Vehicle Trajectory Prediction at Intersections using Interaction based Generative Adversarial Networks

Debaditya Roy, Tetsuhiro Ishizaka, C. Krishna Mohan, and Atsushi Fukuda

***Abstract*— Vehicle trajectory prediction at intersections is both essential and challenging for autonomous vehicle navigation. This problem is aggravated when the traffic is predominantly composed of smaller vehicles that frequently disobey lane behavior as is the case in many developing countries. Existing macro approaches consider the trajectory prediction problem for lane-based traffic that cannot account when there is a high disparity in vehicle size and driving behavior among different vehicle types. Hence, we propose a vehicle trajectory prediction approach that models the interaction among different types of vehicles with vastly different driving styles. These interactions are encapsulated in the form of a social context embedded in a Generative Adversarial Network (GAN) to predict the trajectory of each vehicle at either a signalized or non-signalized intersection. The GAN model produces the most acceptable future trajectory among many choices that conform to past driving behavior as well as the trajectories of neighboring vehicles. We evaluate the proposed approach on aerial videos of intersections from the benchmark VisDrone dataset. The proposed GAN based approach achieves 6.4% relative improvement over state-of-the-art in predicting trajectories.**

I. Introduction

Understanding driving behavior is essential at intersections which account for about 40% of all accidents [1]. Especially in the absence of turning lanes, the propensity of accidents increase substantially. Vehicle trajectory prediction is important in understanding driving behavior as it can help in inferring the navigation style of vehicles at an intersection when vehicle are in close proximity to each other. Traditionally, sensors such as magnetometer detectors, loop detectors, ultrasonic sensors, and surveillance video cameras have been used to monitor intersections. However, these sensors are prohibitively expensive to set up and operate at all intersections. Particularly, surveillance video cameras that are being increasingly employed for traffic monitoring suffer from issues like occlusion, shadows, and a limited field of view. Although many techniques have been proposed to mitigate these challenges [2], [3], traditional surveillance cameras are still not viable for monitoring all the lanes in an intersection. In contrast, an Unmanned Aerial Vehicle (UAV) can be deployed as a cost-effective solution to monitor all the lanes of an intersection. Especially, with the availability of lightweight, high-resolution cameras, even smaller vehicles like motorbikes can be captured in detail. Furthermore, UAVs provide a top-view perspective that is devoid of occlusion and shadows that makes aerial videos ideal for capacity analysis of intersections.

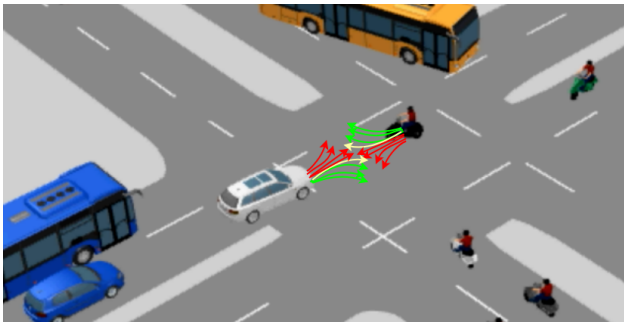
D. Roy, T. Ishizaka, A. Fukuda are with the Department of Transportation Systems Engineering, Nihon University, Chiba, 274-8501, Japan. C. Krishna Mohan is with the Department of Computer Science and Engineering, Indian Institute of Technology Hyderabad, India.

Vehicle trajectory prediction in aerial videos can help in understanding the complex interactions among different types of vehicles especially in developing nations where lane based driving is not followed. For example, motorcycles may maneuver between the gaps of large stationary vehicles at a stop line to move to the front of the queue while in the presence of no gaps, a bus with its considerable size and slow acceleration may impede the progress of smaller vehicles behind it. To deal with such diverse scenarios, we propose a generative adversarial network (GAN) based approach that predicts the best possible route for each vehicle with respect to future interactions that can occur due to close proximity to other vehicles. Some examples of the multiple prediction paths available during various vehicle maneuvers like *overtaking* and *merging* at intersections are presented in Figure 1.

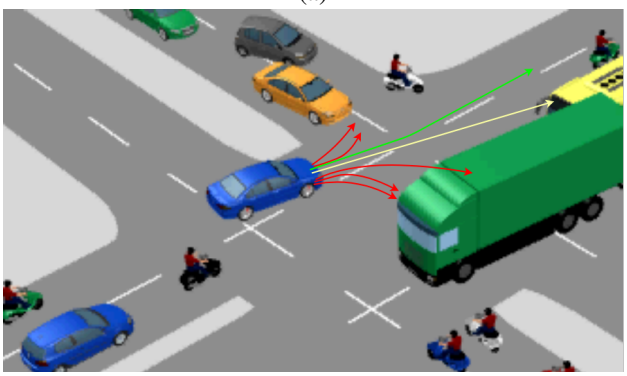
Estimating the interactions between vehicles during the maneuvers shown in Figure 1 requires generating future trajectories that are aligned with the past behavior for each vehicle. The past behavior acts as a prior or condition for the future trajectories and hence, we propose to use a conditional Generative Adversarial Network (GAN) [4] that has been shown to generate multiple predictions from the same prior distribution. During training, these predictions are then pruned based on its closeness to ground-truth. Further, the pooling module in the generator ensures that these trajectories can effectively avoid nearby vehicles. These properties of the trajectories learned during training help the generator in the GAN to produce useful predictions for a given test trajectory. The proposed conditional GAN based on interactions between vehicles shows impressive accuracy in predicting vehicle trajectories for different vehicle maneuvers like merging and overtaking in both signalized and non-signalized intersections.

II. Related Work

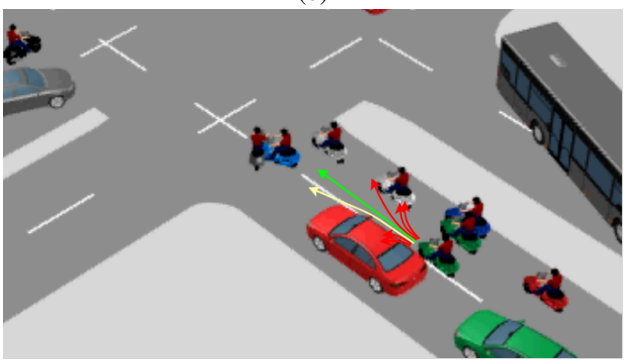
The most popular method for traffic flow analysis is the car-following model [5] that is generally used to describe homogeneous traffic with lane discipline. More recently, to accommodate motorcycle-heavy traffic, a tri-class flow (considering bus, car, and motorcycle as separate flows) was empirically studied in [6]. The traffic flow problem was described as two-wheeler accumulation in different lanes alongside buses and cars which were segmented as vehicle packets. However, these vehicle packets were still segregated by lanes. Such a packet formation fails to account for the unique kinetic characteristics of two-wheelers riding between lanes as suggested by the authors in [6]. Hence, interaction



(a)



(b)



(c)

Fig. 1: Multiple prediction paths are available during various vehicle maneuvers: (a) *avoiding oncoming traffic*, (b) *overtaking*, and (c) *merging*. The paths in green are acceptable while the ones in red are not. The paths in yellow are not optimal for the target vehicle but can be considered based on the decision taken by another vehicle (s). The possible trajectories for all the vehicles are not shown for the sake of clarity. Best viewed in color.

models also known as social force models were developed where each object is considered to be dependent on other objects and environmental factors [7]. Understanding these forces and accounting for them allow effective tracking even in crowded traffic scenes generally encountered at intersections. Social force models categorize target behavior based on two aspects, individual force and group force.

Individual force is defined for each target, and is further subdivided into two forces - *fidelity* that means that the target should not change its desired direction, and *constancy* which means that one should not suddenly change its speed and direction. Group force is further categorized into three types of forces - *attraction* between individuals moving together as a group, *repulsion* that refers to the minimum distance maintained between members in a group, and *coherence* that means individuals moving together in a group move with similar velocity.

The majority of existing publications focus on modeling pedestrian dynamics with social force models [8]–[11] but there is limited literature on traffic modeling using social force dynamics [12]–[14]. However, the traffic models developed with social force carefully consider vehicle dimensions, turning radius, the exact distance between vehicles, etc. In real scenarios, for any arbitrary vehicle at any intersection, information about vehicle dimensions and exact distances are difficult to obtain from aerial videos. Hence, the relative distance between the trajectories of the neighboring vehicles which is independent of the dimensions of the target vehicle used [8], [9]. However, these approaches focus on predicting the average future trajectory by minimizing the Euclidean distance from the ground-truth future trajectory whereas the goal should be to generate multiple good trajectories for every vehicle given the current position. Hence, a natural choice is Generative Adversarial Networks (GAN) that can predict multiple likely trajectories based on a vehicle's past driving path. Further, we utilize a pooling layer to model vehicle-vehicle interactions and a loss function that allows the network to produce multiple diverse future trajectories for the same observed sequence. These future trajectories are evaluated on the distance and probability of collision with neighboring vehicles.

III. Vehicle Interaction Modeling using GAN

In order to estimate the influence of various vehicles in the vicinity of the target vehicle, there is a need to jointly reason and predict the future trajectories of all the vehicles involved in an intersection. Assuming that the trajectories for the vehicles in a scene are obtained from a tracking algorithm as $\mathbf{X} = X_1, X_2, \cdots, X_n$, the goal is to predict the future trajectories $\hat{\mathbf{Y}} = \hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_n$ of all the vehicles simultaneously. The input trajectory of a vehicle i is defined as $X_i = (x_i^t, y_i^t)$ from time steps $t = 1, ..., t_{obs}$, the ground-truth future trajectory is defined as $Y_i = (x_i^t, y_i^t)$ from time steps $t = t_{obs} + 1, \cdots, t_{pred}$, and the predicted trajectory is defined as $\hat{Y}_i$.

A Generative Adversarial Network (GAN) comprises of two neural networks - a generative model G to capture the data distribution, and a discriminative model D to estimate whether a sample arrived from the training data rather than G. The generator G takes a latent variable z as input, and outputs a sample $G(z)$ while the discriminator D takes a sample x as input and outputs $D(x)$ which represents the probability that it is real. The training procedure is akin to

a two-player min-max game with the objective function

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p(z)}[log(1-D(G(z)))]. \quad (1)$$

Conditional GAN expands the functionality of the traditional GAN architecture by accepting an additional input c at both the generator and discriminator to produce $G(z,c)$ and $D(x,c)$, respectively [4]. Such conditional GANs can be used to replicate models conditioned on a prior distribution which in this case is the past trajectory of the vehicle during the observation period $t = 1, ..., t_{obs}$).

Trajectories of vehicle movements are a form of time-series data with many possible futures based on different intentions like giving more space to larger vehicles and avoiding overtaking maneuvers, left turn, right turn, or U-turn on a multi-lane road, etc. This makes the vehicle trajectory prediction problem truly multimodal and GANs can help predict all the different possibilities. In a nutshell, the GAN used in this work consists of a generator, a pooling stage, and a discriminator. The generator is an encoder-decoder framework where the hidden states of encoder and decoder are linked with the help of a pooling module. The generator takes in input X_i and outputs predicted trajectory $\hat{Y}_i$. The discriminator receives the entire sequence comprising both input trajectory X_i and future prediction $\hat{Y}_i$ (or Y_i) as input and classifies them as either real or fake.

In order to produce the input for the generator, the location of each person is embedded into a fixed length vector $\mathbf{e}_i^t$ using a 1-layer multi-layer perceptron (MLP) as in [8]. These embeddings are then used to initialize the hidden state of the encoder in the long short-term memory (LSTM) cell at time t as

$$\begin{aligned} e_i^t &= \phi(x_i^t, y_i^t; \mathbf{W}_{ee}), \\ h_{ei}^t &= LSTM(h_{ei}^{t-1}, e_i^t; \mathbf{W}_{encoder}), \end{aligned} \quad (2)$$

where $\phi(.)$ is an embedding function with Rectified Linear Unit (ReLU) non-linearity, $\mathbf{W}_{ee}$ is the embedding weight, h_{ei}^t is the hidden state of the i^{th} encoder at time t and the LSTM weights, $\mathbf{W}_{encoder}$, are shared between all the vehicles to provide global context of the scene. The encoder learns the state of the vehicle and stores the motion pattern for that particular vehicle. Similar to the social LSTM model [8], a pooling stage (PS) is designed to share the information between the different encoders that models vehicle-vehicle interaction. After observing the motion of each vehicle till t_{obs}, the hidden states of all the vehicles present at the intersection are pooled (max-pooled in our implementation) to obtain a tensor $\mathbf{P}_i$ for each vehicle. As the goal is to produce future trajectories that are synchronized with past driving behavior in the observation period, the hidden state of the decoder is conditioned based on the combined tensor as

$$\begin{aligned} c_i^t &= \gamma(\mathbf{P}_i, h_{ei}^t; \mathbf{W}_c), \\ h_{di}^t &= [c_i^t, z], \end{aligned} \quad (3)$$

where $\gamma()$ is a multi-layer perceptron (MLP) with ReLU non-linearity, h_{di}^t is the hidden state of the i^{th} decoder at time t, and $\mathbf{W}_c$ is the embedding weight.

After initializing the decoder states as above, the predictions can be obtained as

$$\begin{aligned} e_i^t &= \phi(x_i^{t-1}, y_i^{t-1}; \mathbf{W}_{ed}), \\ \mathbf{P}_i &= PS(h_{d1}^{t-1}, ..., h_{dn}^t), \\ h_{di}^t &= LSTM(\gamma(\mathbf{P}_i, h_{di}^{t-1}), e_i^t; \mathbf{W}_{decoder}), \\ (\hat{x}_i^t, \hat{y}_i^t) &= \gamma(h_{di}^t), \end{aligned} \quad (4)$$

where $\phi(.)$ is an embedding function with ReLU non-linearity with $\mathbf{W}_{ed}$ as the embedding weights. The LSTM weights are given by $\mathbf{W}_{decoder}$ and $\gamma(.)$ denotes an MLP.

The discriminator uses a separate encoder which takes as input $T_{real} = [X_i, Y_i]$ or $T_{fake} = [X_i, \hat{Y}_i]$ and classifies them as real or fake. The discriminator learns interaction behavior and classifies unacceptable trajectories as "fake". While a GAN is trained using adversarial loss given in 2, L2 loss is used to estimate the distance of the generated path from the actual ground-truth.

To estimate the trajectory of multiple vehicles, we need to share information across the LSTMs representing each vehicle. However, the number of vehicles at an intersection is high, and the number varies depending on the traffic condition. Therefore, there is a need for a compact representation to store shared information. Further, local interactions are not always sufficient to determine future trajectories, and far-away vehicles might impact the path taken by a vehicle. Hence, the network needs to model the global context. In social pooling [8], [9] based approaches, a grid-based pooling scheme is proposed that considers only local context and fail to capture global context. As per [15], both a compact representation and global context can be learned using a symmetric function on transformed elements of the input set of points. Hence, in this work, the input coordinates are passed through an MLP followed by a symmetric function like Max-Pooling. The pooled vector $\mathbf{P}_i$ summarizes all the information needed for a vehicle to choose a path. Also, the relative position of each person in relation to person i is augmented with input to the pooling module.

Though GAN produces good predictions, these predictions are the "average" prediction in case of multiple outputs. In order to encourage the generation of diverse samples, we use a variety loss given in [16]. For each scene, k possible output predictions are generated by randomly sampling z from $\mathcal{N}(0,1)$ and the best prediction is obtained as

$$\mathcal{L}_{variety} = \min_k \|Y_i - \hat{Y}_i^{(k)}\|_2, \quad (5)$$

where k is a hyper-parameter. By considering the trajectory that is closest in terms of Euclidean distance to the ground-truth trajectory, the network explores only those outputs that are closest to the past trajectory.

IV. Experimental Results

A. Dataset description

The VisDrone dataset contains 96 video clips that include 56 clips for training, 7 for validation, and 33 for testing.

Among them, we chose only the videos that depict vehicular traffic at intersections. Finally, the dataset considered for evaluation in this work consists of 23 clips for training (10,239 frames with approximately 11,000 vehicles), 5 for validation (2,033 frames with approximately 2,400 vehicles), and 6 for testing (2,110 frames with approximately 2,500 vehicles) from the VisDrone dataset.

B. Implementation Details

The generator in the GAN utilizes only a decoder whereas the discriminator utilizes the encoder as well as the decoder. For both the encoder and the decoder, LSTMs are used to represent the trajectories with hidden state dimensions of 16 and 32, respectively. The input trajectory coordinates are embedded as 16 dimensional vectors. The generator and discriminator are trained with a batch size of 16 for 200 epochs using Adam optimizer [17] with a starting learning rate of 0.001.

C. Quantitative Analysis

Most existing trajectory prediction algorithm provide ground truth during training and for initializing the track during the observation period. However, obtaining annotated ground-truth videos is both expensive and seldom available. Hence, to mimic the same format for our experiments, we propose to use tracking outputs obtained from tracking algorithms for the observation period ($t = 1, \cdots, t_{obs}$) and the actual ground-truth from $t = t_{obs}+1, \cdots, t_{pred}$ to determine the prediction performance. This experimental protocol adequately reflects real-world situations where ground-truth trajectories may not be available for vehicles for the hundreds of vehicles at an intersection. So, in order to obtain the trajectories for the observation period, we compared on-line tracking algorithms that can efficiently track a large number of targets - 1) Markov Decision Process (MDP) based tracker [18], 2) simple, online, real-time tracking (SORT) [19], and 3) deep SORT [20] that integrates the SORT algorithm with appearance features from the YOLO [21] detection framework. The metrics used for comparison in Table I are Multiple Object Tracking Accuracy (MOTA) combines three sources of errors and Multiple (MOTP). While we were able to test SORT and MDP with different detection frameworks, deepSORT integrates the appearance features provided by different layers of the YOLO [22] architecture to associate the targets. This prevents the tracking framework to be decoupled from the detection network. It can be observed from Table I that the MDP tracker has the highest recall and MOTA among the other trackers.

We compare the GAN based prediction system to existing approaches using social LSTM (S-LSTM) [8] and social attention-based structural recurrent neural network (S-RNN) [11]. The input coordinates obtained from the tracker output are transformed into a relative coordinate system with the center of the video taken as the origin in order to achieve translation invariance. For the S-LSTM and S-RNN, we use the implementation provided by the respective authors. For the GAN model, we modify and use the implementation given by the authors in [16]. All the prediction networks follow different protocols for observation and prediction lengths. For simplicity of comparison, we follow an observation length of 8 time steps ($t_{obs} = 8$) and prediction length of 8 time steps ($t_{pred} = 8$) for all the networks. Similar to the S-LSTM model, neighboring trajectories were pooled at every time step during testing. Further, for every trajectory prediction, 20 samples were drawn from the generator and the best one was chosen based on L2 distance for quantitative analysis.

TABLE I: Comparison of different trackers and detectors (in %) on intersection videos for all vehicle types in the VisDrone dataset. The choice of the tracking framework is crucial for robust prediction.

Method		Recall	Precision	MOTA	MOTP
DeepSort [20]	YOLO	7.8	79.3	5.2	73.5
SORT [19]	F-RCNN	17.7	**93.5**	16.0	**79.4**
	SSD	5.5	82.6	4.0	75.7
	R-FCN	16.6	91.5	14.6	78.8
MDP [18]	F-RCNN	**25.5**	89.4	**22.2**	78.0
	SSD	11.1	78.4	7.8	74.4
	R-FCN	24.6	87.6	20.8	77.0

The comparison is done using the following evaluation metrics:

- Average Displacement Error (ADE) which measures the average L2 distance between ground-truth and the prediction over all the predicted time steps.
- Final Displacement Error (FDE) that measures the distance between the predicted final destination actual destination at the end of the prediction period t_{pred}.

In Table II, we present the comparison of vehicle trajectory prediction performance of GAN with S-RNN and S-LSTM. It can be observed that GAN produces better ADE and FDE scores than the other two prediction approaches. This can be attributed to the generator in GAN being trained with a variety loss that is able to predict more diverse set of trajectories than both S-LSTM and S-RNN. Further, the global context employed in GAN is more apt for vehicle trajectory prediction at intersections as vehicles can rapidly accelerate or decelerate at intersections. Even vehicles which are separated initially can become close rapidly and a global strategy helps in keeping track of such movements for better predictions.

D. Qualitative Analysis

Traffic prediction using GAN having a social structure helps us predict two basic movement types used by smaller vehicles like motorcycles and scooters in traffic - merging and overtaking. While merging, vehicles avoid collisions while continuing towards their destination by either slowing down or altering their course slightly or a combination of both. This behavior is highly dependent on the context and behavior of other surrounding vehicles. The proposed model can predict the variation in both the speed and direction of a vehicle to effectively navigate nearby traffic. For instance, the model predicts that either vehicle Y (yellow) slows down or

TABLE II: Comparison of prediction performance with state-of-the-art on the intersection videos of VisDrone dataset.

Method			**ADE**	**FDE**
S-RNN [11]	DeepSort		0.88	1.60
	SORT	F-RCNN	0.92	1.76
		R-FCN	0.92	1.75
		SSD	0.85	1.52
	MDP	F-RCNN	0.89	1.56
		R-FCN	0.86	1.61
		SSD	0.94	1.66
S-LSTM [8]	DeepSort		0.89	1.57
	SORT	F-RCNN	0.77	1.62
		R-FCN	0.91	1.66
		SSD	0.92	1.67
	MDP	F-RCNN	0.82	1.53
		R-FCN	0.89	1.51
		SSD	0.79	1.42
GAN	DeepSort		0.91	1.65
	SORT	F-RCNN	0.87	1.66
		R-FCN	0.87	1.65
		SSD	0.77	1.42
	MDP	F-RCNN	0.82	1.53
		R-FCN	0.84	1.58
		SSD	**0.72**	**1.32**

both vehicle R (red) and Y change direction to avoid collision (Figure 2 (a) and (b)). This scenario shows that as the GAN model can evaluate the likelihood of multiple future paths for every vehicle, it steers every vehicle in the direction that is least likely to be taken by other vehicles in the future.

Another common scenario encountered in traffic is where a vehicle might want to either maintain pace or may overtake the vehicle in front. This has been studied with car-following models in literature [5]. The decision making ability while *overtaking* is restricted by the field of view. However, as the GAN model has access to the ground-truth positions of all the vehicles involved in the scene, it results in some interesting predictions. For example, in Figures 2 (c) and (d), the model predicts that vehicle R (in red) is obstructed by vehicle B (blue) and will give way by changing their direction. This global knowledge allows GAN to correctly predict that vehicle Y (yellow) will overtake vehicle B.

Vehicles also avoid each other when moving in opposite directions without any physical barrier separating both the streams of traffic. This tendency manifests in smaller vehicles generally bunching with other vehicles moving in the same direction. Also, smaller vehicles (vehicle Y) mostly observe the movement of larger vehicles (vehicle R) in the opposite direction and overtake only if they predict that there is adequate clearance distance (Figure 2 (f)). However, in the presence of smaller vehicles, the driver in vehicle Y (yellow) makes a choice very late and close to the oncoming vehicle R (red) (Figure 2 (e)). The model is not able to distinguish between these two behaviors as the type of vehicle is not taken into consideration during prediction and the prediction is not aligned with the ground-truth (Figure 2 (e)). In such case, we hypothesize that decisions based on local vicinity can produce better predictions rather than accounting for vehicles that are further away (global context).

V. Conclusion

In this paper, we proposed a Generative Adversarial Network (GAN) based approach in order to predict trajectories of vehicles at both signalized and non-signalized intersections. The prediction algorithm can predict vehicle trajectory resulting from different traffic maneuvers like overtaking, merging, and avoiding oncoming traffic without any additional information about the dimension of the road. An evaluation on the intersection videos of the VisDrone dataset demonstrates the efficacy of GAN in predicting trajectories with minimal deviation compared to the actual trajectories followed by different types of vehicles. Further, as there are no assumptions about the type of traffic and their movements, the method can be applied for the analysis of any type of intersection.

References

[1] N. H. T. S. Administration *et al.*, "Crash factors in intersection-related crashes: An on-scene perspective," *Nat. Center Stat. Anal., National Highway Traffic Safety Administration, Washington, DC, USA, Tech. Rep. DOT HS*, vol. 811366, 2010.

[2] S.-P. Lin, Y.-H. Chen, and B.-F. Wu, "A real-time multiple-vehicle detection and tracking system with prior occlusion detection and resolution, and prior queue detection and resolution," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 1, Aug 2006, pp. 828–831.

[3] T. Gandhi and M. M. Trivedi, "Vehicle surround capture: Survey of techniques and a novel omni-video-based approach for dynamic panoramic surround maps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 3, pp. 293–308, 2006.

[4] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[5] G. F. Newell, "A simplified car-following theory: a lower order model," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 195–205, 2002.

[6] C. Lan and G. Chang, "Empirical observations and formulations of tri-class traffic flow properties for design of traffic signals," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2018.

[7] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[8] 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.

[9] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European conference on computer vision*. Springer, 2016, pp. 549–565.

[10] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 300–311.

[11] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–7.

[12] D. N. Huynh, M. Boltze, and A. T. Vu, "Modelling mixed traffic flow at signalized intersection using social force model," *Journal of the Eastern Asia Society for Transportation Studies*, vol. 10, pp. 1734–1749, 2013.

[13] W. Huang, M. Fellendorf, and R. Schönauer, "Social force based vehicle model for 2-dimensional spaces," in *91st Annual Meeting of the Transportation Research Board. Washington, DC, USA*, 2011.

[14] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[15] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 77–85.

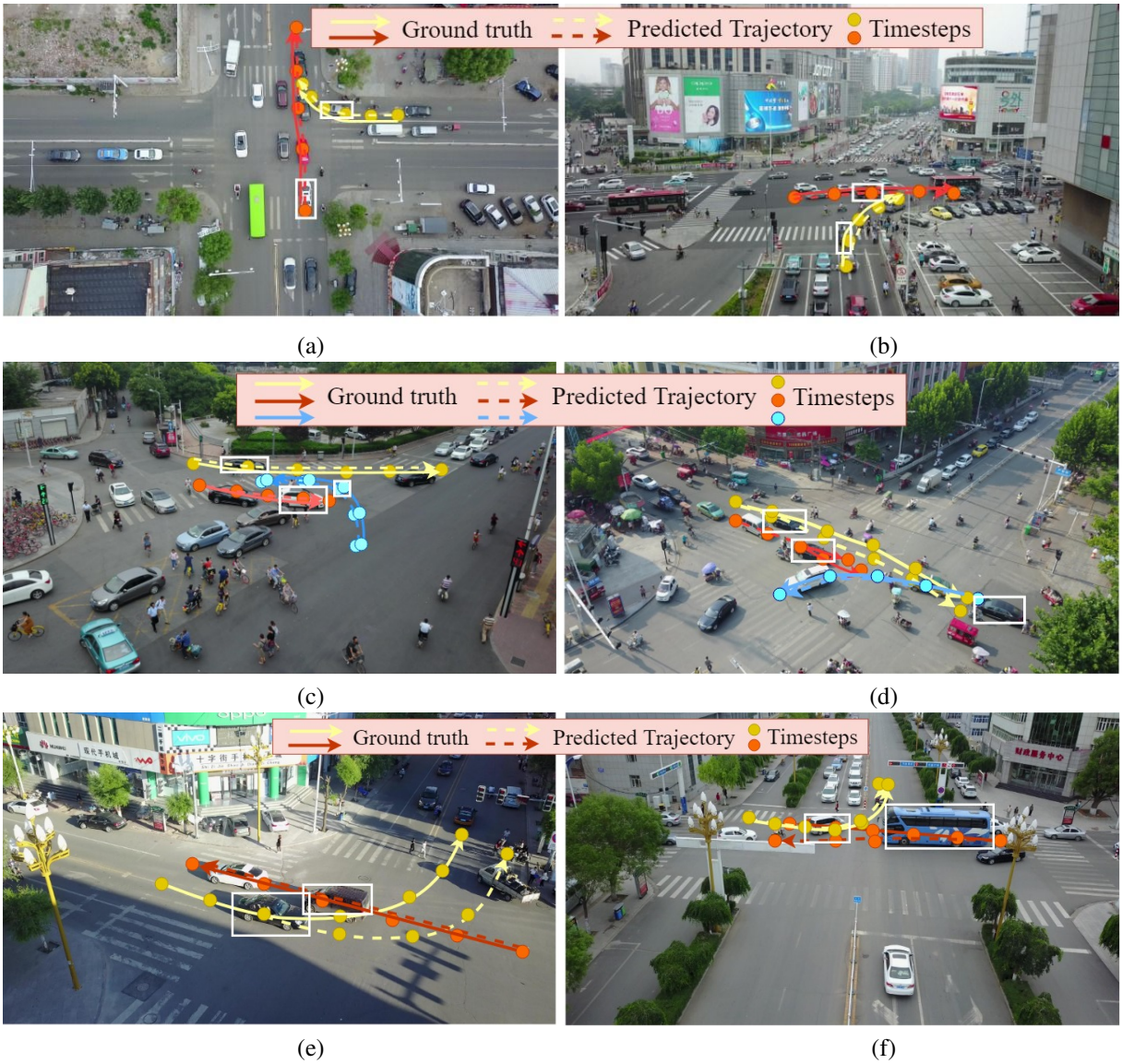


Fig. 2: Prediction of traffic flow using GAN in signalized and unsignalized intersections during various maneuvers - (a) and (b) *merging*, (c) and (d) *overtaking*, and (e) and (f) preventing oncoming traffic. While the traffic flow during merging and overtaking is predicted correctly, overtaking is more challenging as the size of the vehicle dictates the distance maintained by the drivers as seen in (e) and (f). The vehicles in consideration are enclosed in white boxes. The trajectories of other vehicles are not shown to maintain clarity. Best viewed in color.

[16] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 2255–2264.

[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[18] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4705–4713.

[19] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468.

[20] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 3645–3649.

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[22] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 6517–6525.