Vehicle Trajectory Prediction Using LSTMs with Spatial-Temporal Attention Mechanisms

Lei Lin, IEEE Member, Weizi Li*, Huikun Bi, and Lingqiao Qin

***Abstract*—Accurate vehicle trajectory prediction can benefit a variety of Intelligent Transportation System applications such as traffic simulation and driver assistance. The need of this ability is pronounced with the emergence of autonomous vehicles, as they require the prediction of nearby vehicles' trajectories in order to navigate safely and efficiently. Recent studies based on deep learning have greatly improved the prediction accuracy. However, one prominent issue is that these models often lack explainability. We alleviate this issue by proposing STA-LSTM, an LSTM model with spatial-temporal attention mechanisms. STA-LSTM not only achieves performance comparable to other state-of-the-art models in terms of prediction accuracy over $1\ s$ time horizon but, more importantly, identifies the influence of historical trajectories and neighboring vehicles on the target vehicle via spatial-temporal attention weights. We provide the analyses of the learned attention weights in various highway scenarios based on criteria such as target vehicle class, target vehicle location, and traffic density. An analysis showing that STA-LSTM can capture fine-grained lane-changing behaviors is also provided.**

***Index Terms*—vehicle trajectory prediction, deep learning, long short-term memory, model explainability, spatial-temporal attention mechanisms.**

I. Introduction

SINCE the first autonomous driving competition hosted by the Defense Advanced Research Projects Agency (DARPA) in 2005 [1], autonomous vehicles (AVs) have attracted extensive attention from both academia and industry. With the recent advancements in sensing technologies (e.g., LiDAR, radar, and camera) as well as machine learning algorithms, the research and development of autonomous driving have achieved tremendous progress. Many automobile manufacturers and technology companies are developing either partial or fully autonomous vehicles.

There are two main approaches to achieving autonomous driving. The first is the end-to-end approach, which directly maps raw sensor data to control commands using a *single* model, commonly one or more neural networks [2], [3], [4], [5]. The second is the traditional engineering approach [6], [7], which involves *multiple* modules, including detection, tracking, prediction, and planning, for interpreting sensor input in order to generate control commands. Both approaches have merits and drawbacks. However, as the safety of AVs is the leading concern, the traditional engineering approach is likely to prevail in the near future due to its better model interpretability and controllability.

One crucial task of the traditional engineering approach for autonomous driving is the prediction of nearby vehicles' trajectories. The AV requires this information in order to drive safely and efficiently. In this work, we focus on vehicle trajectory prediction on highways, where the dominant traffic participants are motor vehicles, including cars and trucks. In the following text, we refer to the vehicle whose trajectory is being predicted as the *target vehicle*, and the surrounding vehicles of the target vehicle as *neighboring vehicles*.

Among many techniques for predicting vehicle trajectories, Recurrent Neural Networks (RNNs) have offered state-of-the-art performance. This is mainly due to their capability in modeling non-linear temporal dependencies in sequential data [8], [9], [10], [11], [12]. RNNs take historical trajectory data of the target vehicle (for example, the last 10-second trajectory) as the input and predict its trajectory over a certain time horizon. RNNs are particularly effective because they consider both the local information among vehicles (e.g., instantaneous interactions between the target vehicle and its leading vehicle) and the long-term information stored in memory cells [8], [9].

Some studies have explored how neighboring vehicles will affect the target vehicle. For example, Altché and de La Fortelle [12] applied Long Short-term Memory (LSTM) (a variant of RNN) to predict the target vehicle's movement using trajectories of both the target and neighboring vehicles. Deo and Trivedi [11] enhanced LSTMs with convolutional social pooling layers for encoding the historical trajectories of neighboring vehicles for prediction. These results have suggested that including the information of neighboring vehicles into LSTM improves the trajectory prediction accuracy.

While the aforementioned studies can provide state-of-the-art performance, they offer limited model explainability. In particular, how the long-term information embedded in historical trajectories [8], [9] and the information of neighboring vehicles [12], [11] impact the prediction results is left unexplored. In this work, we aim to answer the following questions: Which part of the historical trajectories of the target vehicle or neighboring vehicles determines the future motion of the target vehicle? Among all neighboring vehicles, which one has more influence on future behaviors of the target vehicle? At which positions relative to the target vehicle will neighboring vehicles have significant effects on it? Answering these questions at either the temporal level or the spatial level can help us better understand a driver's decision-making process, identify various driving styles, design realistic traffic simulation models, and ultimately develop safe and efficient autonomous driving.

Attention mechanisms proposed by Bahdanau et al. [13] have been used to improve RNNs' explainability [14], [15], [16]. The idea is to learn attention weights to identify which inputs of a learning task are more influential. This has inspired

*=Corresponding author.

STA-LSTM—an LSTM model with spatial-temporal attention mechanisms for vehicle trajectory prediction.

STA-LSTM is learned and evaluated using the NGSIM dataset [17]. As a result, it not only achieves performance comparable to other state-of-the-art techniques in terms of prediction accuracy but also explains the influence of historical trajectories and neighboring vehicles on the target vehicle via attention weights. We provide in-depth analyses of the learned attention weights in scenarios which contain different sets of vehicles and environment factors, including target vehicle classes (e.g., cars and trucks), target vehicle locations, and neighboring vehicle densities. We have also analyzed the attention weights associated with specific driving behaviors of the target vehicle, and have found that the learned attention weights can be used to interpret the target vehicle's lane-changing behaviors. In summary, the main contributions of this work are the following:

- STA-LSTM, an LSTM model with spatial-temporal attention mechanisms, is developed for predicting vehicle trajectories.
- The proposed attention mechanisms at the temporal level can identify important historical trajectories for determining future behaviors of the target vehicle.
- The proposed attention mechanisms at the spatial level can rank neighboring vehicles in terms of their influences on future motions of the target vehicle.
- In-depth analyses of the learned attention weights in traffic scenarios with various vehicle and environment factors are provided.
- Specific driving behaviors of the target vehicle through the learned attention weights are analyzed. In particular, lane-changing behaviors of the target vehicle are found to be explainable through the attention weights.

II. Related Work

A. Vehicle Trajectory Prediction Using Traditional Methods

Conventionally, three types of approaches exist for vehicle trajectory prediction: namely, physics-based, maneuver-based, and interaction-aware [18]. The physics-based methods usually consider vehicle kinematic and dynamic constraints such as yaw rate and acceleration rate, and environmental factors such as the friction coefficient of a road surface. While this approach can achieve short-term (less than a second) motion prediction, it is incapable of predicting motion changes of a vehicle due to certain maneuvers (e.g., sudden slowing down) or interactions with neighboring vehicles (e.g., braking for the leading vehicle).

Maneuver-based methods are proposed to compensate the limitations of physics-based methods by using specific maneuvers of a driver (e.g., go straight, turn left or right) to predict vehicle trajectories over a longer time horizon (i.e., greater than one second). To list some examples, Mandalia et al. [19] used support vector machines (SVMs) to infer driver intentions with a focus on lane-changing decisions. Schreier et al. [20] proposed a Bayesian method to predict long-term vehicle trajectories and provided a criticality assessment of the prediction results for a driver assistance system. Tomar et al. [21] exploited Multi-Layer Perceptrons (MLP) to forecast vehicle trajectories in lane-changing behaviors.

Most physics-based and maneuver-based approaches do not account for interactions among vehicles. This has motivated the development of interaction-aware methods, which take into account the inter-dependencies of vehicle maneuvers for trajectory prediction. To provide a few examples, Gindele et al. [22] modeled the mutual influence between vehicles using factored states in forecasting. Lefèvre et al. [23] studied the joint motion and conflicting intentions of vehicles and provided a risk assessment for vehicles operating at intersections.

B. Vehicle Trajectory Prediction Using Deep Learning

A number of studies have applied deep learning—especially RNN and its variant, LSTM—for vehicle trajectory prediction [8], [9], [10], [11], [12], [24]. For example, Deo and Trivedi [11] used a convolutional social pooling network combined with LSTMs for predicting vehicle trajectories on highways. Altché and de La Fortelle [12] applied LSTMs for predicting the longitudinal velocity of a vehicle on a highway segment by taking the trajectories of its nine surrounding vehicles into account. Lee et al. [25] proposed a Deep Stochastic IOC-RNN-Encoder-Decoder framework (DESIRE) to predict the trajectories of interacting road users in dynamic scenes. Kim et al. [26] proposed an LSTM-based trajectory prediction approach using an occupancy grid map to characterize a driving environment.

C. Attention Mechanisms

Attention mechanisms can be naturally integrated with RNN for sequential data analysis. For example, Zhou et al. [14] proposed an attention-based bidirectional LSTM model to capture key semantic information for relation classification in natural language processing. Lin et al. [16] applied an LSTM model with attention mechanisms to address time series for explainable disease classification.

In trajectory prediction, attention mechanisms have been applied to pedestrians [27], [28]. Fernando et al. [27] enhanced LSTMs with soft- and hard attention mechanisms for predicting pedestrian trajectories. The soft attention mechanism focuses on the target pedestrian while the hard attention mechanism focuses on neighboring pedestrians. However, this method does not capture the interactions between the target pedestrian and neighboring pedestrians. Zhang et al. [28] also proposed an attention mechanism for pedestrian trajectory prediction. Their method enabled the interpretation of neighboring pedestrians' effect on the target pedestrian at the spatial level but failed to identify which portion of the historical trajectories influences the prediction result at the temporal level. To the best of our knowledge, our technique is among the few that have applied LSTMs with spatial-temporal attention mechanisms for vehicle trajectory prediction.

III. Methodology

Following the same setting proposed by Deo and Trivedi [11], we first discretize the space centered around the target vehicle into a 3×13 grid. The rows represent the left, current, and right lanes with respect to the target vehicle's

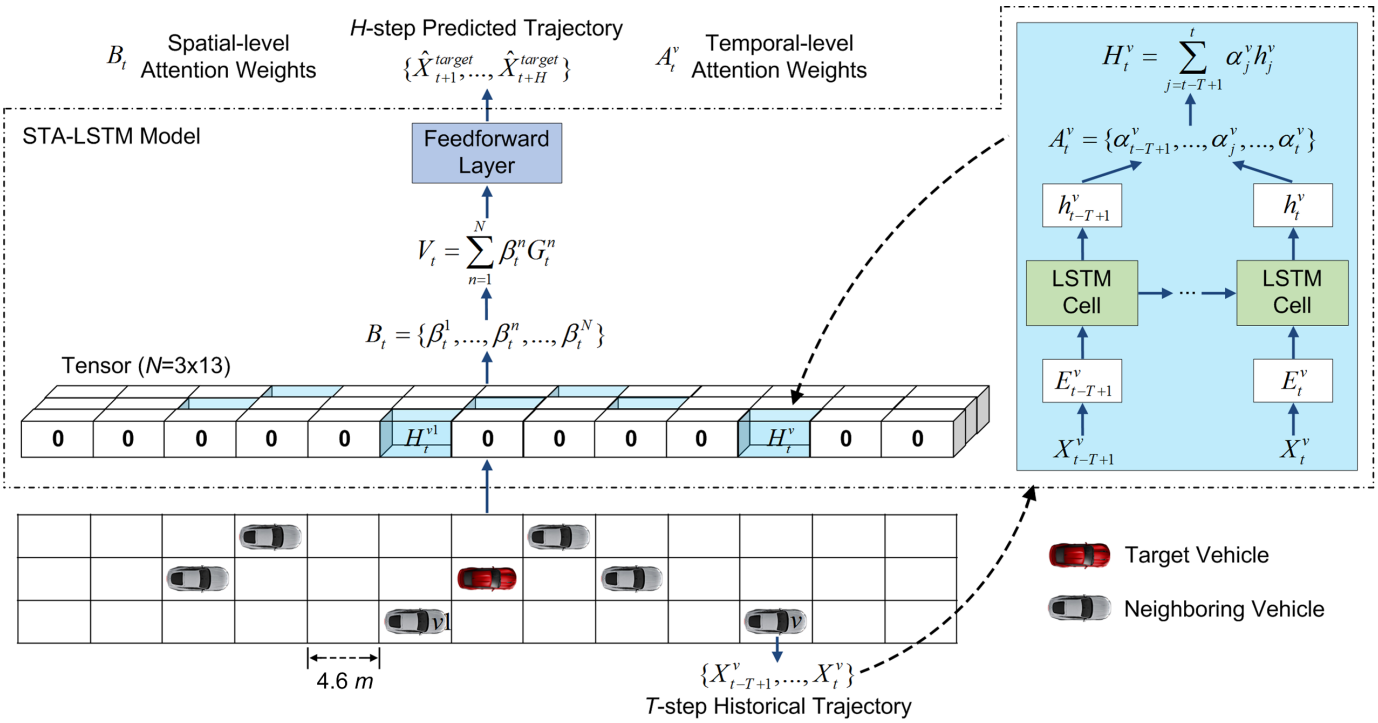


Fig. 1: The schematic view of our approach and the architecture of the STA-LSTM model. The inputs to STA-LSTM are T-step historical trajectories of all vehicles within the 3×13 grid centered around the target vehicle. Each trajectory is processed by an LSTM model. An example at time step t involving vehicle v is shown (v can be either the target vehicle or a neighboring vehicle). The trajectory $\{X^v_{t-T+1}, \ldots, X^v_t\}$ is used to generate hidden states $\{h^v_{t-T+1}, \ldots, h^v_t\}$. Then, all hidden states are used to compute the temporal-level attention weights associated with each vehicle denoted by A^v_t. Next, $\{h^v_{t-T+1}, \ldots, h^v_t\}$ are combined with A^v_t to derive a cell value of the 3×13 tensor denoted by H^v_t. After filling the tensor with either H^v_t (the corresponding grid cell has a vehicle) or $\mathbf{0}$ (the corresponding grid cell has no vehicle), we can use it to compute the spatial-level attention weights associated with all vehicles (denoted by B_t) and predict the H-step trajectory of the target vehicle $\{\hat{X}^{target}_{t+1}, \ldots, \hat{X}^{target}_{t+H}\}$. Note that each vehicle's front-bumper position is used to compute its belonging grid cell and one vehicle only contributes to the computation of one cell.

location. The columns represent discretized grid cells with a width of 4.6 m (15 $feet$) each.

Vehicles that locate inside the 3×13 grid (except for the target vehicle) are considered neighboring vehicles. Each neighboring vehicle is assigned to a grid cell based on its relative position to the target vehicle. We interpret the position of a vehicle as its front-bumper position. One vehicle thus only belongs to one cell. For example, a neighboring vehicle located at 11 m in front of the target vehicle will be assigned to the 3rd cell ($3 = \lceil 11/4.6 \rceil$) ahead of the target vehicle's cell.

The inputs to our STA-LSTM model are T-step historical trajectories of all vehicles within the 3×13 grid. Each vehicle's trajectory is processed by its corresponding LSTM model. The output is an H-step predicted trajectory of the target vehicle. During this process, temporal-level and spatial-level attention weights are learned. The temporal-level attention weights can be used to identify which part of historical trajectories from the target and neighboring vehicles influences the prediction result. The spatial-level attention weights can be used to explain the influence of neighboring vehicles on the prediction result. Next, we introduce the computation of these attention weights.

1) Temporal-level Attention Calculation: At time step t, the T-step historical trajectory $\{X^v_{t-T+1}, \ldots, X^v_t\}$ of vehicle v (v can be either the target vehicle or a neighboring vehicle) is taken as the input to an LSTM model. Consider the hidden states of the LSTM model $S^v_t = \{h^v_{t-T+1}, \ldots, h^v_j, \ldots, h^v_t\}, S^v_t \in \mathbb{R}^{d \times T}, h^v_j \in \mathbb{R}^{d \times 1}$, where d is the hidden state length. After these hidden states are generated, the temporal attention weights associated with v, $A^v_t = \{\alpha^v_{t-T+1}, \ldots, \alpha^v_j, \ldots, \alpha^v_t\}$, are computed as follows:

$$A^v_t = softmax(\tanh(W_\alpha S^v_t)), A^v_t \in \mathbb{R}^{1 \times T}, W_\alpha \in \mathbb{R}^{1 \times d}, \tag{1}$$

where W_α is a set of parameters to be learned.

Next, we combine the hidden states S^v_t and the computed temporal attention weights A^v_t to derive the tensor cell value that corresponds to v:

$$H^v_t = S^v_t (A^v_t)^T = \sum_{j=t-T+1}^{t} \alpha^v_j h^v_j, H^v_t \in \mathbb{R}^{d \times 1}. \tag{2}$$

Collectively, the values of all tensor cells are employed to compute spatial-level attention weights and predict the trajectory of the target vehicle.

2) Spatial-level Attention Calculation: We can represent all tensor cell values at time step t as $G_t = \{G^1_t, \ldots, G^n_t, \ldots, G^N_t\}, G_t \in \mathbb{R}^{d \times N}, G^n_t \in \mathbb{R}^{d \times 1}$, where N

represents the total number of tensor cells (i.e., 39 in this study). G_t^n is taking the following form:

$$G_t^n = \begin{cases} H_t^v, \text{if any vehicle } v \text{ locates at grid cell } n, \\ \mathbf{0} \in \mathbb{R}^{d\times 1}, \quad \text{otherwise.} \end{cases} \tag{3}$$

Then, the spatial-level attention weights associated with all vehicles at time step t, $B_t = \{\beta_t^1, \ldots, \beta_t^n, \ldots, \beta_t^N\}$, $B_t \in \mathbb{R}^{1\times N}$, are calculated as follows:

$$B_t = softmax(\tanh(W_\beta G_t)), W_\beta \in \mathbb{R}^{1\times d}, \tag{4}$$

where W_β is a collection of parameters to be learned.

Finally, we combine all historical information from the target and neighboring vehicles:

$$V_t = G_t(B_t)^T = \sum_{n=1}^{N} \beta_t^n G_t^n. \tag{5}$$

V_t is then fed into a feedforward network layer to predict the H-step trajectory of the target vehicle $\{\hat{X}_{t+1}^{target}, \ldots, \hat{X}_{t+H}^{target}\}$. The whole process along with the architecture of our STA-LSTM model is illustrated in Figure 1.

IV. Experiments

A. Data Introduction and Model Setup

STA-LSTM is learned and evaluated using the Next Generation Simulation (NGSIM) dataset [17]. The NGSIM dataset consists of vehicle trajectories from the segments of highway US-101 and highway I-80 in the United States. The US-101 segment has a length of 482 m (0.3 miles) and five lanes. The I-80 segment has a length of 644 m (0.4 miles) and six lanes. The data from either US-101 or I-80 contain vehicle trajectories sampled at 10 Hz for 45 minutes. Each 45-minute dataset consists of three 15-minute subsets, which are recorded over different time spans, respectively. This gives us in total six 15-minutes trajectory subsets for learning and testing STA-LSTM. Since these trajectory data are collected on highways, they only contain forward-moving and lane-changing trajectories. There are no other types of trajectories such as turning at intersections. We split each of the six 15-minute trajectory subsets into training, validation, and test datasets according to the ratio 0.7:0.1:0.2. As a result, the training dataset includes 5 922 867 records, the validation dataset has 859 769 records, and the test dataset has 1 505 756 records. No extra pre-processing such as normalization is conducted on the dataset.

In order to compare our model with the state-of-the-art CS-LSTM model from Deo and Trivedi [11], we follow the same data processing procedures as with theirs. First, we downsample each vehicle trajectory by a factor of 2. Second, based on vehicle local coordinates (x, y), where the y-axis represents the motion direction of the highway, and the x-axis is perpendicular to the y-axis, we discretize the space centered around the target vehicle as a 3×13 grid.

We choose the time step in this study to be 0.2 seconds. We take 15-step (i.e., 3 seconds, $T = 15$) historical trajectories of the target and its neighboring vehicles within the 3×13 grid as the input to STA-LSTM for predicting 5-step (i.e., 1 second, $H = 5$) future trajectory of the target vehicle.

The goal of STA-LSTM model is to minimize the following cost function:

$$\min \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} \sum_{j=1}^{H} (\hat{X}_{i,j} - X_{i,j})^2, \tag{6}$$

where N_{train} denotes the training set; $\hat{X}_{i,j} = [\hat{x}_{i,j}, \hat{y}_{i,j}]$ is the predicted position at the jth time step; and $X_{i,j} = [x_{i,j}, y_{i,j}]$ is the actual position.

The hyperparameters of STA-LSTM are optimized using a grid search. The dimension of the embedding space is set to 32. The dimension of the hidden vector of the LSTM model (i.e., d) is set to 64. The feedforward layer contains one hidden layer with dimension 128. The optimization method Adam [29] is chosen with the learning rate set to 0.001. The training epoch size is set to 10. All experiments are conducted using an Intel(R) Xeon(TM) W-2123 CPU, an Nvidia GTX 1080 GPU, and 32G RAM. The total training time of STA-LSTM is around 5 hours.

B. Prediction Accuracy Comparison

In order to evaluate STA-LSTM, we have implemented three benchmark models. The first model is CS-LSTM [11], which offers state-of-the-art performance on vehicle trajectory prediction. The second model is an LSTM model built solely using the target vehicle's historical trajectories. We refer to this model as *naïve LSTM*. The comparison to this model is to confirm whether the historical trajectories of neighboring vehicles (beyond the target vehicle information) can be used to improve the prediction accuracy. The third model is the LSTM model with only the spatial-level attention mechanism. The last hidden state from LSTM, which contains the most recent trajectory information [30], is selected to form the spatial-level attention layer. This layer fuses information from the target and neighboring vehicles. We refer to this model as *SA-LSTM*. The comparison to this model aims at identifying whether including temporal-level attention (in addition to spatial-level attention) will affect the prediction result. It takes roughly the same time as STA-LSTM (i.e., 5 hours) to train SA-LSTM and CS-LSTM. The training time of naïve LSTM is around 3 hours because it does not build LSTMs for neighboring vehicles. Note that due to the heterogeneous traffic in the data including both passenger vehicles and trucks, and missing physics-related data such as the friction coefficient of a road surface, it is impractical to implement physics-based models for comparison. So, in this study, we mainly focus on comparing our STA-LSTM with other data-driven deep learning models.

We measure the performance using the Root Mean Square Error (RMSE) between the predicted and actual positions of the target vehicle for 5 time steps into the future at 0.2 seconds per each step. The results are shown in Table I. STA-LSTM performs slightly better than CS-LSTM for the RMSEs across all time steps. SA-LSTM performs a little worse than STA-LSTM and CS-LSTM. The naïve LSTM, which relies solely on the information of the target vehicle, has the worst performance. These results indicate that 1) it is helpful to consider the information of neighboring vehicles for vehicle trajectory prediction; 2) It might be sufficient to

Models	RMSE per prediction time step				
	1	2	3	4	5
naïve LSTM	0.1012	0.2093	0.3384	0.4830	0.6406
SA-LSTM	0.1026	0.2031	0.3157	0.4367	0.5643
CS-LSTM [11]	0.1029	0.2023	0.3146	0.4364	0.5674
STA-LSTM (Ours)	0.0995	0.2002	0.3130	0.4348	0.5615

TABLE I: Comparison of our model and three benchmark models using Root Mean Square Error (RMSE). Each time step is 0.2 seconds; the longest prediction horizon is thus 1 second. SA-LSTM performs a little worse than STA-LSTM and CS-LSTM. The naïve LSTM has the worst performance. These results indicate that our model achieves performance comparable to other state-of-the-art models with the advantage of interpretable spatial-temporal attention weights.

use the most recent trajectories for prediction; 3) Computing spatial-temporal attention will not affect the prediction accuracy. Although our STA-LSTM model does not improve the prediction accuracy of CS-LSTM significantly, the learned spatial-temporal attention weights provide interpretability on the prediction results.

C. Attention Weights Analysis

1) Temporal-level Attention: We start by analyzing the temporal-level attention mechanism. We have computed the temporal-level attention weights of 15 historical time steps (from $t-14$ to t) using each of the six 15-minutes subsets. Figure 2 shows the averaged weights from $t-5$ to t. The weights before $t-5$ are omitted as they are very close to zero. The attention weights at the current time step t are the largest. This indicates that the future trajectory of the target vehicle is mainly influenced by the most recent trajectories of itself and the neighboring vehicles. This result also explains why SA-LSTM, which includes the spatial-level but not the temporal-level attention mechanism, performs only moderately worse than STA-LSTM.

2) Spatial-level Attention by Vehicle Class: We next analyze the spatial-level attention mechanism. For convenience, we label each grid cell by its lane name and relative order to the target vehicle's cell. For example, $(Current, 6)$ represents the 6th grid cell in the current lane and ahead of the target vehicle.

We conduct an analysis based on two main target vehicle types in the NGSIM dataset: *auto* and *truck*. The target vehicle's cell has the largest attention weight: for *auto* 72.14% and for *truck* 79.53%. This result, combined with the previous temporal-level attention analysis, reveals that the future trajectory of the target vehicle largely depends on its own driving status. The larger influence of a *truck* on itself may because of the truck needs a longer time to react to neighboring vehicles. So, its own trajectory plays a heavier role in trajectory prediction. In contrast, a car in the *auto* class is more flexible and can react faster to its neighboring vehicles by altering its trajectory.

In order to better show the distribution of attention weights of neighboring vehicles, we normalize and plot the rest of the attention in the *auto* class (27.86%) and in the *truck* class (20.47%) on the 3×13 grid. These results are shown in Figure 3 TOP and BOTTOM, respectively. The grid cells behind the target vehicle's cell receive virtually no attention weights, indicating the negligible influence of following vehicles on the target vehicle. This may be because drivers pay much less attention to following vehicles during driving.

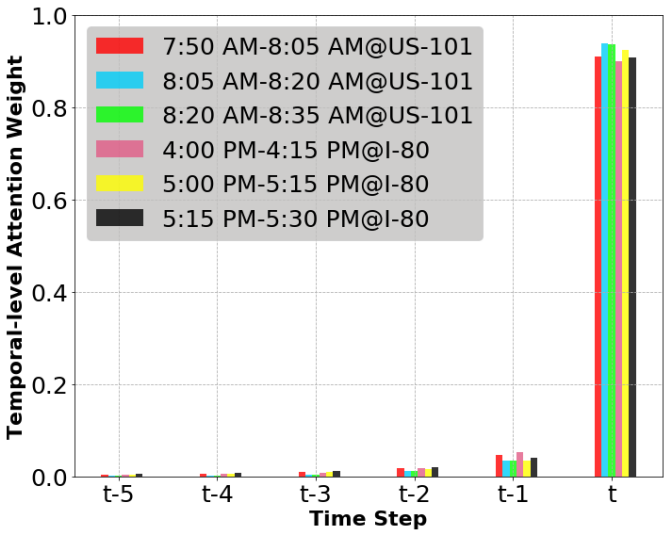


Fig. 2: Averaged temporal-level attention weights of 6 time steps computed using each of the six 15-minutes subsets. The weights before $t-5$ are omitted as they are negligible. The weights at the current time step t are the largest. This indicates that the future trajectory of the target vehicle is mainly impacted by the most recent trajectories of itself and its neighboring vehicles. In addition, this explains why by excluding temporal-level attention mechanism, the performance of SA-LSTM only drops moderately compared to STA-LSTM.

We further observe that when the target vehicle is an *auto*, all front grid cells on the current lane receive attention weights. The grid cells receiving larger values are $(Current, 2)$, $(Current, 3)$, and $(Current, 4)$. When the target vehicle is a *truck*, the larger weights are found at $(Current, 3)$, $(Current, 4)$, and $(Current, 5)$, while $(Current, 1)$ and $(Current, 2)$ receive less weight compared to the *auto* class. This may be because the truck usually keeps a longer distance to the front vehicle in order to maintain safety and subsequently pays more attention to front vehicles at a further distance.

3) Spatial-level Attention by Neighboring Vehicle Density: Since the NGSIM dataset records vehicle trajectories under different traffic conditions, it is possible to explore the influence of neighboring vehicle densities on the distribution of spatial-level attention weights.

Because the average number of neighboring vehicles within the 3×13 grid range computed using the NGSIM dataset is 7, we consider two neighboring vehicle densities: *less than or equal to 7* and *greater than 7*. The results are shown in Figure 4. When the number is more than 7, i.e., traffic is more congested, the weight of the target vehicle's cell decreases from 75% to 68%, showing the gain of the influence from neighboring vehicles. We also observe that among neighboring vehicles, when the number of neighboring vehicles is less

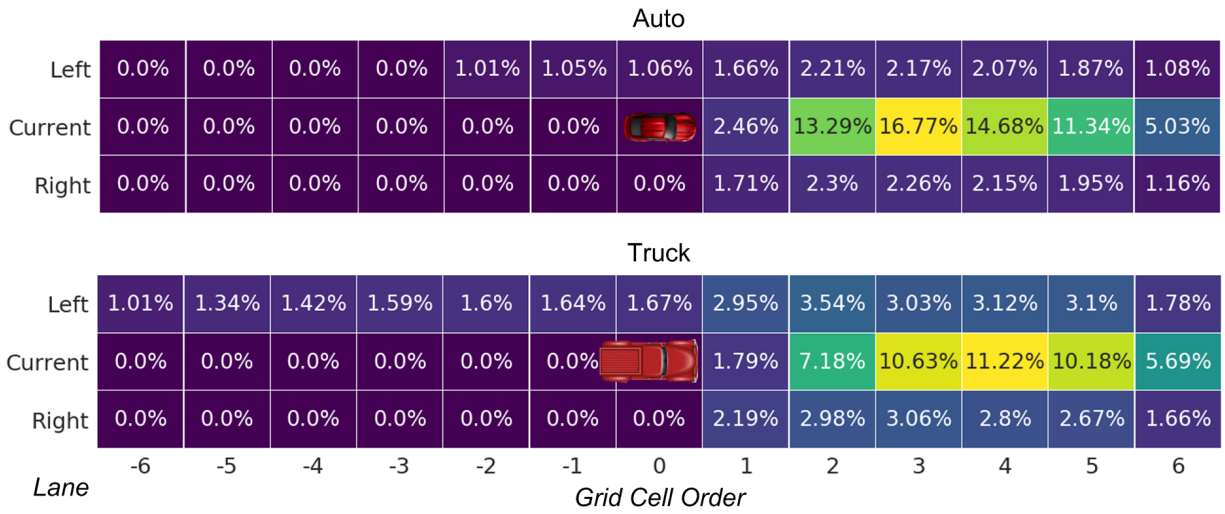


Fig. 3: Distributions of spatial-level attention weights by target vehicle class (excluding weights in the target vehicle's cell): *auto* TOP and *truck* BOTTOM. For all cases, cells behind the target vehicle's cell receive virtually no attention weights showing the negligible influence of vehicles in the back on the target vehicle. For *auto*, the largest weights appear at $(Current, 2)$, $(Current, 3)$, and $(Current, 4)$. For *truck*, the largest weights appear at $(Current, 3)$, $(Current, 4)$, and $(Current, 5)$, while $(Current, 1)$ and $(Current, 2)$ receive less weights compared to the *auto* class. This discrepancy may be because the truck often maintains a longer distance to the front vehicle for safety concerns thus focusing on front vehicles at a further distance. Note that we use vehicles' front-bumper positions to compute their belonging cells, and one vehicle only contributes to one cell.

than or equal to 7, the largest attention weight locates at $(Current, 4)$. In contrast, when the number is more than 7, $(Current, 2)$ has the largest attention weight. This may be because when congestion develops, the leading vehicle that the target vehicle reacts to is closer to the target vehicle.

4) Spatial-level Attention by Location: In the NGSIM dataset, the study segment of US-101 consists of five lanes, and the segment of I-80 contains six lanes. Each segment contains one additional ramp lane. These configurations allow us to analyze the distribution of the maximum spatial-level attention weight (of neighboring vehicles), when the target vehicle is on different lanes.

Here, we use the case of US-101 as an example. We select four lanes from US-101 southbound: the innermost lane, the middle lane, the outermost lane, and the ramp segment. These lanes are shown in Figure 5 LEFT. Figure 5 RIGHT shows grid cells with the frequency counts—each count indicates that one maximum spatial-level attention weight was assigned to this cell (which contains a neighboring vehicle).

As we can see from Figure 5 RIGHT, target vehicles mainly focus on front vehicles in the current lane. An exception is the ramp segment, where target vehicles pay more attention to front vehicles in the left lane indicating their intention to switch to it. Target vehicles in the outermost lane also pay more attention to the lane on the left compared to the lane on the right, showing the preference to change to the left than right. In contrast, a reverse pattern is found on the innermost lane, where target vehicles pay more attention to front vehicles in the right lane in addition to the current lane. Target vehicles in the middle lane, on the other hand, show smaller differences in attention distribution between the left lane and right lane. These results show that STA-LSTM can be used to capture driving attentions, including *stay in the same lane* and *switch lanes*. Next, we show that STA-LSTM can identify the moment when specific lane-changing behaviors are taken place.

5) Spatial-level Attention on Lane-changing Behaviors: In order to study whether spatial-level attention weights can explain specific driving behaviors such as lane-changing, we have selected as the studying subject the vehicle with ID 2858, which conducted two lane-changing maneuvers on I-80.

The target vehicle 2858 executed the first lane-changing maneuver around the 996th time step from lane 4 to lane 5 and the second lane-changing maneuver around the 1220th from lane 5 to lane 6. This is illustrated in Figure 6. In addition, we show the grid cells containing neighboring vehicles that receive the largest attention weight at each time step during this process in Figure 6. We observe that the target vehicle (i.e., vehicle 2858) mainly focused on front vehicles in the current lane for all the first 977 time steps. From the 978th to the 995th time step, it gradually relocated the maximum attention from the current lane to $(Right, 1)$ and then $(Right, 2)$, as it was preparing to change to the right lane.

In order to verify whether STA-LSTM has captured the lane-changing behaviors correctly with the learned attention weights, the trajectories of vehicle 2858 and its neighboring vehicles are examined. As a result, there was a neighboring vehicle with ID 2846 at $(Right, 0)$ at the 964th time step. No other neighboring vehicles were behind vehicle 2846 at the time. Vehicle 2858 slowed down from the 964th to the 967th time step, when the relative position of vehicle 2846 changed from $(Right, 0)$ to $(Right, 1)$. The speed of vehicle 2858 kept decreasing from the 968th to the 983th time step until the relative position of the neighboring vehicle 2846 had changed to $(Right, 2)$. The target vehicle 2858 then started increasing its speed while maintaining the relative position of vehicle 2846 at $(Right, 2)$, and finally changed to the right lane at

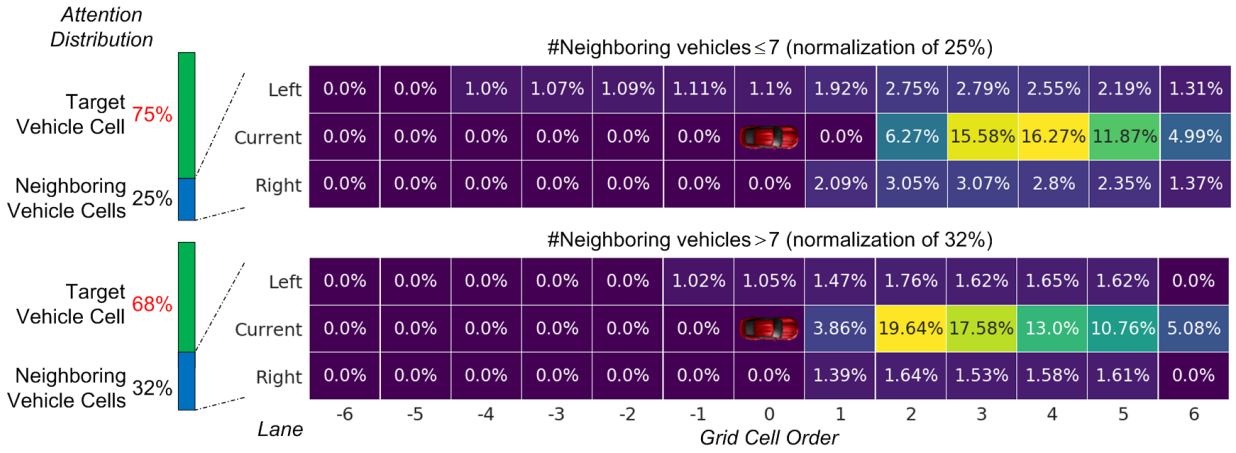


Fig. 4: Averaged spatial-level attention weights by the number of neighboring vehicles. TOP: the number of neighboring vehicles is less than or equal to 7, i.e., traffic is less congested. BOTTOM: the number of neighboring vehicles is greater than 7, i.e., traffic is more congested. As congestion develops (from TOP to BOTTOM), the attention weight of the target vehicle's cell decreases from 75% to 68%. This indicates that the target vehicle is affected more by neighboring vehicles in a congested traffic condition. By normalizing and plotting the attention weights of neighboring vehicles, we can see that the largest attention weight locates at $(Current, 4)$ on TOP and at $(Current, 2)$ on BOTTOM. This may be because the distance between the target vehicle and its leading vehicle is smaller in congestion.

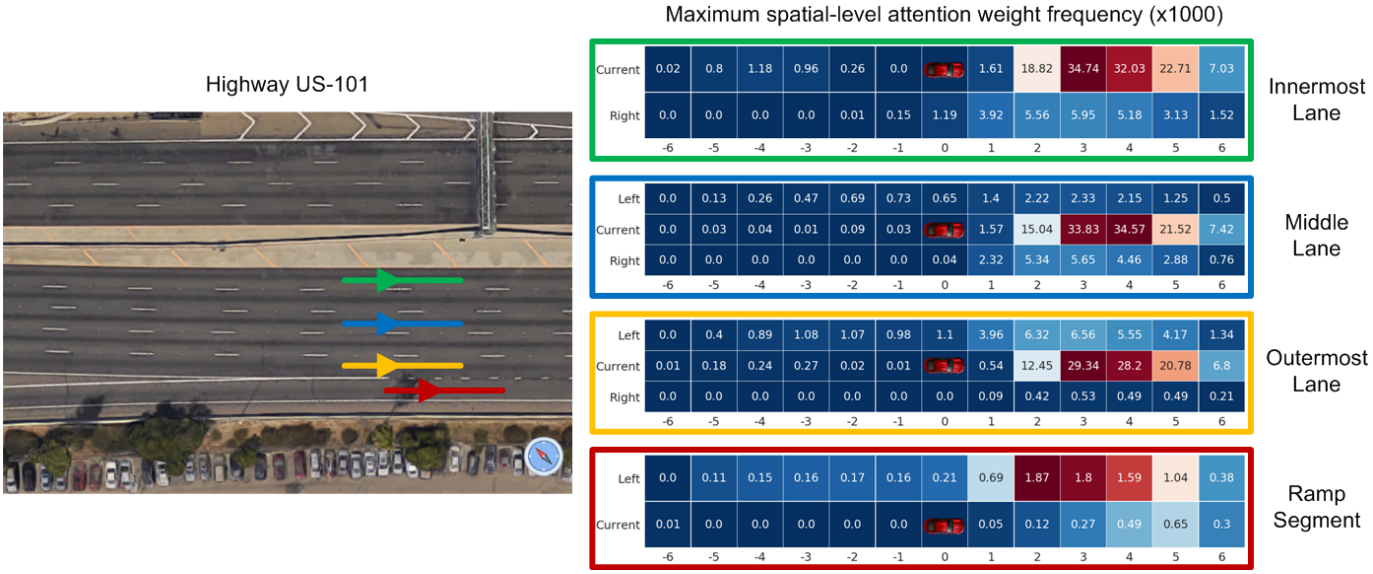


Fig. 5: Maximum spatial-level attention weight frequency by target vehicle location. Four lanes, namely the innermost lane, the middle lane, the outermost lane, and the ramp segment, are selected from US-101 southbound. Grid cells are filled with frequency counts that indicate where the maximum spatial-level attention weights located. Except for the ramp segment, target vehicles mainly focus on the current lane. On ramp segment, target vehicles pay more attention to the left lane showing their intention to switch to it. The frequency distribution on other lanes can be interpreted in a similar manner. These results demonstrate that STA-LSTM can capture various driving intentions, such as *stay in the same lane* and *switch lanes*.

the 996th time step. A similar pattern is observed during the second lane-changing maneuver. These results demonstrate that STA-LSTM is capable of capturing complex lane-changing maneuvers in detail, which can potentially benefit the development of simulation models and the motion planning and control algorithms for autonomous driving.

V. Conclusion

Vehicle trajectory prediction is an essential task in enabling many Intelligent Transportation System (ITS) applications. The importance of this task is emphasized with the emergence of autonomous vehicles, as they require an interpretable prediction of the future motions of surrounding dynamic agents (e.g., vehicles, pedestrians, and bicyclists) in order to navigate safely and efficiently. We have developed STA-LSTM that combines LSTMs with spatial-temporal attention mechanisms for vehicle trajectory prediction.

STA-LSTM is learned and evaluated using the NGSIM dataset [17], which contains real-world vehicle trajectories from the segments of highway US-101 and I-80 in the United States. Our experiment results show that STA-LSTM not only achieves performance comparable to other state-of-the-art

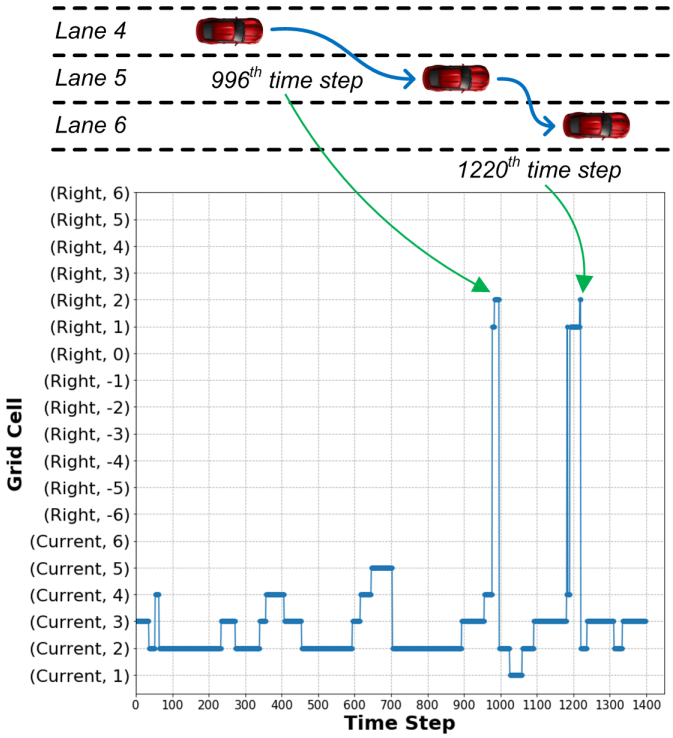


Fig. 6: Maximum spatial-level attention weights regrading the lane-changing behaviors of the target vehicle 2858. Two lane-changing maneuvers were executed: one at the 996th time step and the other at the 1220th time step. Prior to the first lane-changing, the target vehicle mainly focused on front vehicles in the current lane. During the two lane-changing executions to the right lane, the maximum attention of the target vehicle had switched to $(Right, 1)$ and $(Right, 2)$ (i.e., the two spikes are shown in the diagram). These results demonstrate that STA-LSTM can capture intricate driving behaviors in detail.

techniques in prediction accuracy, but more importantly, provides spatial-temporal attention weights for enhancing model explainability. The learned attention weights can be used to identify the influences of historical trajectories and locations of neighboring vehicles on the target vehicle's future motion. We have conducted detailed analyses of the learned attention weights based on various vehicle and environment factors, including target vehicle class, target vehicle locations, and neighboring vehicle densities. In addition, we have found that the learned attention weights can be used to interpret lane-changing behaviors of the target vehicle. Together, these analyses enable the in-depth study of the attention distribution of the target vehicle on itself and neighboring vehicles, thus can potentially benefit the development of many ITS applications such as advanced driver assistance, and autonomous vehicles' motion planning and navigation.

A. Future Work

Many future research directions can stem from this work. Instead of using grid-based discretization to model the relationship between the target vehicle and neighboring vehicles, other data structures can be explored. For example, a graph, in which nodes representing vehicles and edges representing the influences among vehicles, can be used to replace the grid. Therefore, it would be interesting to test whether a graph-based deep learning technique such as the graph convolutional neural network [31] can be used to capture the correlations among vehicles and predict vehicle trajectories.

The data used to learn STA-LSTM are from stationary sensors installed on US-101 and I-80. While these sensors provide complete and accurate traffic measurements, they are mostly found on highways and major roads, which only constitute a small portion of a city. In order to use our approach for autonomous driving on arterial roads, which constitute the majority of a city, we need to work with mobile data such as GPS reports. Given GPS data can be either sparsely or densely sampled, it would be interesting to combine the previous techniques for addressing sparse GPS data [32] or dense GPS data [33], [34] with STA-LSTM to make it more applicable for developing autonomous driving on arterial roads.

STA-LSTM can also be used to enhance traffic simulation models. Realistic virtual traffic as a result of an improved simulation technique has many applications in 1) ITS such as studying congestion causes, identifying network bottlenecks, and testing transport policies at the macroscopic scale [35], [36], [37]; and 2) Virtual Environments such as improving the believability of traffic animation and reconstruction [38], [39], and enhancing the training and testing of autonomous vehicles at the microscopic scale [40]. It would be of great use to develop simulation models that incorporate STA-LSTM.

REFERENCES

[1] G. Seetharaman, A. Lakhotia, and E. P. Blasch, "Unmanned vehicles come of age: The DARPA grand challenge," *Computer*, vol. 39, no. 12, pp. 26–29, 2006.
[2] D. Pomerleau, "ALVINN: An autonomous land vehicle in a neural network," in *Advances in neural information processing systems*, 1989, pp. 305–313.
[3] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp, "Off-road obstacle avoidance through end-to-end learning," in *Advances in neural information processing systems*, 2005, pp. 739–746.
[4] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.
[5] W. Li, D. Wolinski, and M. C. Lin, "ADAPS: Autonomous driving via principled simulations," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7625–7631.
[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
[7] Y. Shen, W. Li, and M. C. Lin, "Autonomous driving via multi-sensor perception and weighted inverse reinforcement learning," *Technical Report, University of Maryland, College Park*, 2020.
[8] X. Wang, R. Jiang, L. Li, Y. Lin, X. Zheng, and F.-Y. Wang, "Capturing car-following behaviors by deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 910–920, 2017.
[9] M. Zhou, X. Qu, and X. Li, "A recurrent neural network based microscopic car following model to predict traffic oscillation," *Transportation research part C: emerging technologies*, vol. 84, pp. 245–264, 2017.
[10] X. Huang, J. Sun, and J. Sun, "A car-following model considering asymmetric driving behavior based on long short-term memory neural networks," *Transportation research part C: emerging technologies*, vol. 95, pp. 346–362, 2018.
[11] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1468–1476.
[12] F. Altché and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *20th International Conference on Intelligent Transportation Systems*, 2017, pp. 353–359.

[13] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[14] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, 2016, pp. 207–212.

[15] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.

[16] L. Lin, B. Xu, W. Wu, T. Richardson, and E. A. Bernal, "Medical time series classification with hierarchical attention-based temporal convolutional networks: A case study of myotonic dystrophy diagnosis," *arXiv preprint arXiv:1903.11748*, 2019.

[17] "Next generation simulation (NGSIM)," https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm.

[18] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH journal*, vol. 1, no. 1, 2014.

[19] H. M. Mandalia and M. D. D. Salvucci, "Using support vector machines for lane-change detection," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 49, no. 22, 2005, pp. 1965–1969.

[20] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *17th International IEEE Conference on Intelligent Transportation Systems*, 2014, pp. 334–341.

[21] R. S. Tomar and S. Verma, "Safety of lane change maneuver through a priori prediction of trajectory using neural networks." *Network Protocols & Algorithms*, vol. 4, no. 1, pp. 4–21, 2012.

[22] T. Gindele, S. Brechtel, and R. Dillmann, "A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments," in *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 1625–1631.

[23] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmàn, "Intention-aware risk estimation for general traffic situations, and application to intersection safety," Ph.D. dissertation, INRIA, 2013.

[24] L. Lin, S. Gong, T. Li, and S. Peeta, "Deep learning-based human-driven vehicle trajectory prediction and its application for platoon control of connected and autonomous vehicles," in *The Autonomous Vehicles Symposium*, vol. 2018, 2018.

[25] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 336–345.

[26] B. Kim, C. M. Kang, S. H. Lee, H. Chae, J. Kim, C. C. Chung, and J. W. Choi, "Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network," *arXiv preprint arXiv:1704.07049*, 2017.

[27] T. Fernando, S. Denman, S. Sridharan, and C. Fookes, "Soft+hardwired attention: An LSTM framework for human trajectory prediction and abnormal event detection," *Neural networks*, vol. 108, pp. 466–478, 2018.

[28] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "SR-LSTM: State refinement for lstm towards pedestrian trajectory prediction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 085–12 094.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[30] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 1480–1489.

[31] L. Lin, Z. He, and S. Peeta, "Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 258–276, 2018.

[32] W. Li, D. Nie, D. Wilkie, and M. C. Lin, "Citywide estimation of traffic dynamics via sparse GPS traces," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 3, pp. 100–113, 2017.

[33] L. Lin, W. Li, and S. Peeta, "Efficient data collection and accurate travel time estimation in a connected vehicle environment via real-time compressive sensing," *Journal of Big Data Analytics in Transportation*, vol. 1, no. 2, pp. 95–107, 2019.

[34] L. Lin, S. Pccta, and J. Wang, "Efficient collection of connected vehicle data based on compressive sensing," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 3427–3432.

[35] D. Wilkie, J. Sewall, W. Li, and M. C. Lin, "Virtualized traffic at metropolitan scales," *Frontiers in Robotics and AI*, vol. 2, p. 11, 2015.

[36] W. Li, D. Wolinski, and M. C. Lin, "City-scale traffic animation using statistical learning and metamodel-based optimization," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 200:1–200:12, Nov. 2017.

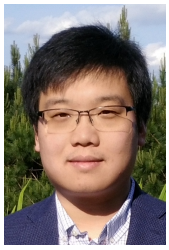
[37] W. Li, M. Jiang, Y. Chen, and M. C. Lin, "Estimating urban traffic states using iterative refinement and wardrop equilibria," *IET Intelligent Transport Systems*, vol. 12, no. 8, pp. 875–883, 2018.

[38] H. Bi, T. Mao, Z. Wang, and Z. Deng, "A data-driven model for lane-changing in traffic simulation." in *Symposium on Computer Animation*, 2016, pp. 149–158.

[39] Q. Chao, Z. Deng, J. Ren, Q. Ye, and X. Jin, "Realistic data-driven traffic flow animation using texture synthesis," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 2, pp. 1167–1178, 2017.

[40] Q. Chao, H. Bi, W. Li, T. Mao, Z. Wang, M. C. Lin, and Z. Deng, "A survey on visual traffic simulation: Models, evaluations, and applications in autonomous driving," *Computer Graphics Forum*, vol. 39, no. 1, pp. 287–308, 2019.

Lei Lin received the B.S. degree in Traffic and Transportation and the M.S. degree in System Engineering from Beijing Jiaotong University, China, in 2008 and 2010. He received the M.S. degree in Computer Science and the Ph.D. degree in Transportation Systems Engineering from the University at Buffalo, the State University of New York, Buffalo, in 2013 and 2015. He is now a Research Scientist at Goergen Institute for Data Science, University of Rochester. His research interests include Transportation Big Data, Artificial Intelligence Applications in Transportation, and Connected and Automated Transportation.



Weizi Li earned his Ph.D. from University of North Carolina at Chapel Hill, M.S. from George Mason University, and B.Eng. from Xiangtan University (China), all in Computer Science. Currently, he is a Michael Hammer Postdoctoral Fellow at the Institute for Data, Systems, and Society (IDSS) of Massachusetts Institute of Technology (MIT). His current research interests include Multi-agent Simulation, Intelligent Transportation Systems, Robotics, and Machine Learning.

Huikun Bi received her Ph.D. degree from University of Chinese Academy of Sciences. She is an assistant professor with Institute of Computing Technology, Chinese Academy of Sciences, China. Her main research interests include crowd simulation, motion forecasting, and deep learning.

Lingqiao Qin received the B.S. in Transportation Engineering from the Beijing Jiaotong University, Beijing, China, the M.S. in Transportation Safety Engineering from the George Washington University, Washington, D.C., US, and the M.S. in Industrial and Systems Engineering from University of Wisconsin-Madison, WI, US. Lingqiao is currently working toward her Ph.D. degree in Transportation Engineering at the University of Wisconsin-Madison, WI, US. Her research interests include traffic operations, the next generation of transportation (autonomous and connected vehicles), and using advanced technologies such as driving simulator and eye trackers to improve the design, operations, and safety of all elements in transportation.