HSTA: A Hierarchical Spatio-Temporal Attention Model for Trajectory Prediction

Ya Wu, Guang Chen, *Member, IEEE*, Zhijun Li, *Senior Member, IEEE*, Lijun Zhang, Lu Xiong, Zhengfa Liu, and Alois Knoll, *Senior Member, IEEE*

***Abstract*—Predicting the future trajectories of surrounding agents has become an crucial problem to be solved for the safety of autonomous vehicles. Recent studies based on Long Short Term Memory (LSTM) networks have shown powerful abilities to model social interactions. However, many of these approaches focus on spatial interactions of the neighborhood agents but ignore temporal interactions that accompany spatial interactions. In this paper, we propose a Hierarchical Spatio-Temporal Attention architecture (HSTA), which activates the utilization of spatial interactions with different weights, and jointly considers the temporal interactions across time steps of all agents. More specially, the graph attention mechanism (GAT) is presented to capture spatial interactions, the multi-head attention mechanism (MHA) is conducted to encode temporal correlations of interactions and a state gated fusion (SGF) layer is used to integrate spatial and temporal interactions. We evaluate our proposed method against baselines on both pedestrian and vehicle datasets. The results show that our model is effective and achieves state-of-the-art achievements.**

***Index Terms*—Trajectory prediction, autonomous driving, spatio-temporal modeling.**

I. Introduction

AS THE central problem of fully autonomous driving vehicles (AVs), anticipating possible future trajectories of surrounding vehicles has received considerable critical attention. As a bridge between perception and decision making blocks, it promotes AVs to better understand their surroundings and make safe as well as efficient decisions [1]. Traditionally, early works use a hand-crafted, explicit dynamical model to generate future trajectories with the position, velocity, and acceleration of target agents, such as the constant velocity model (CV), the constant acceleration model (CA) and Kalman filter-based (KF) [2]. However, human drivers infer surrounding vehicles' trajectories not only with their past states, such as positions, directions, and speeds, but also interactions among vehicles and traffic environments. Recently investigators have examined the effects of interactions on trajectories anticipating, and a large number of studies have been introduced [3], [4].

This observation has led the community to turn to RNN-based methods, which have shown great success in processing temporal dependencies between the input sequence elements. Therefore, several studies [5]–[7] use an LSTM-based encoder decoder architecture to model temporal sequences, which can be called agent-centric approaches. While LSTM-based methods have the ability to capture temporal dependencies, they lack the spatial interactions between surrounding agents. Then, [8]–[11] introduce a novel pooling mechanism that couples the LSTMs corresponding to neighboring agents to capture spatial dependencies, which can be called spatial-centric approaches. Due to the pooling scheme is restricted by spatial proximity, [12]–[14] propose an attention mechanism to assign unequal importance of neighboring agents. Such approaches, however, have failed to tackle spatio-temporal interactions and complex temporal dependencies. For spatial interaction, it is not only related to the Euclidean distance between agents, but also the topological relationship between agents within the whole scene has to be considered. For temporal dependencies, each frame in the observed trajectory plays a different role in predicting the future trajectory, and the model needs to assign attention to each frame rather than processing the sequence frames step by step.

According to previous work, we propose HSTA to solve their limitations, which is a Hierarchical Spatio-Temporal Attention network consisting of three scales to capture spatial and temporal interactions for trajectories prediction. In spatial attention layer, instead of the pure attention mechanism [12], [14], we adopt the Graph Attention networks (GAT) [15] which is more suitable to capture spatial interactions among agents at each time step. Specifically, we model all vehicles in the scene (Fig. 1), the nodes represent vehicles, and edges between two nodes denote the spatial relationships. In temporal attention layer, in contrast to LSTM-based in [16], we propose a multi-head attention (MHA) mechanism to deal with complex temporal dependencies, which could parallel the computation for all agents. Finally, we use a gate fusion layer to adaptive control influence of spatial and

Manuscript received March 25, 2021; revised August 5, 2021; accepted September 16, 2021. Date of publication September 27, 2021; date of current version November 18, 2021. This was was supported in part by the Shanghai Rising Star Program under Grant 21QC1400900, in part by Anhui Provincial Natural Science Foundation, Anhui Energy-Internet Joint Program under Grant 2008085UD01, in part by the National Natural Science Foundation of China under Grant 61906138, in part by the National Natural Science Foundation of China under Grant U1913601, and in part by the European Union's Horizon 2020 Framework Program for Research and Innovation under the Specific under Grant Agreement 945539 (Human Brain Project SGA3). The review of this article was coordinated by Prof. Jun Won Choi. *(Corresponding author: Guang Chen.)*

Ya Wu, Guang Chen, Lijun Zhang, Lu Xiong, and Zhengfa Liu are with the Department of Automotive Engineering, Tongji University, Shanghai 200092, China (e-mail: wuya@tongji.edu.cn; guangchen@tongji.edu.cn; tjedu_zhanglijun@tongji.edu.cn; xiong_lu@tongji.edu.cn; 1811466@tongji.edu.cn).

Zhijun Li is with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, and Department of Automation, University of Science and Technology of China, Hefei 230052, China (e-mail: zjli@ieee.org).

Alois Knoll is with the Chair of Robotics, Artificial Intelligence and Realtime Systems, Technical University of Munich, Munich 80333, Germany (e-mail: knoll@in.tum.de).

Digital Object Identifier 10.1109/TVT.2021.3115018

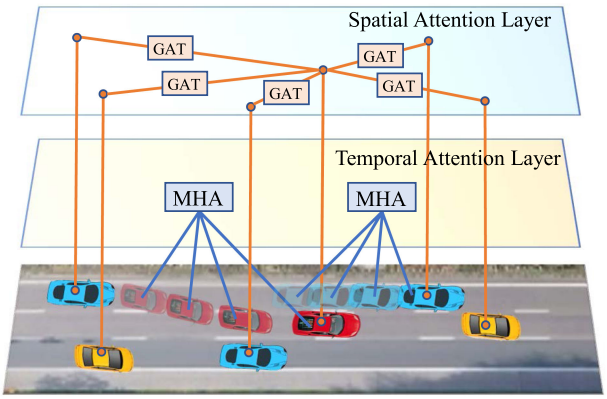


Fig. 1. HSTA is a spatio-temporal attention scheme. To execute this, our approach integrates the spatial dependencies of whole neighborhood vehicles in spatial attention layer, and the temporal dependencies of object agent in temporal attention layer.

temporal features at each time step. And the future trajectories are inferred by an LSTM-based encoder decoder scheme.

This paper is devoted to interactions among vehicles and pedestrians. Quantitative and qualitative experiments are conducted to show the contribution of model, results of quantitative experiments indicate that the accuracy of the proposed method is superior than state-of-the-art methods in complex interactive scenes. On the other hand, the qualitative experiments indicate that our model has the ability to learn different motion behaviours. The primary of our work was listed on the below:

- A novel spatio-temporal attention architecture HSTA that can be trained in an end-to-end fashion is proposed, which uses GAT to model the topological relationship between agents in the whole scene and captures temporal dependencies with MHA. We believe that the learned spatial and temporal attention can help the model to obtain accurate predictions in crowded scenarios.
- Our model is capable of modeling spatio-temporal interactions simultaneously, which bridges the gap between agent-centric and spatial-centric trajectory prediction paradigms.
- We conduct experiments on two different trajectory prediction task, pedestrian and vehicle datasets, and achieve state-of-the-art results for both. This shows that our model can improve generalization by reducing the variance of the predicted trajectory distributions, with the ability to learn different agent types and motion behaviours.

II. Related Work

The problem of trajectory prediction has been studied from different perspectives. Therefore, there are a large number of trajectory prediction models in the literature. Here, we mainly focus on modeling interactions and draw on two main lines of trajectories prediction research: spatial interactions with neighbor agents and temporal interactions in history trajectories.

A. Spatial Interactions With Neighbor Agents

The common pattern in trajectory prediction is basically focused on spatial interactions modeling. The pioneering model can be tracked back to the Social Force model [17], which superimposes attractive forces from a goal vehicle with repulsive forces from other vehicles. Then, the hand-crafted approaches have been applied, such as continuum dynamics [18], Discrete Choice framework [19], Gaussian processes [20] and Bayesian model [21]. It has been demonstrated that they are insufficient to capture interactions in complex scenarios. Recently, data-driven based models have shown enormous potential in spatial interactions modeling. Social LSTM [8] introduces a social pooling layer to model the interactions of people in a neighborhood. [9] proposes a social GAN model, the correlation of all the agents were taken into consideration by training adversarial against a recurrent discriminator in a particularly scene.

Conversely, [12], [22], [23] propose a novel attention-based scheme. It focuses on the agents, which has highest correlation between neighbor agents and targeted agents, but they ignore the inherent topology between vehicles. Therefore, inspired by graph convolutional networks (GCN) [24], [16], [25] adopt GAT to capture spatial interactions which assign different weights to graph nodes by the LSTM hidden states, and achieve superior performance on public human trajectory forecasting benchmarks. In our case, we also borrow ideas from GAT to model spatial interactions in spatial attention layer, and it is very convenient to be a plug and play block. Compared to prior work, we learn spatial features from their ground-truth input embedding. In order to maintain stability in training, we do a Layer Normalization (LN) on it. Our experiments suggest that these components speed up and stabilize training processing and improve the quality and generalizability of trained models.

B. Temporal Interactions in History Trajectories

Compared with spatial interactions, most of temporal interactions methods [8]–[10], [26] adopt Recurrent Neural Networks (RNN) and its variant that are tailored for temporal sequences, such as LSTM [27] and Gated Recurrent Units (GRU) [28]. Since [29], recent works [30]–[32] adopt convolutional neural networks (CNN) which could alleviate the gradient vanishing and exploding made by RNN, and they support increased parallelism and effective temporal representation. In this work, we incorporate the multi-head attention (MHA) mechanism [33] which is widely used in Natural Language Processing (NLP) research. It can also be executed in parallel and have lighter structures. Therefore, our models use the attention mechanism instead of sequential processing model and could more efficiently and robustly capture the temporal dependencies.

C. Spatio-Temporal Interactions for Trajectory Prediction

The spatio-temporal scheme which is based on GNNs has become more and more popular in trajectory prediction, after their strong performance in action recognition [34], [35], visual-spatiability tasks [36], robotic manipulation [37] and traffic prediction [38]. In the field of trajectory prediction, [31] directly extracts spatio-temporal features from the graph representing $G = (V, A)$ by the Spatio-Temporal Graph Convolution Neural Network (ST-GCNN) [39]. [32] proposes a Spatio-Temporal Graph interaction framework, which adopts temporal CNNs

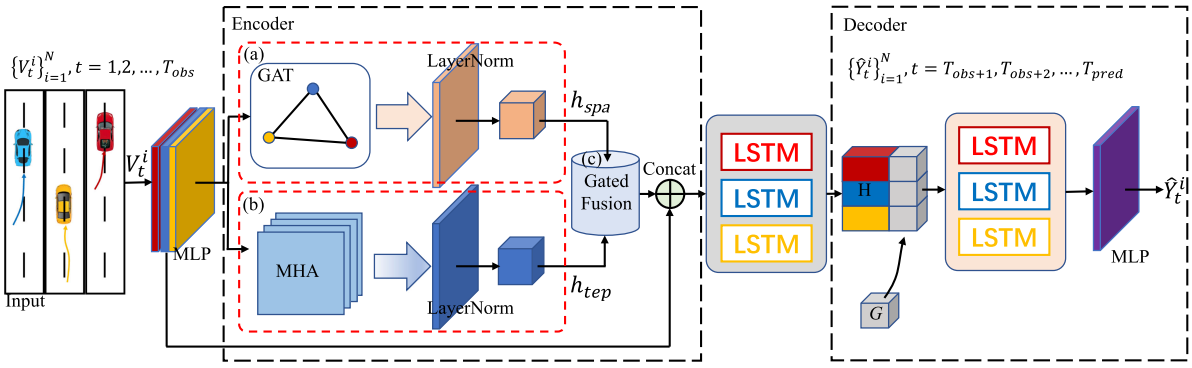


Fig. 2. Architecture for our proposed HSTA model. The encoder was mainly consist of three layers: (a) The spatial attention layer with GAT for spatial interactions; (b) The temporal attention layer with MHA for temporal interactions; (c) The state fusion layer, which concatenates the above two layers. The generated trajectory is mainly through the encoder-decoder framework based on LSTM.

to capture temporal dependencies and GAT to model spatial interaction. [16] builds a novel STAGT, the interactions between temporal correlations were measured by an extra LSTM. [40] presents STAR, a Spatio-Temporal grAph tRansformer framework, which uses the transformer to capture spatial and temporal interactions respectively. However, these methods still have difficulties in capturing temporal dependencies of long sequences, and the computational efficiency is not ideal. Our method combines spatial graph attention networks and temporal multi-head attention networks, which considers both spatio-temporal features and computational efficiency. Moreover, it can adaptively control influence of spatial and temporal features at each time step with a state gate fusion layer.

III. Hierarchical Spatio-Temporal Attention Model

To get insight of the spatial interactions and complex temporal dependencies in trajectory prediction, we develop a hierarchical spatio-temporal attention model to forecast the trajectories of agents.

Fig. 2 shows the structure of our model, HSTA is an encoder-decoder architecture. Before entering into the encoder, the historical features V_t is transformed to ΔV_t, then embedding into a higher dimension vector $X_{emb}(t)$ with a multilayer perception (MLP). The encoder is consist of three layers: a temporal attention layer with a Multi-Head Attention (MHA) module for embedding temporal interactions (Section III-B), a spatial attention layer with a Graph Attention Network (GAT) for capturing spatial interactions (Section III-C), and a gate fusion attention mechanism layer to fusion temporal features h_{tep} and spatial features h_{spa} at each time step as the input for the LSTM-based module (Section III-D). In the decoder module, we use LSTMs to generate diverse feasible trajectories in future timesteps (Section III-E). More details of our framework will be elaborated in the following sections.

A. Problem Definition

We assume that a scene at timestep t contains N agents, represented by a set $\{V_t^i\}_{i=1}^N$, where $V_t^i = (x_t^i, y_t^i)$ denotes absolute coordinates of the i-th agent V^i at timestep t. Then the set V_t^i of agent $i = 1, 2, \ldots, N$ at time steps $t = 1, 2, \ldots, T_{obs}$ as the input of model, the problem evolves into forecasting the future trajectories $(\hat{x}_t^i, \hat{y}_t^i)$, where $t = T_{obs+1}, T_{obs+2}, \ldots, T_{pred}$.

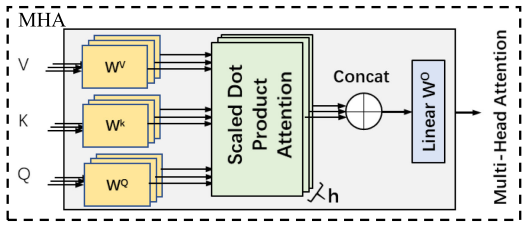


Fig. 3. Temporal attention layer with multi-head attention. The values V and keys K are concatenated from history time steps, and the query Q is produced by the current time step. It allows HSTA to learn on which position in the past is most important for predicting the future trajectories.

B. Temporal Attention Layer Modeling

The temporal attention layer is used to extract the temporal interactions in each agent independently. Inspired by [33], we find that the multi-head attention is similar to CNN and does not depend on the calculation of previous moments. Unlike the RNN variants that calculate hidden states step by step, the module can be calculated in parallel to capture sequence non-linearities. In addition, it can also capture long-distance dependencies. Therefore, we propose to use a multi-head attention to model temporal dependencies as illustrated in Fig. 3.

$$X_{emb}(t) = MLP_{emb}\left(\Delta V_t, W_{emb}\right) \tag{1}$$

where the ΔV_t denotes relative features. According to [9], [16], [25], we embed the agent's relative dynamic features ΔV_t into a high-dimensional vector $X_{emb}(t)$ through a multilayer perception (MLP), W_{emb} is the embedding weight ((1)). Then using the input embedding $X_{emb}(t)$ and three parameter matrices W_Q, W_K, W_V to generate the queries $Q = X_{emb}(t)W_Q$, keys $K = X_{emb}(t)W_K$ and values $V = X_{emb}(t)W_V$, and the parameter matrices $W_Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_K \in \mathbb{R}^{d_{model} \times d_k}$, $W_V \in \mathbb{R}^{d_{model} \times d_v}$, where Q, V represent current frame, K represents all frames in the observed trajectory, then the attention weight of all frames to the current frame is obtained by a scaled dot product. The output is then used to weight the V, which represents the specific value of each frame. The final attention

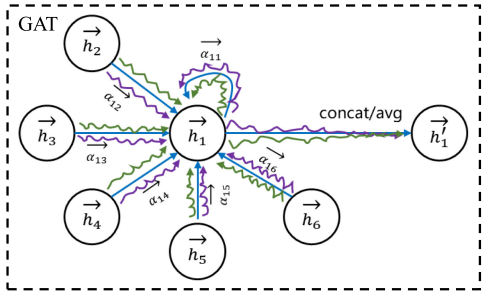


Fig. 4. Spatial attention layer with graph attention network, $\vec{h_i}$ is the feature vector of i-th agent. It allows assigning different weights to neighborhood agents.

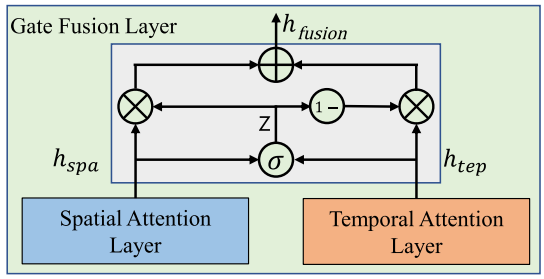


Fig. 5. State gate fusion layer with fusion gate, it combines spatial and temporal interactions via gated fusion.

matrix can be written as (2) that divides each by $\sqrt{d_k}$ to scale the dot product attention and use a softmax function to normalize it.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2}$$

Instead of performing a single scaled dot product attention block with d_{model}-dimensional queries, keys and values, we divide d_{model} into h parts, and $d_k = d_v = d_{model}/h$. Then we execute the attention block in parallel for h times, these independent attention outputs are concatenated and dynamically weighted to capture different feature representation by $W^o \in \mathbb{R}^{hd_v \times d_{model}}$. Then the obtained results h_m is processed by layer normalization to get h_{tep}

$$\begin{aligned} h_m &= M(Q, K, V) = Concat\left(head_1, \ldots, head_h\right) W^o \\ where \quad head_h &= Attention\left(Q_h, K_h, V_h\right) \end{aligned} \tag{3}$$

In this work, we utilize $h = 8$ parallel heads and $d_{model} = 32$. Although the dimension of each attention block has decreased, the overall computational cost is similar to the single scaled dot production attention with d_{model} dimension.

C. Spatial Attention Layer Modeling

Acquiring the complex spatial interaction is a key problem in trajectory prediction. The early LSTM-based method used Euclidean distance to describe the interactions between agents, which sometimes ignores agents that really need attentions. Because the topological mechanism between agents is similar to the graph structure, and prior works [16], [25] have shown that graph attention network (GAT) [15] has been successfully used in capturing the spatial relationships among surrounding vehicles, we adopt GAT as a part of our spatial attention layer.

As shown in Fig 4, we use an undirected graph $G = \{H, E\}$ to represent spatial interactions among agents. Node set H consists of each agent in the scene at each time-step, which is defined as $H = \{\vec{h}_t^i \mid i = 1, \ldots, N, t = 1, \ldots, t_{obs}\}$, where N is the number of observed agents in a scene, and t_{obs} is the history time steps. The $\vec{h}_t^i$ is the feature vector of i-th agent, it is the embedding output of $\{\Delta V_t^i\}_{i=1}^N$. In this paper, we define G is a complete graph, and the edge set $E = \{\vec{h}_t^i \vec{h}_t^j \mid (i, j \in \mathbb{N}_i)\}$, where $\mathbb{N}_i$ indicates the neighbor set of node i.

According to the graph-structured data obtained above, we extract spatial features by the GAT, which is composed of stacking graph attention layers. The input is the feature vector set $H_t = \{\vec{h}_t^1, \vec{h}_t^2, \ldots, \vec{h}_t^N\}, \vec{h}_t^i \in \mathbb{R}^F$ at time step t, which represents the feature of all nodes, and the dimension of H_t is $N \times F$. Then we use a shared attention mechanism a to extract feature, and first calculate the attention coefficients α_{ij} of each node. In order to better calculate and compare, we perform softmax on the α_{ij} and calculate the output features $\vec{h_i}'$ of every node. Finally, to stabilize the learning processing of self-attention, we execute K independent attention mechanisms to obtain the output of multi-head attention. In our implementation, spatial attention is obtained by two graph attention layers, and to reduce the effect of covariate shift, layer normalization is used to process the output $\vec{h}_i'$ of spatial attention layer. The output of the graph attention layer is a new feature set $H_t{}' = \{\vec{h'}_t^1, \vec{h'}_t^2, \ldots, \vec{h'}_t^N\}, \vec{h'}_t^i \in \mathbb{R}^{F'}$, F' and F can be unequal, and is processed by layer normalization. Then the results of layer normalization h_{spa} will be concatenated with embedding of input $X_{emb}(t)$ and temporal attention h_{tep}.

D. State Gate Fusion Layer

To fuse spatial and temporal features, we proposed a gate fusion layer to adaptively control the influence of spatial and temporal attention at each time step. As shown in Fig. 5, for hidden features h_{fusion}, including spatial features h_{spa} and temporal features h_{tep} by fusion gate z.

$$h_{fusion} = z \odot h_{spa} + (1 - z) \odot h_{tep} \tag{4}$$

with

$$z = \sigma\left(h_{spa} W_{z,1} + h_{tep} W_{z,2} + b_z\right) \tag{5}$$

where $W_{z,1}$, $W_{z,2}$ and b_z are learnable parameters, $\odot$ represents the element-wise product, $\sigma(\cdot)$ denotes the sigmoid activation, z is the fusion gate which is computed by (5). Then we concatenate the fusion features h_{fusion} and input embedding $X_{emb}(t)$, and new features H_{fusion} as the input of E_LSTM (LSTM for encoder).

$$\begin{aligned} H_{fusion} &= X_{emb}(t) \parallel h_{fusion} \\ e_t &= E\_LSTM\left(e_{t-1}, H_{fusion}, W_e\right) \end{aligned} \tag{6}$$

where $X_{emb}(t)$ is from Eq. (1), W_e is the weight of E_LSTM, $\parallel$ denotes the concatenate operation. To stabilize the training

process and prevent the gradient from vanishing or exploding, we add a LSTM layer between the encoder and the decoder. It converts the encoded spatio-temporal features to generate hidden representations as the input of the decoder.

E. Future Trajectory Prediction

By the above spatio-temporal attention mechanism, we get a hidden vector $e_{t_{obs}}$. Considering the uncertainty of the future trajectory, a random Gaussian noise G is added to the hidden vector. Then the relative position of the next time step is obtained by the D_LSTM (LSTM for decoder) and a multi-layer perceptron $\sigma(\cdot)$.

$$E_{t_{obs}} = e_{t_{obs}} \parallel G \tag{7}$$

$$\Delta\hat{Y}_{t_{obs+1}} = \sigma\left(D\_LSTM(d_{t_{obs}}, E_{t_{obs}}, W_d)\right) \tag{8}$$

Following the strategy to compute variety loss in [9], we select the smallest distance between predicted relative positions $\Delta\hat{Y}_t$ and ground-truth relative positions ΔY_t to calculate loss. Eq9 and Eq.10 is our loss function for training, which calculate L2 loss L_h for k possible future trajectories, then the variety loss $L_{variety}$ is obtained by the minimum L_h.

$$L_h\left(\Delta\hat{Y}\right) = \frac{\sum_{i=1}^{N}\sum_{t=t_{obs+1}}^{T_{pred}}\left\|\Delta\hat{Y}_t^i - \Delta Y_t^i\right\|_2^2}{NT_{pred}} \tag{9}$$

$$L_{variety} = \min_k\left(L_h\Delta\hat{Y}^{(k)}\right) \tag{10}$$

IV. Experimental Results

We evaluate our model, which we call HSTA, on two different kinds of public datasets that include pedestrian datasets and vehicle datasets. Quantitative and qualitative results are also presented for further analysis. We have also conducted ablation studies to understand the efforts of each proposed module and hope to contribute to the structural design of model in the trajectory forecasting.

As a brief summary, we show that: 1) HSTA has better performance than the attention-based SOTA model on pedestrian and vehicle datasets; 2) the temporal attention improves the accuracy of long-term prediction compared to attention-based methods; 3) the hierarchical spatio-temporal attention model improves sophisticated interaction modeling and model reasoning is faster.

A. Datasets and Metrics

The pedestrian datasets include ETH [41] and UCY [42], which consist of five sets, four scenarios named ETH, HOTEL, UCY, ZARA-01 and ZARA-02. They contain thousands of pedestrian trajectories with rich interactive behaviors, and the recording frequency is 2.5 Hz ($\Delta t = 0.4$ s). We adopt an evaluation strategy similar to [9], which is called the leave-one-out.

Considering that these datasets mentioned above only contain pedestrian trajectories, and we also evaluate our experiments on two publicly available vehicle datasets: NGSIM [43], [44], and highD [45]. The NGSIM includes US-101 and I-80 dataset, which is recorded on real freeway traffic by multiple overhead cameras in the US in 2005, each dataset contains 45 minutes of vehicle data at 10 Hz. HighD is a real-world vehicle dataset recorded by a camera-equipped drone on German highways in 2017 and 2018. It includes more than 110 500 vehicles, each vehicle's trajectory, including vehicle position, velocity, acceleration, type, and size. Following the similar methodology as [10], which the datasets are divided into 70% training, 10% validation, and 20% testing.

TABLE I
Effects of Data Augmentations, Basic Represents No Operation and Absolute Coordinates, Rel Represents Relative Coordinates, Rot+rel Represent Relative Coordinates and Random Rotations

Model	Metric	Basic	Rel	Rot+Rel
V-LSTM	ADE Hotel	6.50	0.64	**0.55**
	FDE Hotel	9.24	1.28	**1.12**
	ADE Avg	2.64	0.61	**0.50**
	FDE Avg	3.85	1.24	**1.00**
HSTA	ADE Hotel	4.70	0.63	**0.44**
	FDE Hotel	5.19	1.23	**0.84**
	ADE Avg	3.90	0.55	**0.42**
	FDE Avg	4.60	1.05	**0.82**

Evaluation Metric: Following reporting conventions [4], [9], [10], [12], we report our results in two error metrics:

1) Average Displacement Error (ADE): the mean square error (MSE) over all prediction trajectories and ground-truth.
2) Final Displacement Error (FDE): the distance between prediction trajectories and ground-truth at T_{pred}.

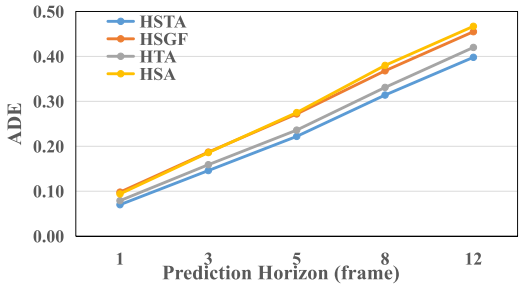
Implementation Details: Our evaluation results are based upon the position information $X_i^t = (x_i^t, y_i^t)$. We use a single-layer MLP to map the inputs to 32 dimensions ((1)). In spatial attention layers, we use two graph attention layers, and the number of features of each node $F = 16$, the attention heads $K = 4, 1$ corresponds to the first and second layer respectively, the parallel heads $h = 8$ in temporal attention layer. Layer Normalization is applied to the output of spatial and temporal attention layers. The dropout value was selected as 0.2 to avoid the risk of overfitting. We use the ReLU as the activation function σ across our model. To acquire the parameters of our network, the Adam optimizer and learning rate was set as 100 epochs and 0.0001, separately. Our model is built using Python with a Pytorch backend and trained with a NVIDIA GTX-1080Ti.

B. ETH and UCY Datasets

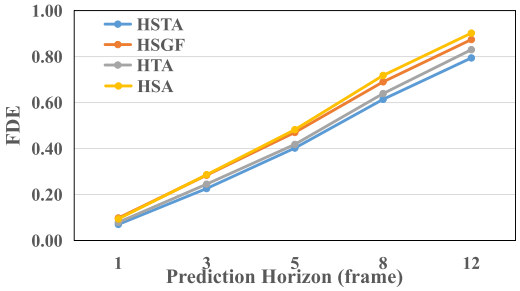
1) Data Pre-Processing: We first conduct experiments on two pedestrian datasets ETH and UCY, then adopt data preprocessing methods in [9], [46], [47]. According to the description in [47], the trajectories of the Hotel scene is different with other scenes, and it is extremely important to narrow this inconsistency. As can be seen from Table I, we use Vanilla LSTM (LSTM) and our model (HSTA) to verify the importance of data augmentations. **Basic** means that no operation is performed by default, and absolute coordinates are used. **Rel** represents the first modification that we use relative coordinates as input instead of absolute coordinates. **Rot+Rel** as the second modification with relative positions and random rotations to reduce directional

TABLE II
ABLATION STUDY ON HSTA. GAT DENOTES THE SPATIAL ATTENTION LAYER, THE ATTENTION HEADS $K = 4, 1$ CORRESPONDS TO THE FIRST AND SECOND LAYER RESPECTIVELY, MHA DENOTES THE TEMPORAL ATTENTION LAYER, THE HEADS $h = 8$, SGF DENOTES THE STATE GATED FUSION LAYER. WE REPORT ON TWO METRICS ADE/FDE FOR $t_{pred} = 12$ (4.8 S) IN METERS

Variant ID	Components			Performance(ADE/FDE)					
	GAT	MHA	SGF	ETH	Hotel	Univ	Zara1	Zara2	Average
1 (HSTA)	✓	✓	✓	**0.40/0.66**	0.44/0.84	**0.56/1.19**	**0.36/0.73**	0.33/0.66	**0.42/0.82**
2 (HSGF)	✓	✓	-	0.42/0.68	0.42/**0.81**	0.64/1.28	0.49/0.95	**0.31/0.64**	0.46/0.87
3 (HSA)	✓	-	-	0.41/0.68	0.50/0.96	0.61/1.23	0.51/0.99	0.32/0.65	0.47/0.90
4 (HTA)	-	✓	-	**0.40**/0.67	**0.41**/0.82	0.60/1.22	**0.36/0.73**	0.34/0.70	0.42/0.83



(a)



(b)

Fig. 6. Quantitative ADE and FDE results for variants on ETH and UCY datasets, HSGF, HSA and HTA represent the variants of ID = 2, 3, 4 in Table II, respectively. The lack of any component will reduce the accuracy of the model. (a) Average ADE results for five scenes. (b) Average FDE results for five scenes.

bias. To ensure the fairness of comparison, we only sample angles from $\mathbb{N}(0, \sigma^2)$ with $\sigma = 180^\circ$ in the training dataset.

The results are displayed in Table I, they show that the metric of Hotel in the Basic is much larger than the average metric, and the performance can be greatly improved through relative and rotation operations. We adopt data augmentations (**Rot+Rel**) in the following experiments.

2) Component Analysis: We verify the effectiveness of each component by ablation experiments, including the spatial attention layer (GAT) (Section III-C), the temporal attention layer (MHA) (Section III-B), and the state gate fusion layer (SGF) (Section III-D). As we consider all neighbors in the scene, and we take the first 8 frames (3.2 seconds) as the history trajectories, then predict the next 12 frames (4.8 seconds) as future trajectories. The quantitative results of various model variants are shown in Table II.

In order to find out which component plays a role in improving long-term prediction, we perform trajectory prediction at different time steps. As shown in Fig. 6, the lack of any component will reduce the accuracy of the model. An interesting thing is that MHA performs best in short-term predictions, because of its higher accuracy in the Hotel scene, which may be the Hotel is more time-dependent.

TABLE III
DIFFERENT ATTENTION HEADS, K REPRESENTS THE HEADS OF THE FIRST LAYER IN GAT, H REPRESENTS THE HEADS OF MHA. WE REPORT THE AVERAGE PERFORMANCE ON TWO METRICS ADE/FDE

ID	Attention heads		Average
	K	H	
1	4	8	**0.418/0.816**
2	4	4	0.424/0.822
3	8	4	0.470/0.904
4	8	8	0.460/0.874

Attention heads: Before performing ablation studies, we first select the appropriate attention heads of GAT and MHA. As shown in Table III, $K = 4, 8$ represents the number of heads of the first layer of GAT, $H = 4, 8$ represents the number of heads of MHA. We can see that the increase of H is beneficial to the improvement of average performance, but the more is not the better. For example, the increase in K will decrease the metrics (ID = 3, 4). We adopt parameter configuration of ID 1 in the following experiments.

State fusion gate: Performing the state fusion gate (4) and considering the performance of two cases with gate (Variant 1) and without gate (Variant 2). We can see that the average accuracy of the model with component SGF is 8.7/5.7(%) better than the model without SGF, which shows that the state fusion gate can better integrate spatio-temporal features into a hidden representation.

Spatial interaction refinement: Employing only the spatial attention layer (Variant 3 with GAT) has the worst performance, which shows that not only the spatial interaction of intra-frame, but also the temporal interaction between inter-frame should be considered for sequence data. The effect of temporal interaction modeling is summarized in following paragraph.

Temporal interaction refinement: With attention head H fixed as 8, only introducing the temporal interaction layer (Variant 4 with MHA) is resultful, which improves the performance by 10.6/7.8(%) compared to Variant 3. This shows that it is more important for sequence data to model temporal interaction than temporal interaction. In addition, it is worth noting that only considering MHA (Variant 4) is better than considering both MAH and GAT without SGF (Variant 2), which further indicates the crucial of SGF.

TABLE IV
EFFECT OF VARYING K, WE TRAIN AND TEST OUR MODEL WITH K SAMPLES. WE SEE THAT THE AVERAGE ERROR METRICS ADE/FDE ARE IMPROVING WITH THE DECREASE OF K, AND OUR MODEL IS LEAST AFFECTED BY THE VARIETY LOSS, DUE TO LOWER DISTRIBUTION VARIANCES. AND TWO ERROR METRICS ARE REPORTED FOR 12 FUTURE TIMESTEPS IN METERS (LOWER IS BETTER)

Model	K=20	K=10	K=5	K=1	% Increase
S-GAN-P [9]	0.558 / 1.118	0.594 / 1.214	0.650 / 1.316	0.846 / 1.758	51.6% / 57.2%
Sophie [22]	0.526 / 1.030	0.566 / 1.122	0.604 / 1.266	0.712 / 1.456	35.3% / 41.4%
Social-BiGAT [25]	0.476 / 0.998	0.488 /1.096	0.527 / 1.260	0.606 / 1.328	27.3% / 33.1%
HSTA(Ours)	**0.398/0.794**	**0.424/0.846**	**0.448/0.898**	**0.468/0.956**	**17.6% / 20.4%**

3) Comparison With Benchmark: We compare our model with following baseline approaches: (1) **V-LSTM**: a vanilla LSTM without modeling interactions, each pedestrian uses an LSTM for modeling, and the parameters are shared between LSTMs. (2) **S-LSTM** [8], **Social-Attention** [13] and **STGAT** [16]: methods that capture human-human interactions with observed trajectory. (3) **Social-BiGAT** [25] and **MATF** [48]: methods that capture human-human interactions with observed trajectory and scene context.

Effect of variety loss: We represent 4 variants with different control settings in Table IV, which are represented by $k = 1, 5, 10, 20$. And S-GAN-P [9] is the first paper to propose the variety loss, Sohpie [22] and Social-BiGAT [25] also used variety loss in their paper and achieved SOTA at the time of publication. Compared with $k = 1$ (essentially represents without variety loss), the performance of all models has been improved, indicating that the variety loss improves performance by stimulating the network to generate multiple samples. Specifically, the ADE and FDE of our model increase more slowly with decreasing k, which is due to the architecture that encodes both spatial and temporal attention reduces the variance of the predicted trajectory distributions. From Fig. 7 we see that the performance is improving with the increase of k, however, our model is least affected by the variety loss, indicating that the hierarchical spatio-temporal attention architecture in HSTA can reduce the variance of the predicted trajectory distributions to improve generalization. In addition, increasing the number of samples without variety loss can also significantly improve the test performance (HSTA1V-1 *vs* HSTA1V-20 in Table V). However, as shown in Fig. 8, for the case without variety loss, although simply adding samples at the beginning can improve performance, it will not help to get better accuracy as the samples increase. On the contrary, the accuracy can be improved substantially with increasing k in the variety loss case.

HSTA vs various baselines: Table V shows the results of our model and various baselines. By capturing interaction, the S-LSTM and attention-based model are notably improved compared to V-LSTM. But the attention-based baselines, which use attention to model interactions, improve upon the S-LSTM by assigning different weights to neighbors. In addition, we see that integrating the spatio-temporal information is helpful for performance improvement, especially for the accuracy of FDE. Our model HSTA1V-1 is evaluated from one sample without variety loss. Compared to V-LSTM, the performance of the model is increased by 34.7/37.7(%).

4) Comparison With Different Prediction Horizon: We adopt the same experimental setting and only change the prediction horizon. Quantitative results are shown in Fig. 9. We compare our model, HSTA20V-20, with V-LSTM and STGAT, and HSTA20V-20 performs most prominently, especially in longer prediction horizons (8-12 time steps). In addition, STGAT and HSTA are both attention-based methods, but STGAT only uses the attention mechanism for spatial interactions without modeling temporal interactions. Our method models spatio-temporal interaction simultaneously. Therefore, the accuracy of our method is improved compared with STGAT.

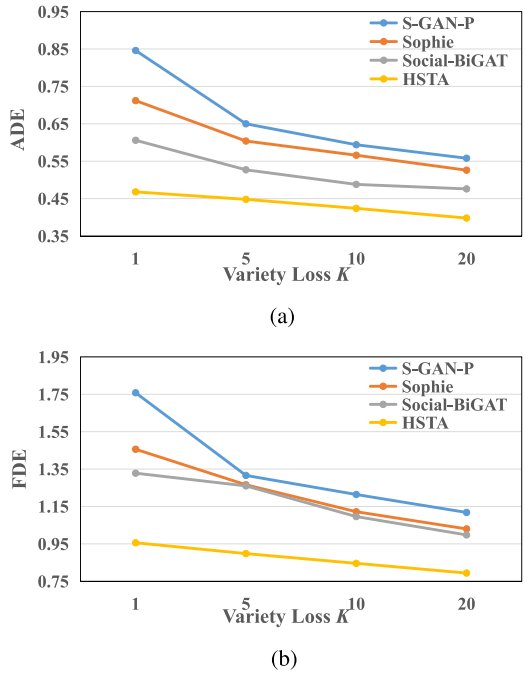


Fig. 7. Effect of variety loss k. Results of S-GAN-P, Sophie, Social-BiGAT refer to [25]. We find that HSTA is less affected by K, due to this architecture reduces the variance of predicted trajectory distribution and improves generalization. (a) Average ADE results for five scenes. (b) Average FDE results for five scenes.

5) Qualitative Results: The qualitative results are shown in Fig. 10, we compare our model (HSTA-20V-20) with LSTM, STGAT (STGAT-20V-20) and variants in the crowd scene. Corresponding to Section IV-B4, the first row (Fig. 10(a)–10(c)) represents the performance of different baselines. Benefiting from the spatio-temporal attention, HSTA is able to capture long-term predictions, which takes advantage from the temporal dependence. And our model outperforms LSTM and STGAT, especially in the long-term prediction we can generate trajectories closer to ground truth.

TABLE V
COMPARISON WITH SEVERAL BASELINES, INCLUDING VANILLA LSTM AND ATTENTION-BASED MODELS. TWO ERROR METRICS ADE/FDE ARE REPORTED FOR 12 FUTURE TIMESTEPS IN METERS (LOWER IS BETTER), AND STGAT WITH $k = 20$ SAMPLES

	Individual	Social			Social + scene context		Ours		
	V-LSTM	Social LSTM [8]	Social Attention [13]	STGAT [16]	Social BiGAT [25]	MATF [48]	HSTA 1V-1	HSTA 1V-20	HSTA 20V-20
ETH	1.13/2.39	0.77/1.60	1.39/2.39	0.65/1.12	0.69/1.29	1.01/1.75	0.50/0.88	0.45/0.77	**0.38/0.62**
Hotel	0.69/1.47	0.38/0.80	2.51/2.91	**0.35/0.66**	0.49/1.01	0.43/0.80	0.52/1.07	0.48/1.00	0.40/0.79
Univ	0.73/1.60	0.58/1.28	0.88/1.75	0.52/1.10	0.55/1.32	**0.44/0.91**	0.55/1.17	0.54/1.13	0.55/1.17
Zara1	0.64/1.43	0.51/1.19	1.25/2.54	0.34/0.69	0.30/0.62	**0.26/0.45**	0.42/0.91	0.38/0.82	0.34/0.71
Zara2	0.54/1.21	0.39/0.89	1.01/2.17	0.29/0.60	0.36/0.75	**0.26/0.57**	0.35/0.75	0.31/0.67	0.32/0.68
Average	0.75/1.59	0.53/1.15	1.41/2.35	0.43/0.83	0.48/1.00	0.48/0.90	0.47/0.96	0.43/0.88	**0.40/0.79**

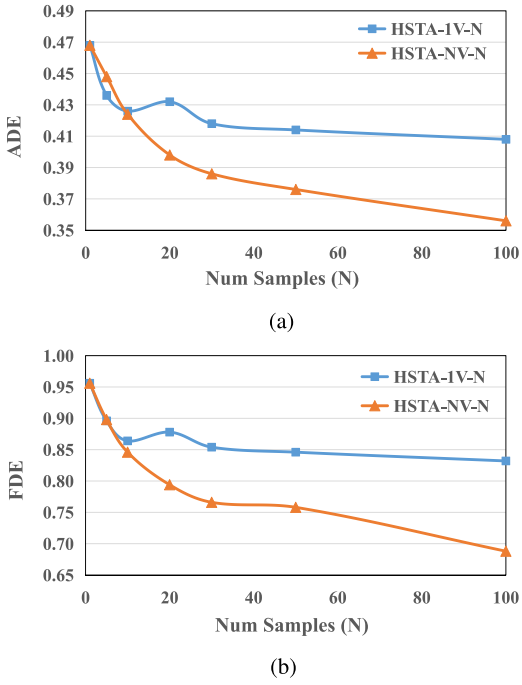


Fig. 8. Effect of the number of samples. HSTA-1V-N represents our model without variety loss during training and is sampled N times during testing. HSTA-NV-N indicates the use of variety loss and uses N samples during both training and testing. (a) Average ADE results for five scenes. (b) Average FDE results for five scenes.

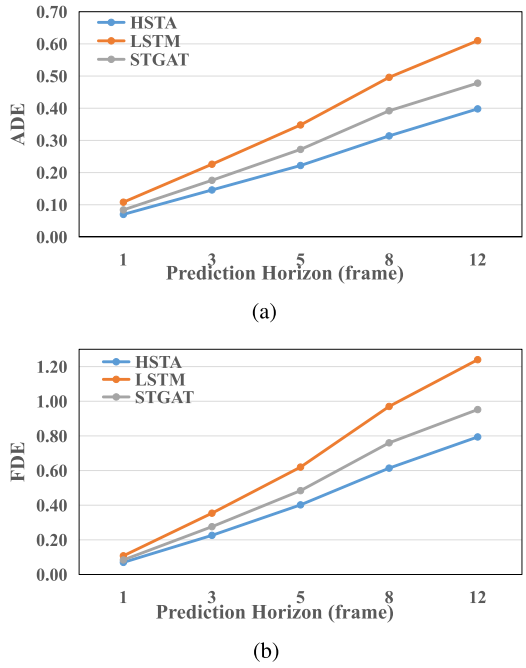


Fig. 9. Quantitative ADE and FDE results for baselines on ETH and UCY datasets, which in meters w.r.t. each future timestep in the prediction horizon are reported. Orange line is the evaluation result of LSTM without modeling interaction, grey for STGAT with spatial interaction, blue for HSTA that includes spatial and temporal interactions. (a) Average ADE results for five scenes. (b) Average FDE results for five scenes.

In order to explore the function of components in Table II, we also visualize each variant. From the Fig. 10(d) and 10(e), we can see that HSA is closer to the ground truth in the short-term forecast, but HTA is more advantageous in the long-term forecast. In addition, by comparing Fig. 10(c) and 10(f), we find that the state gate fusion layer integrates spatio-temporal interaction well. This conclusion is consistent with the former in Fig. 6.

C. NGSIM and Highd Datasets

1) Baselines: We compare HSTA with the following models. **Constant Velocity (CV)** and **Vanilla LSTM (V-LSTM)** are the baselines without modeling interactions. **S-LSTM** [8], **CS-LSTM** [10] and **S-GAN** [9] use the social pooling to capture spatial interactions. **NLS-LSTM** [14] and **MHA-LSTM** [12]

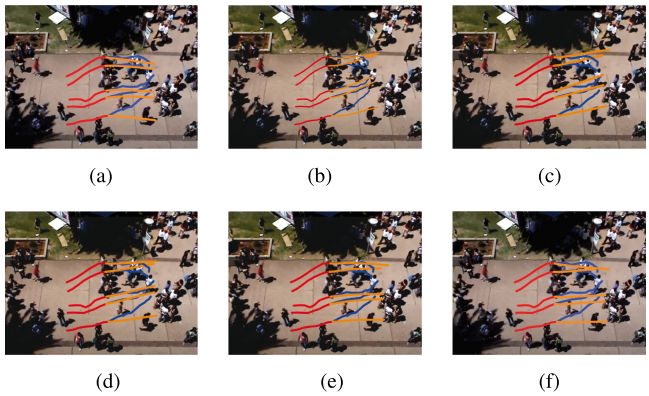


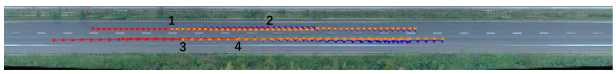
Fig. 10. Comparisons of our model (HSTA-20V-20) with LSTM, STGAT (STGAT-20V-20) and variants. The red represents observed trajectory, the blue represents ground truth, and the yellow represents predicted trajectory. HSA, HTA and HSGF represent the variants of ID = 3,4,2 in Table II, respectively. (a) LSTM. (b) STGAT. (c) HSTA. (d) HSA. (e) HTA. (f) HSGF.

TABLE VI
QUANTITATIVE RESULTS OF BASELINES AND OUR HSTA ON THE HIGHD AND NGSIM DATASET. ALL RESULTS ARE REPORTED IN RMSE IN METERS (LOW NUMERICAL RESULTS ARE BETTER). ALL MODELS TAKE AS AN INPUT 3 S (15 FRAMES). WE NOTICE THAT HSTA (INCLUDING HSTA2 AND HSTA6) HAS BETTER PREDICTIONS THAN OTHER BASELINES

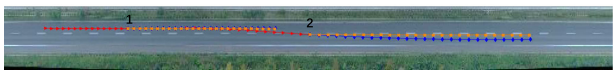
Dataset	Prediction Horizon(s)	CV	V-LSTM	S-LSTM [8]	CS-LSTM [10]	S-GAN [9]	MATF [48]	PiP [49]	NLS-LSTM [14]	MHA-LSTM [12]	HSTA
highD	1	-	-	0.22	0.22	0.30	-	0.17	0.20	0.19	**0.10**
	2	-	-	0.62	0.61	0.78	-	0.52	0.57	0.55	**0.21**
	3	-	-	1.27	1.24	1.46	-	1.05	1.14	1.10	0.32
	4	-	-	2.15	2.10	2.34	-	1.76	1.90	1.84	0.43
	5	-	-	3.41	3.27	3.41	-	2.63	2.91	2.78	0.54
NGSIM	1	0.73	0.68	0.65	0.61	0.57	0.66	0.55	0.56	0.56	**0.52**
	2	1.78	1.65	1.31	1.27	1.32	1.34	1.18	1.22	1.22	**1.12**
	3	3.13	2.91	2.16	2.09	2.22	2.08	1.94	2.02	2.01	**1.82**
	4	4.78	4.46	3.25	3.10	3.26	2.97	2.88	3.03	3.00	**2.61**
	5	6.68	6.27	4.55	4.37	4.40	4.13	4.04	4.30	4.25	**3.49**

TABLE VII
THE QUANTITATIVE RESULTS OF VARIANTS. HTA AND HSA RESPECTIVELY INDICATE WITHOUT SPATIAL ATTENTION LAYER AND TEMPORAL ATTENTION LAYER. TO MEASURE RESULTS IN HIGHD, THE ERROR METRICS RMSE AND FDE ARE APPLIED HERE

	1	2	3	4	5
HSTA	0.10/0.18	0.21/0.40	0.32/0.64	0.43/0.90	0.54/1.14
HTA	0.15/0.23	0.27/0.48	0.40/0.77	0.53/1.07	0.66/1.37
HSA	0.15/0.22	0.26/0.47	0.39/0.75	0.52/1.06	0.66/1.38



(a)



(b)

Fig. 11. Visualized prediction results in highD. The observed history (3 s), ground truth in the future, and the predicted results (5 s) of HSTA model are donated by red, blue, and yellow dashed line, respectively. (a) Keeping lane. (a) Keeping lane.

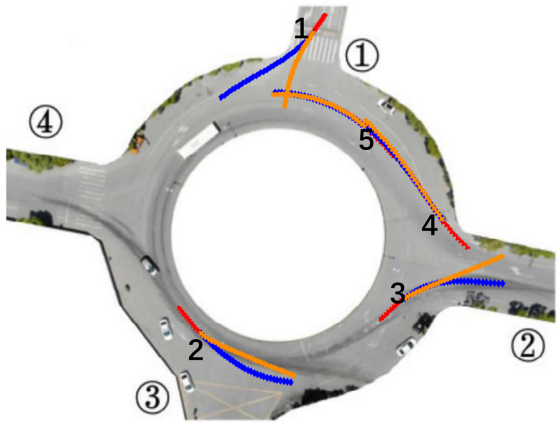


Fig. 12. Visualized prediction results in INTERACTION. Observed history is 1 s and predicted trajectories is 3 s.

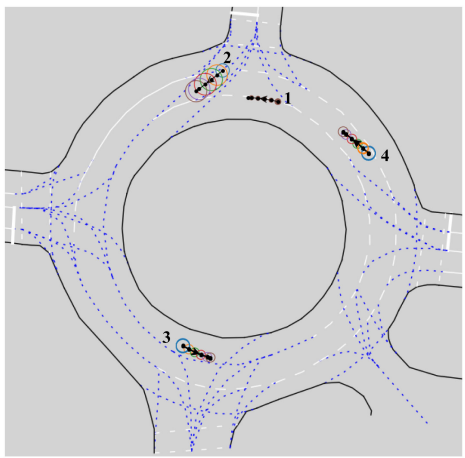


Fig. 13. Attention visualization of the spatial and temporal attention model. The solid dots indicate different time-steps and the arrows mark the direction of trajectories. The target vehicle 1 with solid circles that represent the temporal attention in time-steps. The circles on other vehicles show the spatial attention of surrounding vehicles with respect to the target vehicle. The size of circles represents the attention weight, the larger circle represents higher attention.

are two main attention models, **MATF** [48] and **PiP** [49] are two latest methods.

2) Quantitative Evaluation: **Comparison with Existing Works**

We compare HSTA with various benchmarks in Table VI, reporting the root of the mean squared error (RMSE) for different time steps on the highD and NGSIM datasets. Overall, HSTA exceeds baselines on the vehicle datasets, the previous state of the art on the RMSE metric is PiP with an error of 2.63 at 5 s and 0.17 at 1 s on highD. Our HSTA has an error of 0.54 at 5 s and 0.10 at 1 s on highD, which is about 80% and 40% less than the state of the art respectively.

From each time step, as we can observe from the results, the two attention-based models NLS-LSTM and MHA-LSTM are better than other baselines. However, HSTA shows the best result in highD and NGSIM, which indicates that our GAT based on graph structure can better capture spatial interactions than grid-base attention in NLS-LSTM and MHA-LSTM. In terms of long-term prediction ($T_{pred} = 5$ s), we find that the model with MHA component has better performance because it can better capture time dependencies.

Additionally, We notice the RMSE on the NGSIM dataset is higher than that of the highD dataset. The reason may be the

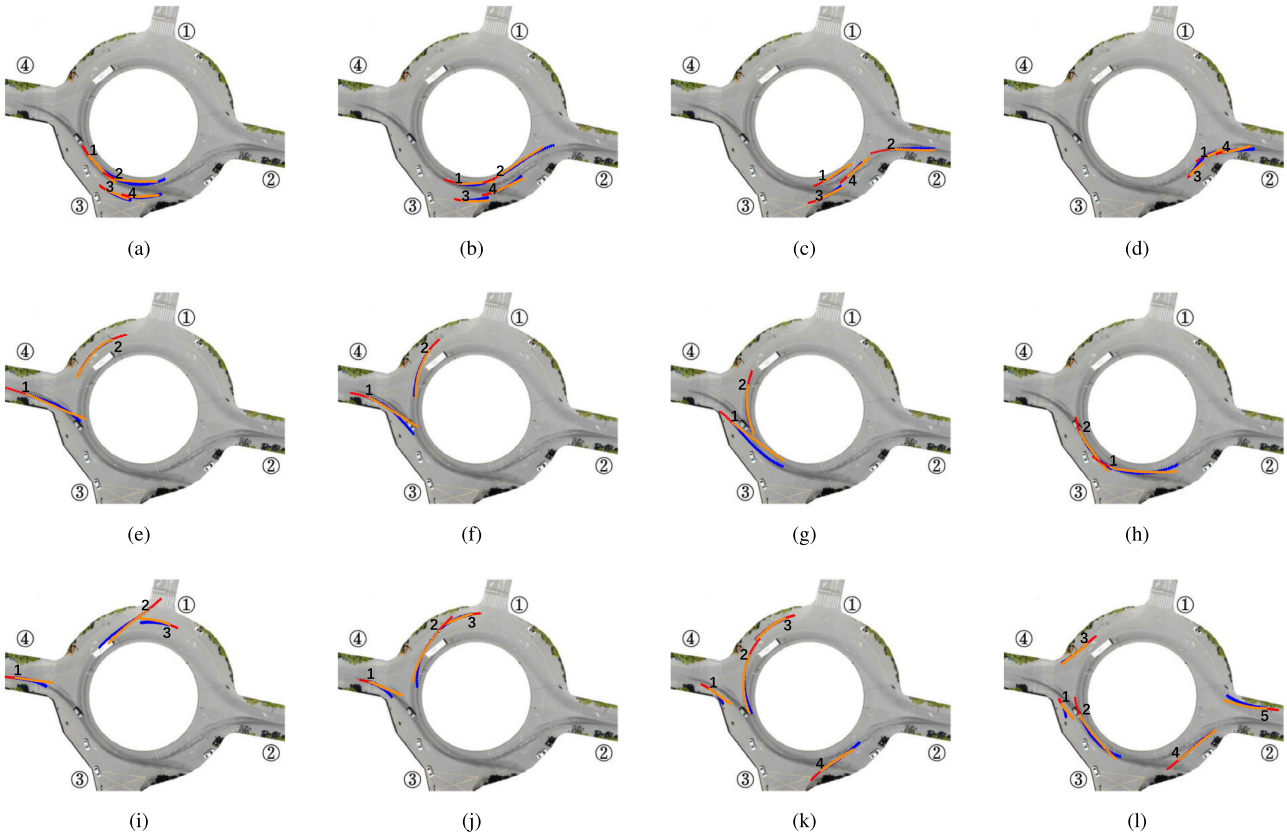


Fig. 14. Examples of complex interactions. (a)–(d) represent the cooperative behavior in the roundabout, (e)–(h) represent aggressive behavior of entering roundabout, (i)–(l) represent conservative behavior of entering roundabout.

noise caused by incorrect annotations of the NGSIM dataset, which leads to unsatisfactory results. Besides, all those methods that incorporate interactions present a better performance than the reference methods, which include the constant velocity and the vanilla LSTM methods. This result indicates that the vehicles in neighborhood have vital influence on the trajectory generation of object vehicle.

Ablation Study: Here we conduct an ablation study to analyze the components of HSTA, including the Graph Attention Network (GAT) block (Section III-C) and the Multihead Attention (MHA) block (Section III-B). To testify the capability of GAT and MHA blocks, we respectively consider two variant models HTA (without spatial attention layer) and HSA (without temporal attention layer), as well as train and test them in the highD dataset. Moreover, two metrics are used to evaluate variants: the root of the mean squared error (RMSE) and the final displacement error (FDE) that considers only the prediction precision at the end point. According to the quantitative results in Table VII, our HSTA which combines GAT and MHA has achieved the state of the art performance. For the RMSE metric, HSA has better results than HTA because the graph structure could better represent the relationships between vehicles. As expected, we see that HTA outperforms HSA in the case of longer predictions on the FDE metric. Interestingly, HTA has the least training time, it achieves this result because it overcomes the limitation of complex graph structures and could parallel the computation for all vehicles.

3) Qualitative Evaluation: In order to better study the ability of spatio-temporal model in modeling social behavior, we conduct qualitative analysis of the predicted trajectory in two typical scenarios (highways and roundabout). Specifically, highways comes from the highD dataset, and roundabout comes from the INTERACTION dataset [50] that contains the highly interactive behavior of heterogeneous traffic participants from different countries.

Highways: Fig. 11(a) shows prediction results when performing a lane keeping in dense traffic. It could be noticed that there are vehicles around vehicle 1, 2, 3 and 4, they actually have no space to change lanes. The results are consistent with our analysis, these vehicles keep going straight without changing lanes. Notably, the future trajectories are close to the ground truth at each time step, indicating that our model has mastered the velocity and acceleration characteristic of the vehicles. Fig. 11(b) shows another example in which the vehicle 2 turns right to change lanes. In this case, turning right to change lanes is reasonable because there is no vehicle in the right lane of vehicle 2 and vehicle 1 is far away from it. Our model successfully predicts its intention and maintained high accuracy in the first 10 time steps.

Roundabout: In order to verify that our model has the ability to model complex interactions, we train and test our model in

INTERACTION dataset, and choose the roundabout scene for qualitative evaluation. From Fig. 12, we can observe that HSTA achieves better performance on vehicle 2, 4, 5 that are already in the roundabout, and there are errors for vehicle 1, 3 just entering or leaving the roundabout. For vehicle 1 and 3, our model can accurately predict their positions before 1 s, but predicted trajectories overshoot after reaching the final point. The reason might be that their velocities are lower and the relative distance traveled within 1 s is short, however, the input of our model is relative distances. If the relative distance is shorter, it will lead to undesirable results.

We also visualize the learned attention weight in Fig. 13, our model successfully assigns spatial attention to the surrounding vehicles and temporal attention to the time-steps. In this case, HSTA allocates more attention to the front vehicle 2, although the rear vehicle 4 is at the same distance as the front vehicle 2, the importance is not as high as the front car, which is consistent with the actual driving situation. In addition, despite vehicle 3 is far away, it is still assigned a similar attention weight to vehicle 4, which indicates that our model assigns spatial attention not only considering the distance but also the current state of the vehicle. On the other hand, HSTA assigns different weights to each frame for the all available observation interval, with the first and last frames receiving higher attention.

In addition, we analyze the cooperative behavior in the roundabout, the aggressive and conservative behavior of entering roundabout in Fig. 14. As shown in Fig. 14(a)–(d), our model has satisfactorily learned to cooperate with each other. They not only keep safe driving in the roundabout, but also drive out of roundabout in an orderly manner. From Fig. 14(e)–(h), vehicle 1 shows an altruistic behavior, which maximizes the reward of vehicle 2 and does not consider its own outcome. And our model has learned to yield to vehicle 2 when vehicle 1 entering the roundabout. In Fig. 14(i)–(l), vehicle 1 performs an egoistic behavior that maximizes its own outcome without concerning of the reward of vehicle 2. Therefore, although vehicle 1 has found that vehicle 2 is approaching, it still chooses to enter the roundabout directly without considering the influence of vehicle 2. These successful cases show that HSTA have the ability to capture complex interactions in different scenes.

V. Conclusion

In this paper, we propose HATS, a hierarchical spatio-temporal architecture for forecasting trajectories that outperforms prior state-of-the-art methods on publicly available datasets. Unlike previous researches, our model is not only able to capture spatial interactions among agents, but is also able to encode temporal dependencies. We combine graph attention mechanisms with multi-head attention mechanisms to extract better features, which is able to generate more accurate future trajectories. Through our visualizations, we demonstrate that HSTA is able to capture complex interactions to generate more reasonable and socially acceptable trajectories in different scenes. Future works will concentrate on the construction of complex scene topology, we hope to fix neighborhood agents, instead of considering all agents in the scene.

References

[1] G. Chen, C. Kai, Z. Lijun, Z. Liming, and A. Knoll, "VCANet: Vanishing point guided context-aware network for small road hazards detection," *Automot. Innov.*, 1–13, 2021.

[2] A. Elnagar, "Prediction of moving objects in dynamic environments using kalman filters," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Automat.*, 2001, pp. 414–419.

[3] A. Jain *et al.*, "Discrete residual flow for probabilistic pedestrian behavior prediction," in *Proc. Conf. Robot Learn.*, 2020, pp. 407–419.

[4] X. Li, X. Ying, and M. C. Chuah, "Grip++: Enhanced graph-based interaction-aware trajectory prediction for autonomous driving," 2019, *arXiv:1907.07792*.

[5] A. Khosroshahi, E. Ohn-Bar, and M. M. Trivedi, "Surround vehicles trajectory analysis with recurrent neural networks," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst.*, 2016, pp. 2267–2272.

[6] F. Altché and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 353–359.

[7] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture," in *Proc. IEEE Intell. Veh. Symp.*, 2018, pp. 1672–1678.

[8] 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.*, 2016, pp. 961–971.

[9] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2255–2264.

[10] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 1468–1476.

[11] H. Xue, D. Q. Huynh, and M. Reynolds, "SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1186–1194.

[12] K. Messaoud, I. Yahiaoui, A. Verroust, and F. Nashashibi, "Attention based vehicle trajectory prediction," *IEEE Trans. Intell. Veh.*, vol. 6, no. 1, pp. 175–185, Mar. 2021.

[13] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 4601–4607.

[14] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," in *Proc. IEEE Intell. Veh. Symp.*, 2019, pp. 975–980.

[15] P. VeliČković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.

[16] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, "Stgat: Modeling spatial-temporal interactions for human trajectory prediction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6272–6281.

[17] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[18] A. Treuille, S. Cooper, and Z. Popović, "Continuum crowds," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 1160–1168, 2006.

[19] G. Antonini, M. Bierlaire, and M. Weber, "Discrete choice models of pedestrian walking behavior," *Transp. Res. Part B: Methodological*, vol. 40, no. 8, pp. 667–687, 2006.

[20] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Transa. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, Feb. 2008.

[21] R. Emonet, J. Varadarajan, and J.-M. Odobez, "Extracting and locating temporal motifs in video scenes using a hierarchical non parametric bayesian model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3233–3240.

[22] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1349–1358.

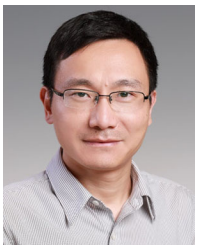
[23] E. Leurent and J. Mercat, "Social attention for autonomous decision-making in dense traffic," in *Proc. NeurIPS Workshop Mach. Learn. Autonomous Driving*, 2019, pp. 1–11.
[24] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–14.
[25] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 137–146.
[26] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "Desire: Distant future prediction in dynamic scenes with interacting agents," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 336–345.
[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. NIPS Workshop Deep Learning*, 2014, pp. 1–9.
[29] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[30] N. Nikhil and B. Tran Morris, "Convolutional neural network for trajectory prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 186–196.
[31] 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 Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14424–14432.
[32] C. Wang, S. Cai, and G. Tan, "Graphtcn: Spatio-temporal interaction modeling for human trajectory prediction," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vision*, 2021, pp. 3450–3459.
[33] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[34] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1227–1236.
[35] G. Chen, H. Cao, J. Conradt, H. Tang, F. Rohrbein, and A. Knoll, "Event-based neuromorphic vision for autonomous driving: A paradigm shift for bio-inspired visual sensing and perception," *IEEE Signal Process. Mag.*, vol. 37, no. 4, pp. 34–49, Jul. 2020.
[36] A. Santoro *et al.*, "A simple neural network module for relational reasoning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4967–4976.
[37] A. Ajay *et al.*, "Combining physical simulators and object-based networks for control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 3217–3223.
[38] H. Yao, X. Tang, H. Wei, G. Zheng, Y. Yu, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5668–5675.
[39] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 7444–7452.
[40] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *Proc. Eur. Conf. Comput. Vision*, 2020, pp. 507–523.
[41] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *Proc. IEEE 12th Int. Conf. Comput. Vis.* IEEE, 2009, pp. 261–268.
[42] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3542–3549.
[43] C. James and J. Halkias, "Us highway 101 dataset," in *Federal Highway Administration (FHWA)*, Tech. Rep. FHWA-HRT-07-030, 2007.
[44] C. James and J. Halkias, "Us highway i-80 dataset," in *Federal Highway Administration (FHWA)*, Tech. Rep. FHWA-HRT-07-030, 2007.
[45] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highD Dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 2118–2125.
[46] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "SR-LSTM: State refinement for LSTM towards pedestrian trajectory prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12085–12094.
[47] C. Schöller, V. Aravantinos, F. Lay, and A. Knoll, "What the constant velocity model can teach us about pedestrian motion prediction," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 1696–1703, Apr. 2020.
[48] T. Zhao *et al.*, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12126–12134.
[49] H. Song, W. Ding, Y. Chen, S. Shen, M. Y. Wang, and Q. Chen, "PiP: Planning-informed trajectory prediction for autonomous driving," in *Proc. Eur. Conf. Comput. Vision*, 2020, pp. 598–614.
[50] W. Zhan *et al.*, "INTERACTION dataset: An INTERnational, adversarial and cooperative moTION dataset in interactive driving scenarios with semantic maps," 2019, *arXiv:1910.03088*.

Ya Wu received the B.Eng. degree in thermal energy and power engineering from Harbin Engineering University, China, in 2015, and the M.Eng. degree in power engineering and engineering thermo-physics from Jilin University, China, in 2018. He is currently working toward the Ph.D. degree in energy and environmental protection with Tongji University, China. His research interests include the trajectory prediction, motion planning and reinforcement learning.

Guang Chen (Member, IEEE) received the B.S. and M.Eng. degrees in mechanical engineering from Hunan University, China, and the Ph.D. degree from the Faculty of Informatics, the Technical University of Munich, Germany. He is currently a Research Professor with Tongji University and a Senior Research Associate (guest) with the Technical University of Munich. He is Leading the Intelligent Perception System group, Tongji University. His research interests include computer vision, image processing and machine learning, and the bio-inspired vision with applications in robotics and autonomous vehicle. He was a Research Scientist with Fortiss GmbH, the Research Institute of Technical University of Munich from 2012 to 2016, and a Senior Researcher with the Chair of Robotics, Artificial Intelligence and Real-time Systems, Technical University of Munich from 2016 to 2017. He was Awarded the program of Tongji Hundred Talent Research Professor 2018.

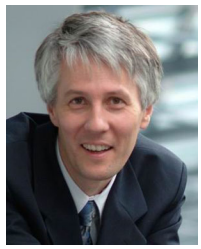
Zhijun Li (Senior Member, IEEE) received the Ph.D. degree in mechatronics, Shanghai Jiao Tong University, China, in 2002. From 2003 to 2005, he was a Postdoctoral Fellow with the Department of Mechanical Engineering and Intelligent systems, The University of Electro-Communications, Tokyo, Japan. From 2005 to 2006, he was a Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore, and Nanyang Technological University, Singapore. From 2017, he is a Professor with the Department of Automation, University of Science and Technology, Hefei, China. From 2019, he is the Vice Dean of the School of Information Science and Technology, University of Science and Technology of China, China. From 2016, he was the Co-Chairs of IEEE SMC Technical Committee on Bio-mechatronics and Bio-robotics Systems (B^2S), and IEEE RAS Technical Committee on Neuro-Robotics Systems. He is the Editor-at-large of *Journal of Intelligent & Robotic Systems*, and an Associate Editor for several IEEE Transactions. His research interests include wearable robotics, tele-operation systems, nonlinear control, neural network optimization, etc.

Lijun Zhang is currently a Professor of Tongji University, the Dean of the School of Automotive Studies, Tongji University, and the Vice Director of Collaborative Innovation Center for Intelligent Electric Vehicle. His research interests include automotive comfort engineering, active chassis system dynamics control, electric vehicle powertrain dynamics, and intelligent vehicle sensing and data fusion. As Principal Investigator, he had led more than 15 national and provincial projects, including the National Key Basic Research Program (973 Program), National Natural Science Foundation of China (NSFC), National High Technology Research and Development Program of China (863 Program), National Key Technology R&D Program, Shanghai Key Science and Technology Program, Shanghai Automotive Industry Development Funding Program, etc. He had gained one 2nd Class National Science and Technology Prize and three 1st Class Shanghai Science and Technology Prizes.



Lu Xiong received the B.E., M.E., and the Ph.D. degrees in vehicle engineering from the School of Automotive Studies, Tongji University, Shanghai, China, in 1999, 2002, and 2005, respectively. From November 2008 to 2009, he was a Postdoctoral Fellow with the Institute of Automobile Engineering and Vehicle Engines, University of Stuttgart, Germany, with Dr. Jochen Wiedemann. He is currently a Professor with Tongji University. He is also an Executive Director of the Institute of Intelligent Vehicles and an Associate Director of the Clean Energy Automotive Engineering Center, Tongji University. His research interests include perception, decision and planning, dynamics control and state estimation and testing and evaluation of autonomous vehicles.

Zhengfa Liu received the B.E. degree in electronic and communication engineering from National Center for Space Science, Chinese Academy of Sciences, Beijing, China, in 2018. He is currently working toward the Ph.D. degree in vehicle engineering with Tongji University, Shanghai, China. His research interests include the adversarial learning, multi-sensor fusion, and domain adaptation.



Alois Knoll (Senior Member, IEEE) received the diploma (M.Sc.) degree in electrical/communications engineering from the University of Stuttgart, Germany, in 1985, and the Ph.D. degree (*summa cum laude*) in computer science from the Technical University of Berlin, Germany, in 1988. He was the faculty of the Computer Science Department of TU Berlin until 1993. He joined the University of Bielefeld, as a Full Professor and the Director of the Research Group Technical Informatics until 2001. Since 2001, he has been a Professor with the Department of Informatics, TU München. He was also on the Board of Directors of the Central Institute of Medical Technology, TUM (IMETUM). From 2004 to 2006, he was an Executive Director of the Institute of Computer Science, TUM. His research interests include cognitive, medical and sensor-based robotics, multi-agent systems, data fusion, adaptive systems, multimedia information retrieval, model-driven development of embedded systems with applications to automotive software and electric transportation, as well as simulation systems for robotics and traffic.