of the x-axis is meter (m) and the unit of the y-axis is percentage (%).

TABLE VI
COMPARATIVE RESULTS FOR TIME CONSUMPTION

Methods	Frames/per second	FDE (m)
SL	16.2	0.22
SA	17.3	0.21
Trafficpredict GNN	12.6	0.2
Hierarchical GNN (Proposed)	10.7	0.11

TABLE VII
COMPARATIVE RESULTS FOR THREE SELECTED SCENARIOS

ID	Methods	Recognition accuracy of IER	ADE (m)	FDE(m)	Average error in 1s
#1	LSTM	0.71	0.63	0.71	0.58
	SL	0.83	0.22	0.32	0.17
	SA	0.85	0.23	0.30	0.19
	TrafficPredict	0.89	0.19	0.17	0.14
	Proposed	**0.91**	**0.11**	**0.15**	**0.12**
#2	LSTM	0.68	0.58	0.77	0.59
	SL	0.87	0.24	0.41	0.18
	SA	0.79	0.18	0.35	0.20
	TrafficPredict	0.95	0.14	0.24	0.12
	Proposed	**0.96**	**0.11**	**0.18**	**0.10**
#3	LSTM	0.73	0.67	0.92	0.54
	SL	0.82	0.30	0.41	0.27
	SA	0.84	0.31	0.28	0.28
	TrafficPredict	0.96	0.22	0.21	0.15
	Proposed	**0.98**	**0.13**	**0.17**	**0.13**

presented in Fig.7 illustrates the convergence of the training process. Meanwhile, the time cost of the baseline methods in the trajectory prediction are detailed in experiments.

1) Results: The variation of loss in the multi-step prediction of trajectories is shown in Fig.7. Comparative results for multi-step prediction of trajectories are shown in Table V. The number of historical observations and the length of multi-step prediction are 30 frames (3s) and 10 frames (1s), respectively. In four baseline methods, the basic LSTM has the worst performance in ADE and FDE compared to other methods. The average error of SL is lower than basic LSTM by 60.5% and 50.0% for ADE and FDE, respectively. By combining the social pooling with basic LSTM, SL predicts trajectories according to the information of the surrounding participants. However, in SL, surrounding participants share the same weight in the prediction of interactive behaviours. By adding the attention mechanism in SL, SA improves the performance of SL by reducing 13.3% and 4.5% for ADE and FDE, respectively. Compared to above baseline methods, the proposed hierarchical framework obtains the best results in ADE and FDE for each type of traffic participants and reduces 41.7% (ADE) and 40.9% (FDE) compared to GNN without the hierarchical structure.

2) Analysis: According to the analysis and comparison presented above, the proposed framework outperforms other baseline methods in terms of ADE and FDE. In SL, although trajectories of participants and the influence between participants are considered in the prediction by conventional layers, interactions in spatial-temporal space and different types of traffic participants are not modelled specifically, which are solved by the instance layer and category layer in the proposed framework. In the heterogeneous urban scenario, traffic participants with different types have different motion patterns [39], in SL and SA, all traffic participants are modelled indiscriminately, which causes a decline in the accuracy of prediction. The proposed framework solves this problem by adding a category layer for each type of participant.

Compared to GNN including the category layer, which is an end-to-end prediction model, labelled interactive events for each surrounding traffic participant are considered in the IER module of the proposed framework. The hierarchical structure firstly recognises interactive events, and combines recognised results and historical trajectories as the input to TP module. Experimental results indicate that the proposed hierarchical framework outperforms *trafficpredict* GNN (with category layer). The proposed hierarchical framework combines advantages of manoeuvre-based and interaction-aware methods, which improves the performance in the prediction compared to *trafficpredict* GNN.

In order to verify the performance of time consumption for interactive aware methods, comparative results in the efficiency of multi-step prediction are detailed in Table VI. Results of time consumption for four social-aware methods remain in the same order of magnitude (10.7∼17.3 frames/per second). The comparison for error and efficacy of prediction is presented in Table VI. Compared to three interactive aware baseline methods, the proposed hierarchical GNN can obtain the best performance in the prediction with an acceptable time consumption.

Fig. 9. Comparative results for the prediction of trajectories are illustrated in camera-based corresponding 2D images. The prediction of trajectories in different scenarios are presented in nine pictures. The ground truth (GT) is drawn in green and the prediction results of baseline methods (GNN, LSTM, SL and SA) are shown with different lines. (a): Passing a zebra crossing with pedestrians and riders. (b) and (g): A rider and a vehicle pass each other side by side. (c): A rider travels across in front of ego vehicle. (d): A vehicle turns around in front of ego vehicle. (e): A pedestrian and a rider cross the road in parallel. (f): A rider overtakes a vehicle. (h): An intersection with three types of participants. (i) A pedestrian walks through the traffic.

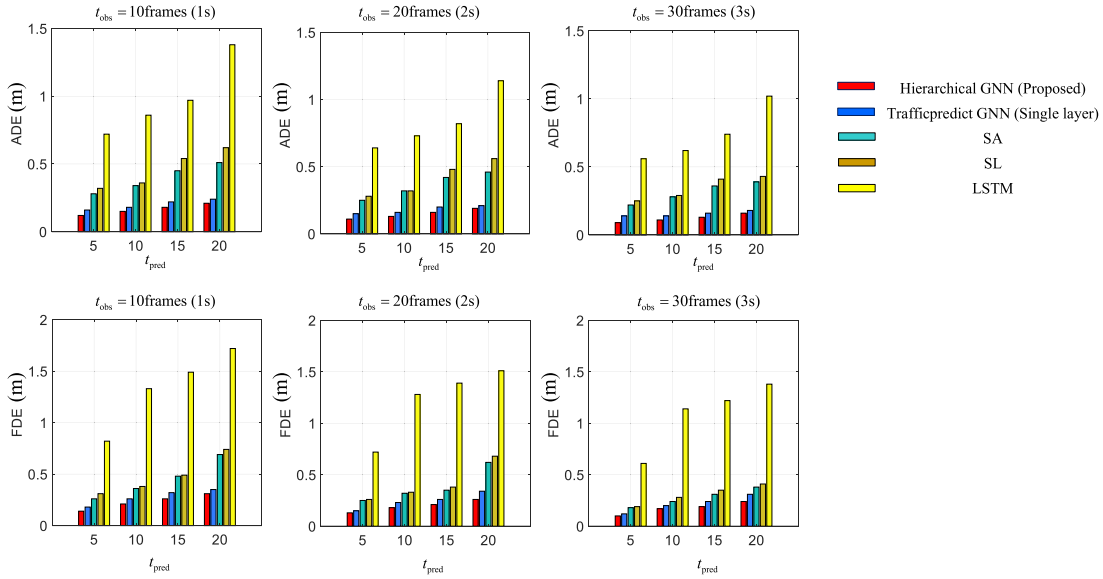

Fig. 10. Comparative results of ADE and FDE with different observed and predicted length. (Top: ADE; Bottom: FDE) The unit of ADE and FDE is meter (m).

Fig.8 shows distributions of ADE and FDE for different methods. The number of the historical observations and the length of multi-step prediction are 30 frames (3s) and 10 frames (1s), respectively. For ADE, the prediction error of the proposed framework mainly distributes in 0.03∼0.06 m, while the error of four baseline methods mainly distribute in 0.09∼0.12 m (GNN), 0.12∼0.15 m (SA), 0.12∼0.15 m (SL) and 0.45∼0.48 m (Basic LSTM), respectively. Comparative results in the error distribution illustrate that the proposed framework obtains the best performance.

The prediction of trajectories in different scenarios are presented in Fig.9. Nine pictures in Fig.9(a)-(i) show typical urban traffic scenarios containing traffic participants which are not strictly following rules. Although the front-facing camera cannot fully capture the whole scenario, the proposed framework performs reasonable prediction of trajectories and is close to ground truth.

To verify the performance of the proposed framework in different conditions, Fig.10 presents the results with different length of observation and multi-step prediction. The lengths of observations include 1s (10 frames), 2s (20 frames), 3s (30 frames), while the lengths of the multi-step prediction includes 0.5s (5 frames), 1s (10 frames), 1.5s (15 frames) and 2s (20 frames). The comparative results illustrate that the

variation of observation-prediction pair cannot influence the accuracy of ADE and FDE compared to baseline methods.

In order to test the performance of the proposed framework in different scenarios, an additional experiment is conducted in three typical driving scenarios. Comparative results are detailed in Table VII. The number of historical observations and the lengths of multi-step prediction are 30 frames (3s) and 20 frames (2s), respectively. The three selected scenarios are driving straight in highway (scenario #1), overtaking in highway (scenario #2) and driving through the intersection (scenario #3). Testing results include recognition accuracy of the IER module, ADE and FDE. Table VII shows that the proposed framework obtains the best performance compared to the four baseline methods. In the highway scenario (scenario #1 and scenario #2), the hierarchical GNN can get the highest accuracy for recognition and the lowest errors for the prediction. Even in the intersection (scenario #3), a typical challenging scenario for trajectory prediction, the proposed framework can outperform *Trafficpredict* GNN and other social-aware methods. Besides the comparative study of ADE and FDE, a detailed comparison for the predicted error in 1s is also presented in three typical scenarios. The error in 1s can describe the relative error to time, which can be applied to analyse the comparative results from a new point of view. The results indicate that the proposed framework can obtain the best performance in the fix predicted time (1s).

V. Conclusion and Future Work

In this paper, considering interactive behaviours between heterogeneous traffic participants, a hierarchical GNN-based framework is proposed to predict interactive events and trajectories of surrounding traffic participants with different types. The proposed framework combines the manoeuvre-based and interaction-aware methods by IER and TP modules to make multi-step prediction based on recognised events. A category layer is applied in the framework to learn similar characteristics for traffic participants with the same type.

The proposed framework is verified using BLVD dataset, which contains explicit labels for interactive events and continuous trajectories for traffic participants. In the IER module, the framework is applied to recognise interactive events with the ego vehicle by considering the interaction of all participants. In the TP module, the framework is tested by predicting multi-step trajectories for heterogeneous participants. Comparative results with the state-of-the-art methods show that it presents a higher accuracy in recognition and a lower ADE/FDE in the multi-step prediction. The improvement is due to modeling the interaction of traffic participants explicitly with the combination of interaction-aware and manoeuvre-based approaches. Meanwhile, the proposed hierarchical framework is a general method, which can be applied in different scenarios with interactive behaviours.

The research in this paper focuses on interactions with dynamic traffic participants. The influence of static obstacles and road information will be considered and modelled in the framework in the future work.

References

[1] S. Yoon, H. Jeon, and D. Kum, "Predictive cruise control using radial basis function network-based vehicle motion prediction and chance constrained model predictive control," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3832–3843, Oct. 2019.

[2] S. Ammoun and F. Nashashibi, "Real time trajectory prediction for collision risk estimation between vehicles," in *Proc. IEEE 5th Int. Conf. Intell. Comput. Commun. Process.*, Aug. 2009, pp. 417–422.

[3] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH J.*, vol. 1, no. 1, pp. 1–14, Dec. 2014.

[4] P. Wu, S. Chen, and D. N. Metaxas, "MotionNet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11385–11395.

[5] A. Eidehall and L. Petersson, "Statistical threat assessment for general road scenes using Monte Carlo sampling," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 137–147, Mar. 2008.

[6] J. Hillenbrand, A. M. Spieker, and K. Kroschel, "A multilevel collision mitigation approach-its situation assessment, decision making, and performance tradeoffs," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 528–540, Dec. 2006.

[7] N. Deo, A. Rangesh, and M. M. Trivedi, "How would surround vehicles move? A unified framework for maneuver classification and motion prediction," *IEEE Trans. Intell. Vehicles*, vol. 3, no. 2, pp. 129–140, Jun. 2018.

[8] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Behavior classification algorithms at intersections and validation using naturalistic data," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 601–606.

[9] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Driver behavior classification at intersections and validation on large naturalistic data set," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 724–736, Jun. 2012.

[10] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 797–802.

[11] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2287–2301, Nov. 2011.

[12] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 334–341.

[13] J. Schlechtriemen, F. Wirthmueller, A. Wedel, G. Breuel, and K.-D. Kuhnert, "When will it change the lane? A probabilistic regression approach for rarely occurring events," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2015, pp. 1373–1379.

[14] J. Li, C. Lu, Y. Xu, Z. Zhang, J. Gong, and H. Di, "Manifold learning for lane-changing behavior recognition in urban traffic," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3663–3668.

[15] C. Lu, F. Hu, D. Cao, J. Gong, Y. Xing, and Z. Li, "Virtual-to-real knowledge transfer for driving behavior recognition: Framework and a case study," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6391–6402, Jul. 2019.

[16] C. Lu, F. Hu, D. Cao, J. Gong, Y. Xing, and Z. Li, "Transfer learning for driver model adaptation in lane-changing scenarios using manifold alignment," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3281–3293, Aug. 2020.

[17] Z. Li *et al.*, "Transferable driver behavior learning via distribution adaption in the lane change scenario," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 193–200.

[18] C. Gong, Z. Li, C. Lu, J. Gong, and F. Hu, "A comparative study on transferable driver behavior learning methods in the lane-changing scenario," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3999–4005.

[19] Z. Li, J. Gong, C. Lu, and J. Xi, "Importance weighted Gaussian process regression for transferable driver behaviour learning in the lane change scenario," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12497–12509, Nov. 2020.

[20] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1179–1184.

[21] G. Aoude, J. Joseph, N. Roy, and J. How, "Mobile agent trajectory prediction using Bayesian nonparametric reachability trees," in *Proc. AIAA Infotech Aerosp.*, 2011, p. 1512.

[22] C. Laugier *et al.*, "Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety," *IEEE Intell. Transp. Syst. Mag.*, vol. 3, no. 4, pp. 4–19, Oct. 2011.
[23] J. Wiest, M. Hoffken, U. Kresel, and K. Dietmayer, "Probabilistic trajectory prediction with Gaussian mixture models," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 141–146.
[24] J. Li, W. Zhan, Y. Hu, and M. Tomizuka, "Generic tracking and probabilistic prediction framework and its application in autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3634–3649, Sep. 2020.
[25] L. Hou, L. Xin, S. E. Li, B. Cheng, and W. Wang, "Interactive trajectory prediction of surrounding road users for autonomous driving using structural-LSTM network," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4615–4625, Nov. 2020.
[26] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 961–971.
[27] J. Sun, Q. Jiang, and C. Lu, "Recursive social behavior graph for trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 660–669.
[28] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, and B. Dariush, "Egocentric vision-based future vehicle localization for intelligent driving assistance systems," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9711–9717.
[29] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14424–14432.
[30] Z. Li, J. Gong, C. Lu, and Y. Yi, "Interactive behaviour prediction for heterogeneous traffic participants in the urban road: A graph neural network-based multi-task learning framework," *IEEE/ASME Trans. Mechatronics*, early access, Apr. 16, 2021, doi: 10.1109/TMECH.2021.3073736.
[31] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2255–2264.
[32] C. Li, Y. Meng, S. H. Chan, and Y.-T. Chen, "Learning 3D-aware egocentric spatial-temporal interaction via graph convolutional networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 8418–8424.
[33] S. Dai, L. Li, and Z. Li, "Modeling vehicle interactions via modified LSTM models for trajectory prediction," *IEEE Access*, vol. 7, pp. 38287–38296, 2019.
[34] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5308–5317.
[35] B. Du, X. Hu, L. Sun, J. Liu, Y. Qiao, and W. Lv, "Traffic demand prediction based on dynamic transition convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 1237–1247, Feb. 2021.
[36] M. Lv, Z. Hong, L. Chen, T. Chen, T. Zhu, and S. Ji, "Temporal multi-graph convolutional network for traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3337–3348, Jun. 2021.
[37] B. Yu, Y. Lee, and K. Sohn, "Forecasting road traffic speeds by considering area-wide spatio-temporal dependencies based on a graph convolutional neural network (GCN)," *Transp. Res. C, Emerg. Technol.*, vol. 114, pp. 189–204, May 2020.
[38] L. Zhao *et al.*, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.
[39] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "TrafficPredict: Trajectory prediction for heterogeneous traffic-agents," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 6120–6127.
[40] J. Xue *et al.*, "BLVD: Building a large-scale 5D semantics benchmark for autonomous driving," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6685–6691.
[41] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*. [Online]. Available: http://arxiv.org/abs/1710.10903
[42] J. Shore and R. Johnson, "Properties of cross-entropy minimization," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 4, pp. 472–482, Jul. 1981.
[43] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1468–1476.
[44] J. Li, W. Zhan, and M. Tomizuka, "Generic vehicle tracking framework capable of handling occlusions based on modified mixture particle filter," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 936–942.
[45] J. Li, H. Ma, W. Zhan, and M. Tomizuka, "Generic probabilistic interactive situation recognition and prediction: From virtual to real," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 3218–3224.
[46] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1–7.
[47] F. Altche and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 353–359.

Zirui Li received the B.S. degree from the Beijing Institute of Technology (BIT), Beijing, China, in 2019, where he is currently pursuing the Ph.D. degree in mechanical engineering. He is also a Visiting Researcher with the Delft University of Technology (TU Delft). His research interests include intelligent vehicles, driver behavior modeling, and transfer learning.

Chao Lu (Member, IEEE) received the B.S. degree in transport engineering from the Beijing Institute of Technology (BIT), Beijing, China, in 2009, and the Ph.D. degree in transport studies from the University of Leeds, Leeds, U.K., in 2015. In 2017, he was a Visiting Researcher with the Advanced Vehicle Engineering Centre, Cranfield University, Cranfield, U.K. He is currently a Lecturer with the School of Mechanical Engineering, BIT. His research interests include intelligent transportation and vehicular systems, driver behavior modeling, reinforcement learning, and transfer learning and its applications.

Yangtian Yi received the B.E. degree from the Beijing Institute of Technology (BIT), Beijing, China, in 2020, where he is currently pursuing the M.A.Sc. degree in mechanical engineering. He is also studying the application of graph neural network in the field of intelligent vehicles.

Jianwei Gong (Member, IEEE) received the B.S. degree from the National University of Défense Technology, Changsha, China, in 1992, and the Ph.D. degree from the Beijing Institute of Technology (BIT), Beijing, China, in 2002. From 2011 to 2012, he was a Visiting Scientist with the Robotic Mobility Group, Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently a Professor and the Director of the Intelligent Vehicle Research Centre, School of Mechanical Engineering, BIT. His research interests include intelligent vehicle environment perception and understanding, decision making, path/motion planning, and control.