GISNet:Graph-Based Information Sharing Network For Vehicle Trajectory Prediction

Ziyi Zhao, Haowen Fang, Zhao Jin, Qinru Qiu

Department of Engineering and Computer Science, Syracuse University, Syracuse, New York, USA

Email: {zzhao37, hfang02, zjin04, qiqiu}@syr.edu

Abstract—The trajectory prediction is a critical and challenging problem in the design of an autonomous driving system. Many AI-oriented companies, such as Google Waymo, Uber and DiDi, are investigating more accurate vehicle trajectory prediction algorithms. However, the prediction performance is governed by lots of entangled factors, such as the stochastic behaviors of surrounding vehicles, historical information of self-trajectory, and relative positions of neighbors, etc. In this paper, we propose a novel graph-based information sharing network (GISNet) that allows the information sharing between the target vehicle and its surrounding vehicles. Meanwhile, the model encodes the historical trajectory information of all the vehicles in the scene. Experiments are carried out on the public NGSIM US-101 and I-80 Dataset and the prediction performance is measured by the Root Mean Square Error (RMSE). The quantitative and qualitative experimental results show that our model significantly improves the trajectory prediction accuracy, by up to 50.00%, compared to existing models.

Index Terms—GNN, Information Sharing, ADS, Vehicle Trajectory Prediction

Fig. 1: Trajectory Prediction Example

I. INTRODUCTION

Although the autonomous driving car is fully committed to liberating human from the boring driving activities, its safety and efficiency are still the primary concerns. In real road traffic, the number of different possible trajectories a car can take in just a few seconds may be countless. Figure 1 illustrates some examples. A precise trajectory prediction helps the autonomous car take the correct action in next stage.

Trajectory prediction is a challenging problem, because it does not only depend on the historical information of the target vehicle, but also the historical information of the surrounding vehicles. In recent years, lots of intricate problems become solvable due to the growing application of the learning in many fields, such as computer vision [1] [2] [3], natural language processing [4], intelligence hardware [5] [6], etc. Many papers have been published to improve trajectory prediction [7] [8] [9]. Despite the efforts, the accuracy of the existing prediction models is not high enough. This is because the surrounding vehicles will also respond to its environment and adjust its trajectory accordingly. Without considering this, the model can not make an accurate trajectory prediction. It is necessary to have information from all neighboring vehicles, and consider the potential evolvements of their trajectories in the near future. For example, the human drivers observe and surmise other drivers' latent intention from the mirrors of the car. To emulate this behavior, an information sharing network should be established among all vehicles.
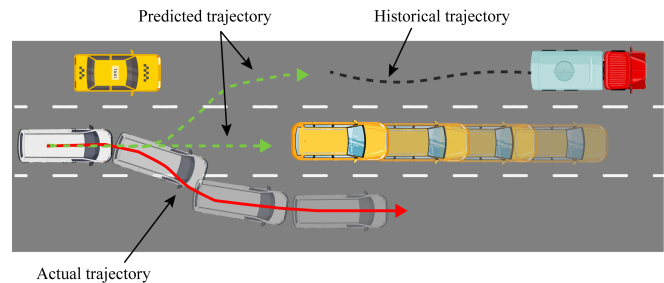
In the rest of the paper, we use the name "*target vehicle*" to refer to the vehicle whose trajectory is to be predicted and use the name "*neighbors*" to refer to the surrounding vehicles. We propose a novel graph-based information sharing network (GISNet), which allows the target and neighbor vehicles to propagate and communicate the trajectory features among themselves. Our proposed network is evaluated using the NGSIM highway vehicle trajectory dataset. The RMSE of the prediction is compared to several existing models. Compared with other existing trajectory prediction methods, our approach can reduce the prediction error by up to 50.00%. The following summarizes the major contributions of our work:

- A new trajectory prediction model is developed, which allows information sharing among vehicles using a graph neural network.
- The prediction is based on the embedding feature, which is derived from multi-dimensional input sequences including the historical trajectory of target and neighboring vehicles, and their relative social positions.
- The model allows us to consider the latent intention of surrounding neighbors during the prediction. Compared with other existing trajectory prediction methods, our approach can reduce the prediction error by up to 50.00% and achieve the state-of-the-art performance.

The rest of the paper is structured as follows: In Section II, we review the existing methods, from which we got the inspirations. This is followed by details of our GISNet in Section III. Section IV and Section V describe our experiment setup and report the results. Finally, Section VI concludes this work and discusses possible future extensions.

II. RELATED WORKS

Over the past several years, autonomous driving has become increasingly critical [10] [11] [12]. Lots of researchers dedicated their attention to accurate vehicle trajectory prediction [13] [14]. Due to space limitation, in this section, we focus on the pros and cons of some most recent works in this area.

The Long Short-term Memory (LSTM) network is an effective model to memorize the trajectory of vehicles. Several works took the LSTM as the backbone network for the trajectory prediction [15] [16] [17] [18]. In [19], the authors proposed to use a convolutional neural network (CNN) to replace the LSTM. Their work aimed at increasing the model parallelism and efficiency. It shows that, without the LSTM, the efficiency of the model can be improved significantly. However, the input of the model is still formulated using the sequence-to-sequence format. The historical trajectory input are embedded into a fixed size of vector through the fully connected (FC) layers. In order to preserve the temporal information, the historical trajectory data are stacked together based on their time order.

Recently, many researchers started to investigate the relationship between the target vehicle and its surrounding neighbors [20] [21]. Hand-crafted features were integrated into the model for trajectory prediction [22] [23] [24]. Nonetheless, the performance of the motion prediction highly depends on the quality of the hand-crafted features. The method of social pooling was first proposed in [13]. The interactions among all individuals can be shared between multiple LSTMs through the social pooling layer. As an extension of the original social pooling, the convolution operation was introduced into the model in [25]. The LSTM layer encodes the historical trajectory of each vehicle into a feature vector. Each encoded trajectory feature was put into the corresponding location in a 3D tensor, which is the same as its location in the background scene. Finally, all the features were constructed as a 3D tensor. Therefore, the reception field in the convolution operation can explore the interaction between different objects. The non-local multi-head attention mechanism was invented to combine the relevant neighbor information [26]. The model divides the road environment into grids. The learned attention weight specifies the amount of attentions that need be placed on the trajectory features associated to each grid during the prediction. Instead of considering the interactions among neighboring objects, [27] considers the relationship between the object and its scene background. The authors concatenated the multi-agent encoding and the scene context encoding as the input of the trajectory prediction network. The predicted trajectory was regulated by the constraints which was learnt from the scene background.

In this work, we focus on enhancing the information sharing between vehicles. By adopting Graph Convolutional Network (GCN), the vehicles can learn the latent intention of its surrounding neighbors. The experimental results show that our proposed approach achieves the state-of-the-art performance.

III. METHODS

Our proposed network is an end-to-end model. Each component module is fully differentiable. The loss is calculated by measuring the difference between the predicted trajectories and the ground truth trajectories. In this section, each component of the network will be elaborated.

A. Historical Trajectory Formulation

The historical trajectory of each vehicle is formulated as a sequence:

$$X_{coor} = \{C^{t-1}, C^{t-2}, ..., C^t\} \qquad (1)$$

where,

$$C^t = \{x^t, y^t\} \qquad (2)$$

is the collection of the historical trajectory coordinates which contain x, y values. The t is the time horizon of the historical trajectory, which is set to be 3 in this paper. The unit of t is "second".

B. Vehicle Information Embedding

The LSTM model has the ability to memorizing the long term dependency from the past information. So, it can be used for extracting the features from the vehicle's historical trajectories, as shown in the left part of Figure 2. The LSTMs, which are used for extracting the embedding trajectory features from all vehicles, share the same weight. In this way, the hidden states of all vehicles have consistent representations. For each vehicle i, a 1-d embedding vector x_{traj} with size l is extracted by the LSTM, which captures the trajectory features of the vehicle. It will be placed into a $m * n$ grid system to from a 3D tensor. The placement position is determined by the grid location of the vehicle in the scene background. Hence, the relative positional relationship of each vehicle can be preserved. In this paper, the m and n are defined as 13 and 3 following the setting in [25]. Those grids that do not have a vehicle will be filled with zeros. The tensor is processed by a convolutional layers followed by pooling layers to extract the 1-d feature vector, x_{social}. It contains the social relationships among the vehicles, as shown in the middle part of Figure 2.

At the same time, we keep a separate copy of the trajectory feature of the target vehicle, x_{traj}, as shown at the bottom of Figure 2. It will be integrated with the x_{social} later in the prediction stage.

C. Graph-based Information Sharing

After the LSTM encoding, the embedding collection of all vehicles' historical trajectory is generated, which is denoted as:

$$X_{vehicles} = \{x_{traj}^1, x_{traj}^2, ..., x_{traj}^n\} \qquad (3)$$

where n is the number of vehicles that can be observed in the current scene. For better prediction, the target vehicle needs to learn the latent intention from its surrounding neighbors. However, the structure of the information sharing network are generated from the non-euclidean domain.

In this paper, we employ the spectral graph convolutional operation to allow the information sharing among all vehicles.
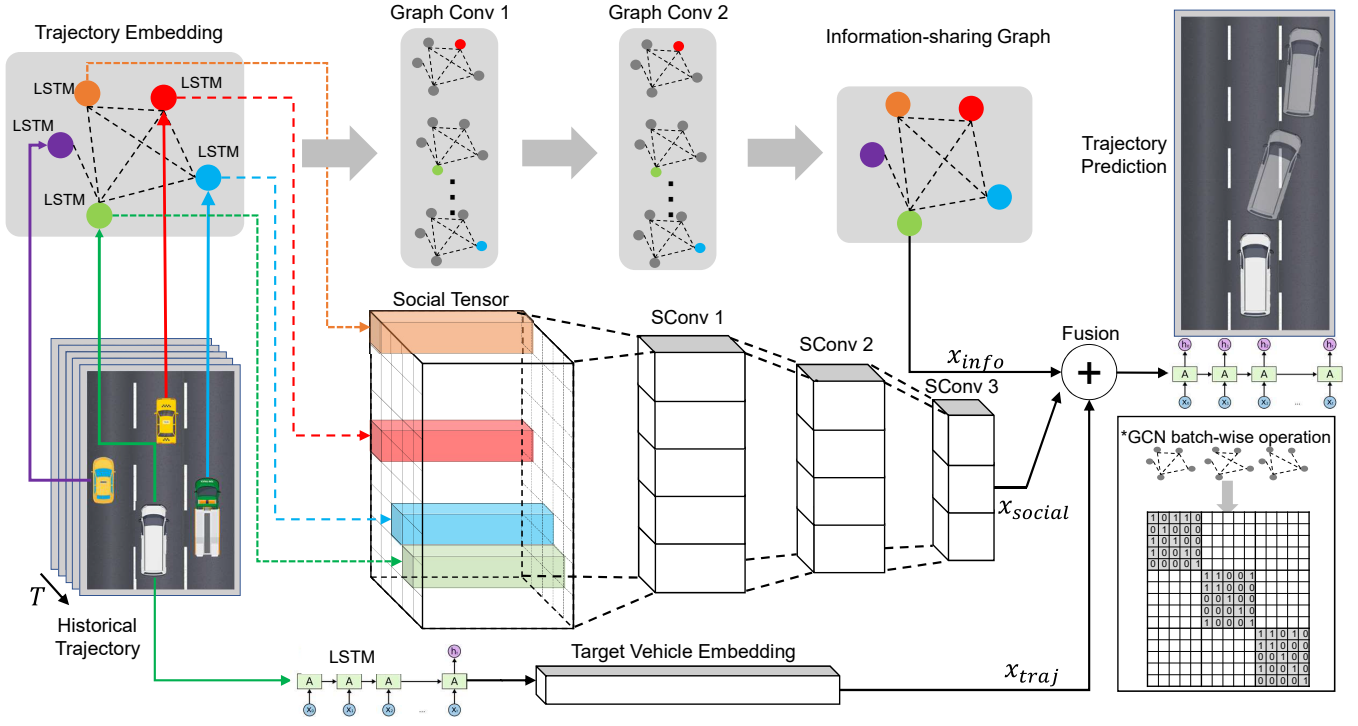
Fig. 2: Graph-based Information Sharing Network (GISNet) Architecture

The spectral graph convolution is described as the multiplication of a graph signal with the filter in the Fourier space. The graph Fourier transform is characterized as the multiplication of a graph signal with the eigenvector matrix of the graph Laplacian. Compared with the traditional CNN model, the graph convolutional network has its talent in exploring the meaningful features from the irregular structures [28]. The equation of information propagation between layers in the Graph Convolution Network (GCN) is defined as following:

$$H^{(l+1)} = \sigma(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \qquad (4)$$

where,

$$\hat{A} = A + I \qquad (5)$$

$\hat{A}$ is the matrix contains the adjacency matrix A and the identity matrix I. The $\hat{A}$ matrix has two objectives: 1. allow the information sharing between the node and its adjacent neighbors. 2. each node can consider the lower level feature from itself. The $\hat{D}$ is a matrix that describes the degree of each node. The feature of each node at layer l is defined as $H^{(l)}$. For the input layer, the H is equal to X. The σ is the activation function to improve the representability of the model. The GISnet architecture applies a two-layer graph convolutional network (GCN). The feature size in both layers is set to 64. The $\hat{A}$ is a zero-one matrix. Its ijth entry $(\hat{a_{ij}})$ is zero if there is no connection between vertices i and j, otherwise it is 1. In our case, the vehicle which is being predicted is connected

with all surrounding vehicles. The forward path of the model is defined as following:

$$f(X, A) = (\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}ReLU(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}XW^{(0)})W^{(1)})$$
$$(6)$$

where the $ReLU$ is the activation function between layers. The $W^{(0)}$ and $W^{(1)}$ are the parameters within the two graph convolution layers. The architecture of the information sharing module is given in the top part of Figure 2. After the two layer convolution operations, a 1-d feature x_{info} that summarizes the latent intention of surrounding neighbors is acquired.

In order to improve the training efficiency, we employ the batch-wise operation for multi graph in the GCN, as shown in the bottom right of Figure 2. First, all graphs are concatenated together to build a fusion graph. Then, a fusion block diagonal matrix is established. Each one of them represents the connectivity of the graph instance.

D. Future Trajectory Generation

Finally, all three features, $\{x_{traj}, x_{social}, x_{info}\}$, are concatenated together to construct an embedding of the vehicle future trajectory, which is to be predicted. Then, the embedding feature is passed to the trajectory generation module, as shown in the top right of Figure 2. In the generation module, the LSTM based decoder is applied to generate the sequence of x, y coordinates for the next 5 cycles, which is the prediction horizon that we are interested. The output of the model is denoted as the following sequence:

$$Y_{coor} = \{C^{t-1}, C^{t-2}, ..., C^t\} \qquad (7)$$

where,

$$C^t = \{x^t, y^t\} \tag{8}$$

is the predicted x, y values of the target vehicle at time t.

IV. EXPERIMENTS

A. Dataset

Our proposed model is evaluated on the public Next Generation Simulation (NGSIM) dataset. The NGSIM dataset collects the detailed vehicle trajectory information on eastbound I-80 in the San Francisco Bay area [29] and southbound US 101 in Los Angeles [30]. The study area for I-80 and 101 are 500 meters (1,640 feet) and 640 meters (2,100 feet), respectively. Figure 3 shows the study area of the NGSIM dataset. All the data are segmented into three 15 minute periods. The dataset is splitted into three subsets: training, validation and testing. We follow the same approach in [25] to split the vehicle trajectories into 8 second segments. The first 3 seconds are treated as the historical data, and the following 5 seconds trajectories are to be predicted.
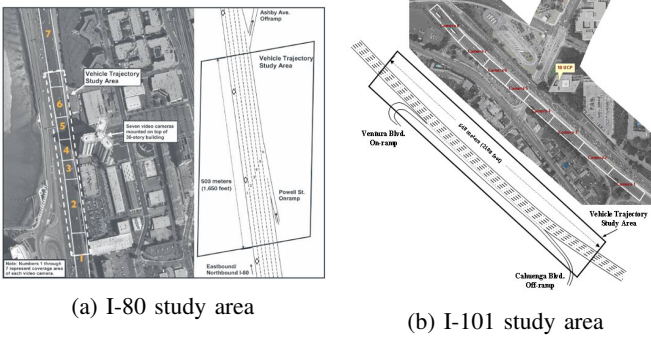

(a) I-80 study area

(b) I-101 study area

Fig. 3: Data Collection Procedure

B. Evaluation Metrics

In this paper, we use the Root Mean Square Error (RMSE) [31] to evaluate the performance of our proposed approach. This metric measures the difference between the predicted trajectory and the observed trajectory (ground truth) using the following equations:

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{m=1}^{n}(x_m^T - {x'_m}^T)^2 + (y_m^T - {y'_m}^T)^2} \tag{9}$$

where x_m^t and y_m^t are the predicted coordinates. The m is the sample index. The total number of testing sample is denoted as n, and ${x'_m}^t$ and ${y'_m}^t$ are the ground truth coordinates. T is the prediction horizon of the model. In our experiment, T is varying from 1 to 5 seconds.

C. Comparison Baselines

- Constant Velocity (CV) [32]: A baseline method that uses the constant velocity (CV) Kalman filter to forecast vehicle trajectory .
- GAIL-GRU [33]: A generative adversarial imitation learning model that takes the ground truth trajectories

of all adjacent neighbors as the model input. The GRU is like a long short-term memory (LSTM) but has less network parameters than LSTM.
- Vanilla LSTM (V-LSTM): The typical LSTM based encoder-decoder model. The vehicle historical trajectory is fed into the model as the input. Then, the LSTM output is decoded as the vehicle trajectory prediction.
- Social-LSTM (S-LSTM) [34]: The model applies the social pooling layer, which allows the information sharing between each individual LSTM.
- ConvSocial-LSTM (CS-LSTM) [25]: The model uses the convolution operation to extract the features from the social tensor. The prediction-centric vehicle's feature is concatenated with social feature.
- Non-local Social Pooling (NLS-LSTM) [26]: This model is based on an LSTM encoder-decoder. The social pooling is applied to capture the interactions between all vehicles. Besides, non-local multi-head attention mechanism is used to summarize the relevant information.
- Multi-Agent Tensor Fusion (MATS) [27]: This model concatenates the background scene feature and the vehicle historical trajectory feature into a multiagent tensor. A generative adversarial networks (GAN) based module is included for generating the future trajectory prediction.

D. Experiment Setup

We run our experiments on a desktop server running Ubuntu 16.04 OS with 3.60GHz Intel Xeon W-2123 CPU, 256GB Memory and a NVIDIA 2080Ti GPU. During the training, the Adam optimizer is applied with a 0.001 learning rate. The graph-based information sharing model has a 64 dimensional embedding state. We use the ReLU to be the activation function. Batch normalization and dropout are also applied for preventing the overfitting. The training and testing framework is built in PyTorch. The Dataset split method is the same as [25].

V. RESULTS

A. Predicted Trajectory Accuracy Improvement

Table I shows the RMSE results for the models being compared. The first thing we can notice is that all the deep learning based methods outperform the traditional model (CV). It demonstrates the effectiveness of the deep learning based model. The vanilla LSTM considers the temporal trajectory of the target vehicle. And the generative adversarial imitation learning - gated recurrent unit (GAIL-GRU) extends the lstm architecture by importing the GAN. However, none of them consider the impact of the neighbor cars. Hence, they also perform poorly in the prediction.

The second thing we can observe is that, all the models which consider the surrounding vehicles give the lower RMSE. It proves that the neighbors information does help the vehicle trajectory prediction. Moreover, our proposed method outperforms all other baselines due to the information-sharing mechanism. Compared with the original convolutional social pooling (CS-LSTM) method, we can achieve 45.35%, 34.02%,

TABLE I: The evaluation of RMSE in meters on NGSIM dataset

Horizon (s)	CV	GAIL-GRU	V-LSTM	S-LSTM	CS-LSTM	NLS-LSTM	MATS	GISNet
1	0.73	0.69	0.68	0.65	0.61	0.56	0.66	**0.33**
2	1.78	1.51	1.65	1.31	1.27	1.22	1.34	**0.83**
3	3.13	2.55	2.91	2.16	2.09	2.02	2.08	**1.42**
4	4.78	3.65	4.46	3.25	3.10	3.03	2.97	**2.14**
5	6.68	4.71	6.27	4.55	4.37	4.30	4.13	**3.23**

(a) Fast Speed Traffic

(b) Low Speed Traffic
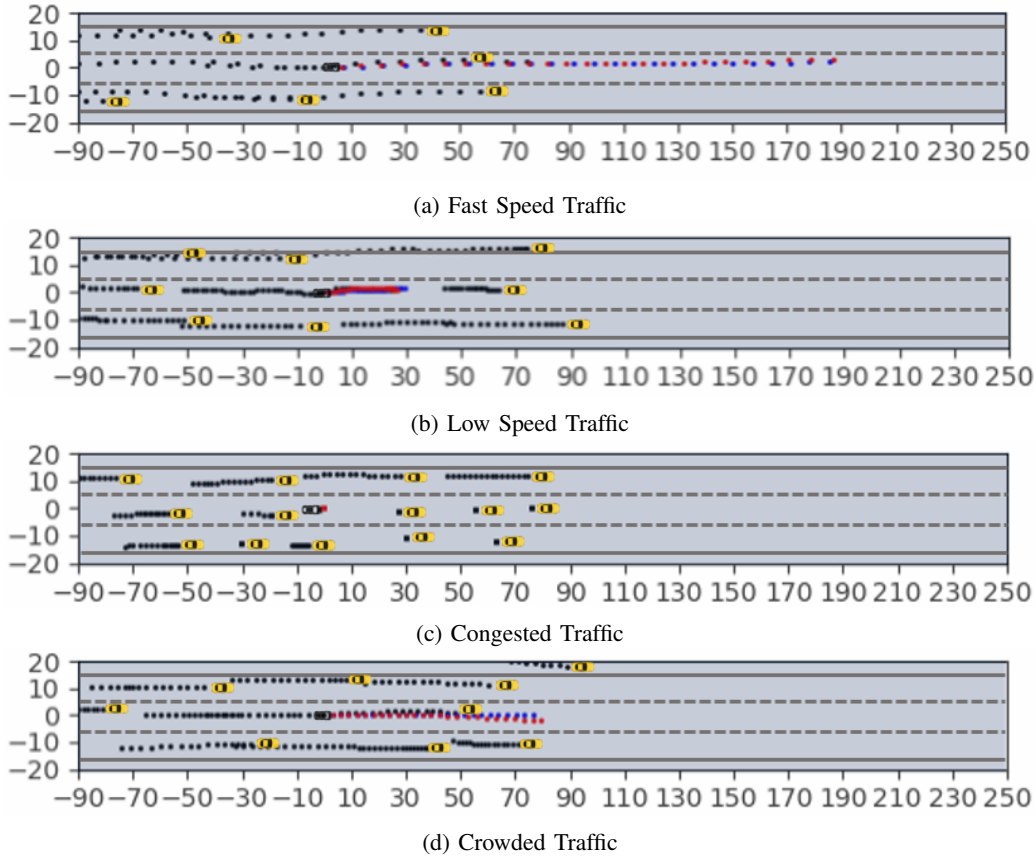

(c) Congested Traffic

(d) Crowded Traffic

Fig. 4: Trajectory Prediction Visualization. Black vehicle: the car which is being predicted. Yellow vehicles: surrounding neighbors. Black dash lines: historical trajectory. Blue dash line: predicted trajectory. Red dash line: ground truth trajectory.

31.69%, 30.74% and 25.86% accuracy improvements when prediction horizon varies from 1s to 5s. Compared with the MATS, our model also achieves 50.00%, 38.06%, 31.73%, 27.95% and 21.79% accuracy improvements for the 5 prediction horizons. Although CS-LSTM and MATS both considers the features from neighboring vehicles, they were placed in the social tensor and processed by a convolutional neural network. The GISNet outperforms MATS and CS-LSTM due to two reasons: 1) Small CNN kernels are used in these model, therefore, they only consider the joint features of vehicles in adjacent area. Although the covered area of the joint features increases as the network goes deeper, the resolution of the information is also reduced due to the pooling layers. 2) The social tensor does not only have the useful features of the neighboring vehicles, but also has lots of empty features located at the grid location not occupied by any vehicles.

Finally, compared with the NLS-LSTM, the proposed model reduces the RMSE by 41.07%, 31.97%, 29.70%, 29.37% and 24.88% in different prediction horizons, respectively.

B. Vehicle Predicted Trajectory Visualization

In this section, we visualize several predicted trajectories and the ground truth to give a qualitative demonstration of the prediction performance. All the results are sampled from the NGSIM data set. And the data in the NGSIM is collected from the real world. We select 4 different scenarios to reproduce some typical scenes in daily-life: a) Fast speed traffic, b) Low speed traffic, c) Congested traffic, d) Crowded Traffic. The results are given in Figure 4. In the figure, the black vehicle is the car which is being predicted, and the yellow vehicles are the surrounding neighbors. The black dash lines are the historical trajectory of each vehicle. The blue dash line is the

trajectory predicted by model, and the red dash line is the ground truth trajectory.

As we can see that, our predicted trajectories are close to the ground truth. In the high speed traffic scenario, the cars are driving at a relatively high speed. The final location of the predicted trajectory is almost the same as the observed location, as shown in Figure 4.a. In the low speed traffic scenario (Figure 4.b), the GISNet learns that the vehicle is in a relative low speed, and the predicted trajectory is shorter. For the congested traffic scenario, the result is given in Figure 4.c. We can see that all vehicles on the left most lane are moving, however, the car which is being predicted is in a congested lane. Consequently, the model is not affect by the surrounding cars and can predict the stationary trajectory. Last, the most complex situation in daily life is the crowded traffic, where the target vehicle is moving but crowded with many cars. Figure 4.d shows the predicted result in this scenario, where the vehicle is trying to make a lane change. In this scenario, our model can still predict the motion. In general, our model can output an accurate car location for the near future (1s, 2s and 3s) and make good prediction of the trend of the trajectory for the longer term (4s and 5s).

VI. CONCLUSION AND FUTURE WORKS

In this paper, we propose a novel graph-based information sharing network (GISNet). The network has the ability to encode the historical trajectory of each vehicle and allow the information sharing between the target vehicle and its surrounding neighbors. Furthermore, the model can fuse the features extracted from both Euclidean domain and non-Eculidan domain to make the future trajectory prediction. We apply our network to a public dataset to demonstrate its capability to predict an accurate trajectory in the future. Meanwhile, our method outperforms other reported trajectory prediction methods and can reduce the prediction error by up to 50.00%. The qualitive results also demonstrate that the GISNet can capture the vehicle motion trend and generate the accurate prediction result.

In future work, the background scene feature should also be taken into account for the trajectory prediction. In [27], the additional features are extracted from the image to help making a more accurate prediction. However, the general feature may not help the model a lot, in contrast, it may introduce extra noises. Hence, a more detailed and targeted feature need to be extracted and included in model prediction, for example, the road segmentation information. Intuitively, the additional feature could be used to further improve the accuracy of the vehicle trajectory prediction.

REFERENCES

[1] Y. Jia, Y. Lu, J. Shen, Q. A. Chen, H. Chen, Z. Zhong, and T. Wei, "Fooling detection alone is not enough: Adversarial attack against multiple object tracking," in *International Conference on Learning Representations*, 2020.

[2] Y. Lu and S. Velipasalar, "Autonomous footstep counting and traveled distance calculation by mobile devices incorporating camera and accelerometer data," *IEEE Sensors Journal*, vol. 17, no. 21, pp. 7157–7166, Nov 2017.

[3] Y. Lu and S. Velipasalar, "Autonomous choice of deep neural network parameters by a modified generative adversarial network," in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 3846–3850.

[4] Z. Zhao, K. Pugdeethosapol, S. Lin, Z. Li, C. Ding, Y. Wang, and Q. Qiu, "Learning topics using semantic locality," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 3710–3715.

[5] G. Yuan, C. Ding, R. Cai, X. Ma, Z. Zhao, A. Ren, B. Yuan, and Y. Wang, "Memristor crossbar-based ultra-efficient next-generation baseband processors," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2017, pp. 1121–1124.

[6] H. Fang, A. Shrestha, Z. Zhao, Y. Li, and Q. Qiu, "An event-driven neuromorphic system with biologically plausible temporal dynamics," in *38th IEEE/ACM International Conference on Computer-Aided Design, ICCAD 2019*. Institute of Electrical and Electronics Engineers Inc., 2019, p. 8942083.

[7] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *2013 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2013, pp. 4363–4369.

[8] Z. Jin, Z. Zhao, C. Luo, F. Basti, A. Solomon, M. C. Gursoy, C. Caicedo, and Q. Qiu, "Simulation of real-time routing for uas traffic management with communication and airspace safety considerations," *arXiv preprint arXiv:2002.11861*, 2020.

[9] Z. Zhao, Z. Jin, C. Luo, H. Fang, F. Basti, M. C. Gursoy, C. Caicedo, and Q. Qiu, "Temporal and spatial routing for large scale safe and connected uas traffic management in urban areas," in *2019 IEEE 25th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. IEEE, 2019, pp. 1–6.

[10] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The apolloscape dataset for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 954–960.

[11] S. Otaki, T. Kindo, Y. Sakaguchi, R. Funayama, and A. SHIMIZU, "Autonomous driving control system for vehicle," Jul. 2 2019, uS Patent App. 10/338,584.

[12] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1243–1274, 2018.

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

[14] Z. Zhao, C. Luo, J. Zhao, Q. Qiu, M. C. Gursoy, C. Caicedo, and F. Basti, "A simulation framework for fast design space exploration of unmanned air system traffic management policies," in *2019 Integrated Communications, Navigation and Surveillance Conference (ICNS)*. IEEE, 2019, pp. 1–10.

[15] Z. Shi, M. Xu, Q. Pan, B. Yan, and H. Zhang, "Lstm-based flight trajectory prediction," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–8.

[16] F. Altché and A. de La Fortelle, "An lstm network for highway trajectory prediction," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 353–359.

[17] H. Xue, D. Q. Huynh, and M. Reynolds, "Ss-lstm: A hierarchical lstm model for pedestrian trajectory prediction," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1186–1194.

[18] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi, "Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 399–404.

[19] N. Nikhil and B. Tran Morris, "Convolutional neural network for trajectory prediction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 0–0.

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

[21] X. Li, X. Ying, and M. C. Chuah, "Grip: Graph-based interaction-aware trajectory prediction," *arXiv preprint arXiv:1907.07792*, 2019.

[22] M. Bahram, C. Hubmann, A. Lawitzky, M. Aeberhard, and D. Wollherr, "A combined model-and learning-based framework for interaction-aware

maneuver prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1538–1550, 2016.

[23] W. Choi and S. Savarese, "Understanding collective activitiesof people from videos," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 6, pp. 1242–1257, 2013.

[24] S. Savarese and W. Choi, "A unified framework for multi-target tracking and collective activity recognition," in *European Conference on Computer Vision*. Springer, 2012, pp. 215–230.

[25] 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.

[26] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," 2019.

[27] T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 126–12 134.

[28] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[29] J. Colyar and J. Halkias, "Us highway 80 dataset," federal highway administration (fhwa), vol," *Tech, no. Rep*, 2006.

[30] J. Halkias and J. Colyar, "Us highway 101 dataset," *Federal Highway Administration (FHWA), Tech. Rep. FHWA-HRT-07-030*, 2007.

[31] T. Chai and R. R. Draxler, "Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature," *Geoscientific model development*, vol. 7, no. 3, pp. 1247–1250, 2014.

[32] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive bayesian filters: A comparative study," in *German Conference on Pattern Recognition*. Springer, 2013, pp. 174–183.

[33] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 204–211.

[34] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 961–971.