Center for Advanced Multimodal Mobility

Solutions and Education

Project ID: 2019 Project 08

Deep-Learning Based Trajectory Forecast for Safety of Intersections with Multimodal Traffic (Phase II)

Final Report

by

Abduallah Mohamed (ORCID ID: https://orcid.org/0000-0002-6074-6010)
PhD student, University of Texas Austin
Department of Civil, Architectural and Environmental Engineering
301E E Dean Keeton St c1700 Austin, Texas 78712

Kun Qian (ORCID ID: https://orcid.org/0000-0002-9063-102X)
PhD student, University of Texas Austin
Department of Civil, Architectural and Environmental Engineering
301E E Dean Keeton St c1700 Austin, Texas 78712

Christian Claudel. (ORCID ID: https://orcid.org/0000-0002-3783-4928)
Assistant Professor, University of Texas Austin
Department of Civil, Architectural and Environmental Engineering
301E E Dean Keeton St c1700 Austin, Texas 78712

for

Center for Advanced Multimodal Mobility Solutions and Education
(CAMMSE @ UNC Charlotte)
The University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223

August 2020

ACKNOWLEDGEMENTS

This project was funded by the Center for Advanced Multimodal Mobility Solutions and Education (CAMMSE @ UNC Charlotte), one of the Tier I University Transportation Centers that were selected in this nationwide competition, by the Office of the Assistant Secretary for Research and Technology (OST-R), U.S. Department of Transportation (US DOT), under the FAST Act. The authors are also very grateful for all of the time and effort spent by DOT and industry professionals to provide project information that was critical for the successful completion of this study.

DISCLAIMER

The contents of this report reflect the views of the authors, who are solely responsible for the facts and the accuracy of the material and information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation University Transportation Centers Program and the National Science Foundation in the interest of information exchange. The U.S. Government National Science Foundation assumes no liability for the contents or use thereof. The contents do not necessarily reflect the official views of the U.S. Government. This report does not constitute a standard, specification, or regulation.

EXECUTIVE SUMMARY

Safety at multimodal intersections is a critical problem in urban environments. While the focus has historically been on better road design and improvement of road geometries to minimize the risk of collision, to date, little has been done with respect to active safety: preventing these collisions from occurring by tracking different vehicles and road users. Active safety becomes a topic of interest with the emergence of autonomous vehicles-human interactions as well.

Among all road users, pedestrians are particularly vulnerable to collisions, and can have erratic behaviors that make collision avoidance a challenging problem. One of the most critical problems is the prediction of pedestrian future trajectories. These trajectories are not only influenced by the pedestrian itself but also by interactions with surrounding objects and with the environment (road type, presence of sidewalks, crossings).

Previous methods investigated in the literature modeled these interactions by using a variety of aggregation techniques that integrate different learned pedestrians states. In this report, we present the Social Spatio-Temporal Graph Convolutional Neural Network (Social-STGCNN), which uses graph theory to encode pedestrian/environment interactions. This method is based on artificial neural netowrks (CNNs) and graph theory-based potential functions.

We validate this algorithm on experimental drone-captured data. Validation shows that the performance improves with respect to state of the art methods by 20% on the Final Displacement Error (FDE) metric with 8.5 times less parameters and up to 48 times faster inference speed than previously reported methods. In addition, the proposed model is data efficient, and exceeds previous state of the art on the ADE metric with only 20% of the training data. We propose a kernel function to embed the social interactions between pedestrians within the adjacency matrix. Through qualitative analysis, we show that the model inherited social behaviors that can be expected between pedestrians trajectories, including collision avoidance or merging of pedestrian trajectories.

Contents

Chapter 1. Introduction

1.1 Problem Statement

Predicting pedestrian trajectories is of major importance for several applications including autonomous driving and surveillance systems. In autonomous driving, an accurate prediction of pedestrians trajectories enables the controller to plan ahead the motion of the vehicle in an adversarial environment. For example, it is a critical component for collision avoidance systems or emergency braking systems. In general traffic collision avoidance systems (Figure 1), such component would be very important in the synthesis of controller actions, preventing a collision

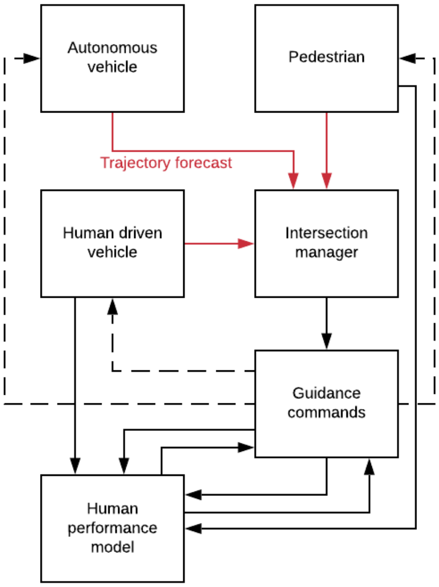


Figure 1. Overview of a collision avoidance system in a mixed autonomous-non autonomous setting. Trajectory forecasts are essential to the intersection manager. Such trajectories can be reliably forecast for some vehicles (for example human or autonomous or human driven vehicles), though they are much more unpredictable in some other cases.

1.2 Objectives

The objective of this report is to introduce a novel computational method for solving the trajectory prediction problem, using a combination of artificial neural networks (convolutional neural networks) and graph theory.

1.3 Expected Contributions

The contributions of this work are threefold: improve the performance of state-of-the-art methods in predicting pedestrian trajectories, on the following criteria: average displacement error (errors over the trajectory) and final displacement error (prediction error for the final trajectory point)

1.4 Report Overview

The remainder of this report is organized as follows: Chapter 2 presents a literature survey of existing work on this topic. Chapter 3 discusses the formulation of the problem, while chapter 4 introduces the deep learning based model, and the kernel function used for modeling interactions between pedestrians. Chapter 5 shows a comparison of this model with state of the art models, on benchmark datasets.

Chapter 2. Literature Review

2.1 Introduction

This chapter presents a review of the literature of existing deep models for pedestrian trajectories prediction.

2.2 Related models

2.2.1 Early work

Human trajectory prediction using deep models SocialLSTM [1] is one of the earliest deep model focusing on pedestrian trajectory prediction. Social-LSTM uses a recurrent network to model the motion of each pedestrian, then they aggregated the recurrent outputs using a pooling mechanism and predict the trajectory afterwards. SocialLSTM assumes the pedestrian trajectory follow a bi-variate Gaussian distribution, in which we follow this assumption in our model. Later works such as Peek Into The Future (PIF) [14] and State-Refinement LSTM (SR-LSTM) [30] extends [1] with visual features and new pooling mechanisms to improve the prediction precision. It is noticeable that SR-LSTM [30] weighs the contribution of each pedestrian to others via a weighting mechanism. It is similar to the idea in Social-BiGAT [10] which uses an attention mechanism to weigh the contribution of the recurrent states that represent the trajectories of pedestrians.

2.2.2 Graph Convolutional Neural Network models

Recent Advancements in Graph CNNs Graph CNNs were introduced by [8] which extends the concept of CNNs into graphs. The Convolution operation defined over graphs is a weighted aggregation of target node attributes with the attributes of its neighbor nodes. It is similar to CNNs but the convolution operation is taken over the adjacency matrix of the graphs. The works [9, 4, 24] extend the graph CNNs to other applications such as matrix completion and Variational Auto Encoders. One of the development related to our work is the ST-GCNN [27]. ST-GCNN is a spatio-temporal Graph CNN that was originally designed to solve skeleton-based action recognition problem. Even though the architecture itself was designed to work on a classification task, we adapt it to suit our problem. In our work, ST-GCNNs extract both spatial and temporal information from the graph creating a suitable embedding. We then operate on this embedding to predict the trajectories of pedestrians. Details are shown in Chapter 4.

Chapter 3. Modeling

3.1 Problem definition

Given a set of *N* pedestrians in a scene with their corresponding observed positions $tr^n_o, n \in \{1, \ldots, N\}$ over a time period T_o, we need to predict the upcoming trajectories tr^n_p over a future time horizon T_p. For a pedestrian *n*, we write the corresponding trajectory to be predicted as the following:

$$tr^n_p = \{\, \mathbf{p}^n_t = (\mathbf{x}^n_t, \mathbf{y}^n_t) \,|\, t \in \{1, \ldots, T_p\}\}$$

where $(\mathbf{x}^n_t, \mathbf{y}^n_t)$ are random variables describing the probability distribution of the location of pedestrian *n* at time *t*, in the 2D space. We make the assumption that $(\mathbf{x}^n_t, \mathbf{y}^n_t)$ follows a bivariate Gaussian distribution such that $\mathbf{p}^n_t \sim \mathrm{N}(\mu^n_t, \sigma_t^n, \rho^n_t)$. Besides, we denote the predicted trajectory as $\hat{\mathbf{p}}^n_t$ which follows the estimated bi-variate distribution $\mathrm{N}(\hat{\mu}^n_t, \hat{\sigma}_t^n, \hat{\rho}^n_t)$.

3.2 Objective function

Our model is trained to minimize the negative log-likelihood, which defined as:

$$L^n(\mathbf{W}) = -\sum_{t=1}^{T_p} \log(\mathrm{P}((\mathbf{p}^n_t | \hat{\mu}^n_t, \hat{\sigma}_t^n, \hat{\rho}^n_t)) \qquad (1)$$

in which **W** includes all the trainable parameters of the model, μ^n_t is the mean of the distribution, σ_t^n is the variances and ρ^n_t is the correlation.

Chapter 4. Derivation of the graph-based deep learning model

4.1 Overview of the Social-STGCNN model

The Social-STGCNN model consists of two main parts: the Spatio-Temporal Graph Convolution Neural Network (ST-GCNN) and the Time-Extrapolator Convolution Neural Network (TXP-CNN). The ST-GCNN conducts spatiotemporal convolution operations on the graph representation of pedestrian trajectories to extract features. These features are a compact representation of the observed pedestrian trajectory history. TXP-CNN takes these features as inputs and predicts the future trajectories of all pedestrians as a whole. We use the name Time-Extrapolator because TXP-CNNs are expected to extrapolate future trajectories through convolution operation. Figure 2 illustrates the model.

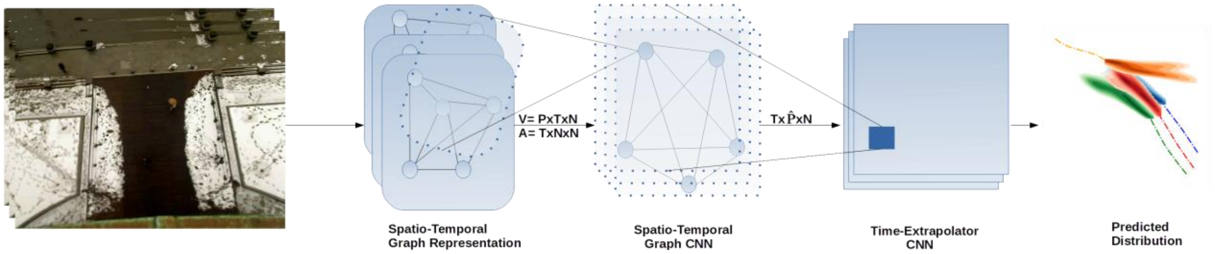


Figure 2. The Social-STGCNN Model. Given T frames, we construct the spatio-temporal graph representing $G = (V,A)$. Then G is forwarded through the Spatio-Temporal Graph Convolution Neural Networks (ST-GCNNs) creating a spatio-temporal embedding. Following this, the TXP-CNNs predicts future trajectories. P is the dimension of pedestrian position, N is the number of pedestrians, T is the number of time steps and $\hat{P}$ is the dimension of the embedding coming from ST-GCNN

4.2 Graph representation of pedestrian trajectories

We first introduce the construction of the graph representation of pedestrian trajectories. We start by constructing a set of spatial graphs G_t representing the relative locations of pedestrians in a scene at each time step t. G_t is defined as $G_t = (V_t,E_t)$, where $V_t = \{v_t^i \mid \forall i \in \{1,...,N\}\}$ is the set of vertices of the graph G_t. The observed location (x_t^i, y_t^i) is the attribute ofv_t^i. E_t is the set of edges within graph G_t which is expressed as

$$E_t = \{e^{ij}{}_t \mid \forall i,j \in \{1,...,N\}\}$$

$e_t^{ij} = 1$ if v_t^i and v_t^j are connected, $e^{ij}{}_t = 0$ otherwise. In order to model how two nodes influence each other, we attach a value a_t^{ij}, which is computed by some kernel function for each e_t^{ij}. a_t^{ij} .

We introduce$a_{sim,t}^{ij}$ as a kernel function to be used within the adjacency matrix A_t. $a_{sim,t}^{ij}$ is defined in equation 2.

$$a^{ij}_{sim,t} = \begin{cases} 1/\|v_t^i - v_t^j\|_2 & , \|v_t^i - v_t^j\|_2 \neq 0 \\ 0 & \text{, Otherwise.} \end{cases} \qquad (2)$$

4.3 Graph convolutional neural network

With the graph representation of pedestrian trajectories, we introduce the spatial convolution operation defined on graphs. For convolution operations defined on 2D grid maps or feature maps, the convolution operation is shown in equation 3.

$$z^{(l+1)} = \sigma(\sum_{h=1}^{k}\sum_{w=1}^{k}(p(z^{(l)}, h, w)).\mathbf{w}^{(l)}(h, w)) \qquad (3)$$

where k is the kernel size and $p(.)$ is the sampling function which aggregates the information of neighbors centering around z [5] and σ is an activation function and (l) indicates layer l.

The graph convolution operation is defined as:

$$v^{i(l+1)} = \sigma(\frac{1}{\Omega}\sum_{v^{j(l)} \in B(v^{i(l)})} p(v^{i(l)}, v^{j(l)}).\mathbf{w}(v^{i(l)}, v^{j(l)})) \qquad (4)$$

where $\frac{1}{\Omega}$ is a normalization term, $B(v^i) = \{v^j | d(v^i,v^j) \leq D\}$ is the neighbor set of vertices v^i and $d(v^i,v^j)$ denotes the shortest path connecting v^i and v^j. Note that Ω is the cardinality of the neighbor set. Interested readers are referred to [8, 27] for more detailed explanations and reasoning.

4.4 Spatio-temporal graph convolutional neural networks

ST-GCNNs extends spatial graph convolution to spatio-temporal graph convolution by defining a new graph G whose attributes are the set of the attributes of G_t. G incorporates the spatio-temporal information of pedestrian trajectories. It is worth noticing that the topology of $G_1,...,G_T$ is the same, while different attributes are assigned to v_t^i when t varies. Thus, we define G as (V,E), in which $V = \{v^i \mid i \in \{1,...,N\}\}$ and $E = \{e^{ij} \mid \forall i,j \in \{1,...,N\}\}$. The attributes of vertex v^i in G is the set of $v_t^i, \forall t \in \{0,...,T\}$. In addition, the weighted adjacency matrix A corresponding to G is the set of $\{A_1,...,A_T\}$. We denote the embedding resulting from ST-GCNN as $\bar{V}$.

4.5 Time extrapolators (TXP)

The functionality of ST-GCNN is to extract spatiotemporal node embedding from the input graph. However, our objective is to predict further steps in the future. We also aim to be a stateless system and here where the TXPCNN comes to play. TXP-CNN operate directly on the temporal dimension of the graph embedding $\bar{V}$ and expands it as a necessity for prediction. Because TXP-CNN depend on convolution operations over the feature space, they have less parameters than recurrent units. A property to note regarding the TXP-CNN layer is that it is not permutation invariant. Therefore, a change in the graph embedding right before TXP-CNN can lead to different outcomes. However, if the order of pedestrians is permutated at the initial time only (but not during intermediate times), then the results are invariant.

Overall, there are two main differences between SocialSTGCNN and the ST-GCNN [27]. First, Social-STGCNN constructs the graph differently than ST-GCNN with a novel kernel function. Second, beyond the spatiotemporal graph convolution layers, we added the flexibility in manipulating the time dimension using the TXP-CNN. ST-GCNN was originally designed for classification. By using TXP-CNN, the model was able to utilize the graph embedding originating from ST-GCNN to predict the future trajectories.

Chapter 5. Validation

5.1 Validation metrics

The model is trained on two human trajectory prediction datasets: ETH [21] and UCY [11]. ETH contains two scenes named ETH and HOTEL, while UCY contains three scenes named ZARA1, ZARA2 and UNIV. The trajectories in datasets are sampled every 0.4 seconds. Our method of training follows the same strategy as Social-LSTM [1]. In Social-LSTM, the model was trained on a portion of a specific dataset and tested against the rest and validated versus the other four datasets. When being evaluated, the model observes the trajectory of 3.2 seconds which corresponds to 8 frames and predicts the trajectories for the next 4.8 seconds that are 12 frames.

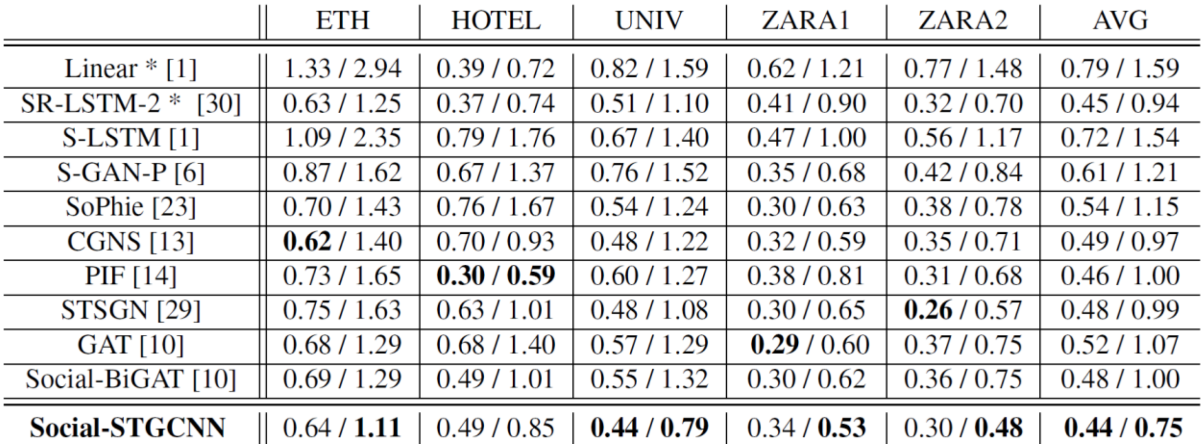
Two metrics are used to evaluate model performance: the Average Displacement Error (ADE) [21] defined in equation 6 and the Final Displacement Error (FDE) [1] defined in equation 7. Intuitively, ADE measures the average prediction performance along the trajectory, while the FDE considers only the prediction precision at the end points. Since Social-STGCNN generates a bi-variate Gaussian distribution as the prediction, to compare a distribution with a certain target value, we follow the evaluation method used in Social-LSTM [1] in which 20 samples are generated based on the predicted distribution. Then the ADE and FDE are computed using the closest sample to the ground truth. This method of evaluation were adapted by several works such as Social-GAN [6] and many more.

$$\text{ADE} = \frac{\sum\limits_{n\in N}\sum\limits_{t\in T_p}\|\hat{p}_t^n - p_t^n\|_2}{N \times T_p} \qquad (6)$$

$$\text{FDE} = \frac{\sum\limits_{n\in N}\|\hat{p}_t^n - p_t^n\|_2}{N}, t = T_p \qquad (7)$$

5.2 Results

In this section, we compare the performance of the Social STGCNN with existing algorithms from the literature, on some benchmark datasets, including ETH, HOTEL, UNIV, ZARA1 and ZARA2. The best performance in ADE and FDE are represented as bold characters.



	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Linear * [1]	1.33 / 2.94	0.39 / 0.72	0.82 / 1.59	0.62 / 1.21	0.77 / 1.48	0.79 / 1.59
SR-LSTM-2 * [30]	0.63 / 1.25	0.37 / 0.74	0.51 / 1.10	0.41 / 0.90	0.32 / 0.70	0.45 / 0.94
S-LSTM [1]	1.09 / 2.35	0.79 / 1.76	0.67 / 1.40	0.47 / 1.00	0.56 / 1.17	0.72 / 1.54
S-GAN-P [6]	0.87 / 1.62	0.67 / 1.37	0.76 / 1.52	0.35 / 0.68	0.42 / 0.84	0.61 / 1.21
SoPhie [23]	0.70 / 1.43	0.76 / 1.67	0.54 / 1.24	0.30 / 0.63	0.38 / 0.78	0.54 / 1.15
CGNS [13]	**0.62** / 1.40	0.70 / 0.93	0.48 / 1.22	0.32 / 0.59	0.35 / 0.71	0.49 / 0.97
PIF [14]	0.73 / 1.65	**0.30 / 0.59**	0.60 / 1.27	0.38 / 0.81	0.31 / 0.68	0.46 / 1.00
STSGN [29]	0.75 / 1.63	0.63 / 1.01	0.48 / 1.08	0.30 / 0.65	**0.26** / 0.57	0.48 / 0.99
GAT [10]	0.68 / 1.29	0.68 / 1.40	0.57 / 1.29	**0.29** / 0.60	0.37 / 0.75	0.52 / 1.07
Social-BiGAT [10]	0.69 / 1.29	0.49 / 1.01	0.55 / 1.32	0.30 / 0.62	0.36 / 0.75	0.48 / 1.00
Social-STGCNN	0.64 / **1.11**	0.49 / 0.85	**0.44 / 0.79**	0.34 / **0.53**	0.30 / **0.48**	**0.44 / 0.75**

Figure 3. Validation of the Social STGCNN with respect to other published works. Best results are shown in bold characters. The first metric is the average displacement error over the trajectory (6), while the second metric is the final displacement error (7)

5.3 Limits

While the Social STGCNN algorithm attemps to encode regular human behaviors including merging and avoiding collitions, it is not entirely free of errors. In Figure 4, we show that in some situations the predictions of Social STGCNN are not realistic (third row)

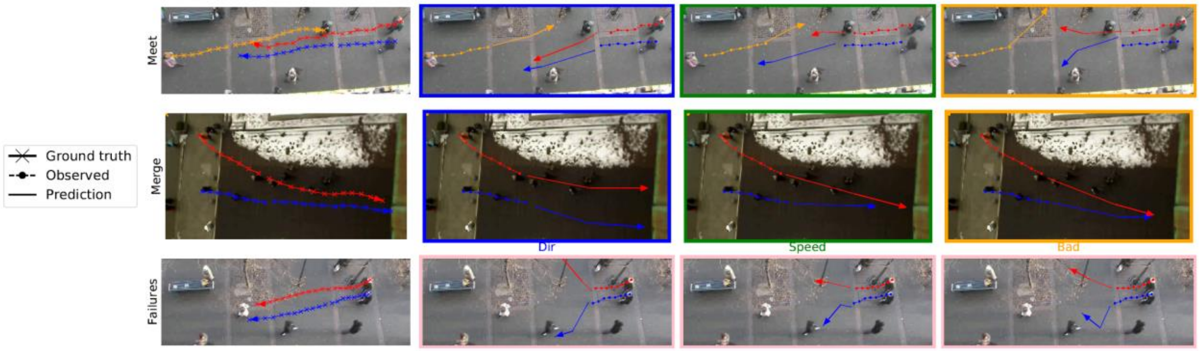


Figure 4. The first column is the ground truth, while the other columns illustrate samples from our model. The first two rows show two different scenarios where pedestrians merge into a direction or meet from opposite directions. The second and third columns show changes in speed or direction in samples from our model. The last column shows undesired behaviors. The last row show failed samples

Chapter 6. Conclusion and future work

In this report, we introduce a graph-based spatio-temporal setup for pedestrian trajectory prediction which improves over previous methods on several key aspects, including prediction error. By applying a specific kernel function in the weighted adjacency matrix together with the model design, Social-STGCNN outperforms state-of-art models over four different publicly available datasets. The weighted adjacency matrix allows the learning of well known pedestrian behaviors such as merging and avoiding collisions when on a collision course. However, the predion can sometimes fail.

In the future, we intend to extend Social-STGCNN to multi-modal settings that involve other moving objects including bicycles and cars. We also intend to work on the problem of predicting the trajectories of partially controlled agents (for example drivers or pedestrians that receive guidance information).

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[2] Haoyu Bai, Shaojun Cai, Nan Ye, David Hsu, and Wee Sun Lee. Intention-aware online pomdp planning for autonomous driving in a crowd. In *2015 ieee international conference on robotics and automation (icra)*, pages 454–460. IEEE, 2015.

[3] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[4] Rianne van den Berg, Thomas N Kipf, and Max Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, 2017.

[5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[6] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[8] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[9] Thomas N Kipf and Max Welling. Variational graph autoencoders. *arXiv preprint arXiv:1611.07308*, 2016.

[10] Vineet Kosaraju, Amir Sadeghian, Roberto Mart ń-Mart ń, Ian Reid, S Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *arXiv preprint arXiv:1907.03395*, 2019.

[11] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer graphics forum*, volume 26, pages 655–664. Wiley Online Library, 2007.

[12] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9267–9276, 2019.

[13] Jiachen Li, Hengbo Ma, and Masayoshi Tomizuka. Conditional generative neural system for probabilistic trajectory prediction. *arXiv preprint arXiv:1905.01631*, 2019.

[14] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019.

[15] Matthias Luber, Johannes A Stork, Gian Diego Tipaldi, and Kai O Arras. People tracking with human motion predictions from social forces. In *2010 IEEE International Conference on Robotics and Automation*, pages 464–469. IEEE, 2010.

[16] Yuanfu Luo, Panpan Cai, Aniket Bera, David Hsu, Wee Sun Lee, and Dinesh Manocha. Porca: Modeling and planning for autonomous driving among many pedestrians. *IEEE Robotics and Automation Letters*, 3(4):3418–3425, 2018.

[17] Huynh Manh and Gita Alaghband. Scene-lstm: A model for human trajectory prediction. *arXiv preprint arXiv:1808.04018*, 2018.

[18] Kohei Morotomi, Masayuki Katoh, and Hideaki Hayashi. Collision position predicting device, Sept. 30 2014. US Patent 8,849,558.

[19] Mehdi Moussaïd, Niriaska Perozo, Simon Garnier, Dirk Helbing, and Guy Theraulaz. The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PloS one*, 5(4):e10047, 2010.

[20] Basam Musleh, Fernando García, Javier Otamendi, José Mª Armingol, and Arturo De la Escalera. Identifying and tracking pedestrians based on sensor fusion and motion stability predictions. *Sensors*, 10(9):8028–8053, 2010.

[21] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE, 2009.

[22] Pongsathorn Raksincharoensak, Takahiro Hasegawa, and Masao Nagai. Motion planning and control of autonomous driving intelligence system based on risk potential optimization framework. *International Journal of Automotive Engineering*, 7(AVEC14):53–60, 2016.

[23] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019.

[24] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer, 2018.

[25] Jean-Philippe Vert, Koji Tsuda, and Bernhard Schölkopf. A primer on kernel methods. *Kernel methods in computational biology*, 47:35–70, 2004.

[26] Travis Williams and Robert Li. Wavelet pooling for convolutional neural networks. In *International Conference on Learning Representations*, 2018.

[27] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[28] Masahiro Yasuno, Noboru Yasuda, and Masayoshi Aoki. Pedestrian detection and tracking in far infrared images. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 125–125. IEEE, 2004.

[29] Lidan Zhang, Qi She, and Ping Guo. Stochastic trajectory prediction with social graph network. *arXiv preprint arXiv:1907.10233*, 2019.

[30] Pu Zhang, Wanli Ouyang, Pengfei Zhang, Jianru Xue, and Nanning Zheng. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12085–12094, 2019.