A Set of Control Points Conditioned Pedestrian Trajectory Prediction*

Inhwan Bae and Hae-Gon Jeon

AI Graduate School, GIST, Gwangju, South Korea
inhwanbae@gm.gist.ac.kr and haegonj@gist.ac.kr

Abstract. Predicting the trajectories of pedestrians in crowded conditions is an important task for applications like autonomous navigation systems. Previous studies have tackled this problem using two strategies. They (1) infer all future steps recursively, or (2) predict the potential destinations of pedestrians at once and interpolate the intermediate steps to arrive there. However, these strategies often suffer from the accumulated errors of the recursive inference, or restrictive assumptions about social relations in the intermediate path. In this paper, we present a graph convolutional network-based trajectory prediction. Firstly, we propose a control point prediction that divides the future path into three sections and infers the intermediate destinations of pedestrians to reduce the accumulated error. To do this, we construct multi-relational weighted graphs to account for their physical and complex social relations. We then introduce a trajectory refinement step based on a spatio-temporal and multi-relational graph. In experiments, the proposed network achieves state-of-the-art performance on various real-world benchmarks.

Keywords: Pedestrian Trajectory Prediction · Multi-agent Forecasting

1 Introduction

Predicting the future trajectories of humans in crowds is an important task, especially for social robots, autonomous navigation, and surveillance systems. However, this task is challenging because such predictions require considering the desired destinations of each pedestrian, and the social norms of other moving agents, simultaneously.

Early works have attempted to capture social interactions using handcrafted Langevin equations, however, they often fail to model the complex social interactions that occur in crowded scenes. The recent development of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) combines with social pooling [6] and social attention [15], and has improved understanding of the social interactions among pedestrians. However, these approaches still suffer from severe errors in final destination, because they accumulate the errors inherent to problems in the recursive predictions.

* This paper is a summary presentation of the paper which has been published in AAAI'23 by request of the IW-FCV2023 program committee to share the research results.

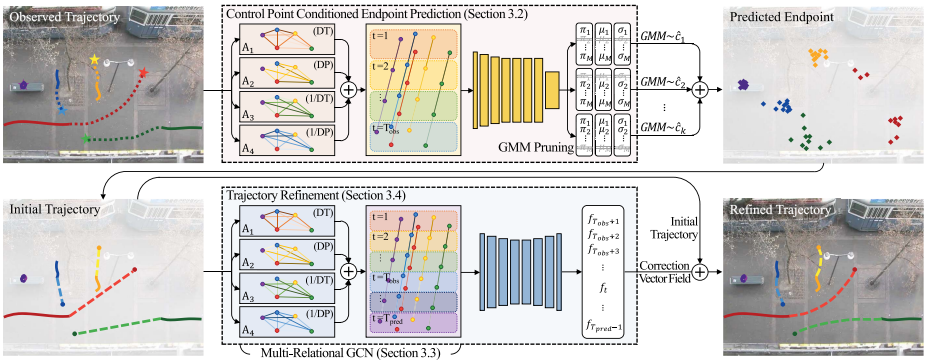

Fig. 1. An overview of our Graph-TERN. First, a control point prediction module takes the observed trajectory $S_{1:T_{obs}}$, and then constructs a multi-relational pedestrian graph. With a spatio-temporal aggregation using GCN and CNN, we can predict hypothetical endpoints $\hat{E}$ through a summation of a set of randomly sampled control points $\hat{C}$. Second, a trajectory refinement modules takes the predicted endpoint $\hat{E}$ and the observed path $S_{1:T_{obs}}$ to predict a correction vector field. A refined path $\hat{S}_{T_{obs}+1:T_{pred}}$ is obtained by summing the initial trajectory with the correction vector field.

To overcome this issue, Endpoint conditioned trajectory prediction is introduced [2, 4, 3], which infers the hypothetical arrival points and then interpolates their paths like vehicle navigation systems. Although these methods show promising performance improvements, issues remained. (1) Long-term predictions are performed without considering events occurring in the intermediate steps, and (2) social interactions are not regarded in endpoint prediction.

In this paper, we propose a Graph-based pedestrian Trajectory Estimation and Refinement Network (Graph-TERN) using a set of control points that combines the advantages of both step-by-step methods and endpoint conditioned methods. Firstly, we divide each pedestrian's future path into three sections and infer each stochastic goal, called control points. Secondly, we generate realistic future paths by introducing a refinement module that yields correction vector fields. Lastly, we design a multi-relational GCN operator to take account of complex social interactions in both the control point prediction and the trajectory refinement. By effectively incorporating the three modules, our model achieves state-of-the-art results using a variety of public pedestrian trajectory prediction benchmarks.

2 Proposed Method

Graph-TERN consists of two key components: (1) learning the probabilistic distribution for sampling endpoint candidates based on the control point; (2) yielding socially acceptable path prediction using a refinement module. Using an MRGCN framework, we develop a model that can successfully predict future trajectories while considering complex social relations. The overall framework is shown in Figure 1.

2.1 Preliminaries

Problem Definition. Pedestrian trajectory prediction attempts to determine future position sequences from observed position sequences for all agents in a scene. Suppose that there are N pedestrians in a scene at specific time t, and the corresponding positions of each pedestrian $n \in \{1, ..., N\}$ can be represented as $p_t^n = (x_t^n, y_t^n)$. The trajectory sequence from the first time frame to the observed time T_{obs} can be denoted as $S_{1:T_{obs}}^n = \{p_t^n \in \mathbb{R}^2 | t \in \mathbb{N}, 1 \leq t \leq T_{obs}\}$. The consecutive prediction time frames are represented as $S_{T_{obs}+1:T_{pred}}$.

An additional goal of this work is to estimate a set of potential destinations for each pedestrian, called endpoint E^n. The endpoint $\hat{E}$ can be predicted with the observed sequence $\hat{E}^n = \hat{p}_{T_{pred}}^n | S_{1:T_{obs}}$, then the future trajectories $\hat{S}_{T_{obs}+1:T_{pred}}$ can be inferred from the observed sequence $S_{1:T_{obs}}$ and the predicted endpoint $\hat{E}$.

Graph Convolutional Network. In general, the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represents a set of nodes $\mathcal{V}$ and edges $\mathcal{E}$. In a pedestrian graph, the spatio-temporal graph $\mathcal{G}$ consists of a pedestrian node $\mathcal{V} = \{p_t^n | n, t \in \mathbb{N}, 1 \leq n \leq N, 1 \leq t \leq T\}$ and a set of spatial and temporal edges $\mathcal{E} = \mathcal{E}_t \cup \mathcal{E}_n$. The spatial edge $\mathcal{E}_t = \{a_t^{i,j} | i, j \in \mathbb{N}, 1 \leq i, j \leq N\}$ represents a spatial relation for each pedestrian at a specific time t, and the temporal edge $\mathcal{E}_n = \{a_n^{i,j} | i, j \in \mathbb{N}, 1 \leq i, j \leq T\}$ represents the temporal relation of each pedestrian n within an observed sequence. Node features are aggregated with both spatial and temporal dimensions using GCNs and CNNs [16, 11]. With the node feature $H = \{h_t^n | n, t \in \mathbb{N}, 1 \leq n \leq N, 1 \leq t \leq T_{obs}\}$ and adjacency matrix $A = \{a_t^{i,j} | i, j, t \in \mathbb{N}, 1 \leq i, j \leq N, 1 \leq t \leq T_{obs}\}$, the GCN feature update rule is defined as $H' = \sigma(\hat{A}HW)$. Here, W and $\hat{A}$ indicate the learnable weight matrix and the normalized form with the formula $\hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$, respectively. We denote the self-loop added adjacency matrix as $\tilde{A} = A + I$ and diagonal node degree matrix as $\tilde{D}$ from $\tilde{A}$.

2.2 Control Point Conditioned Endpoint Prediction

Graph Control Point Prediction. The key idea of our model is to use multiple control points when inferring potential endpoints. First, we define a set of control points C based on a displacement in one section, which is equally divided into future sequences $S_{T_{obs}+1:T_{pred}}$ in Figure 2, which is formulated as:

$$\begin{aligned} C^n &= \left\{ c_k^n = p_{T_{obs}+\tau \times k}^n - p_{T_{obs}+\tau \times (k-1)}^n \right\} \\ &\text{for } \forall k \in \{1, ..., K\}, \quad \tau = \frac{T_{pred} - T_{obs}}{K}, \end{aligned} \tag{1}$$

Next, we present a control point prediction module. We update feature maps for the observed input sequence using a spatio-temporal and multi-relational GCN in Figure 1. We then use a multivariate Gaussian mixture model (GMM) to sample the 2D displacements of a set of control points in a Mixture Density Network (MDN). In contrast to previous works [10], we incorporate our control point prediction into a GCN framework, and in this way, our model provides a socially compliant endpoint that considers intermediate social interactions.

GMM Pruning. Public pedestrian trajectory datasets contain abnormal behaviors of agents. Since statistical models need to allocate a portion of its capacity to ensure the abnormal cases, it is left with relatively less capacity for generating realistic paths. These abnormal cases are considered as out-of-distribution samples drawn far away from the training distribution statistically, and leads to performance drops. The truncation trick is widely used to restrict the distribution of samples, it limits the distributions of reasonable control points as well in the GMM-based model. To address this issue, we devise a GMM pruning which cuts off a lower half of the bi-variate Gaussian based on predicted mixing coefficients as follows:

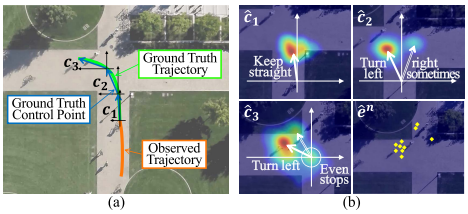

Fig. 2. A set of control points prediction. (a) When a person turns left at the cross-road, three control points are defined based on the displacement. (b) Examples of predicted distributions for the control points and endpoint sampling.

$$M^* = \left\lfloor \frac{M}{2} \right\rfloor, \quad z^* = \underset{z' \subset z, |z'|=M^*}{\operatorname{argmax}} \sum_{z'^{\pi} \subset z'} z'^{\pi}. \tag{2}$$

Through the GMM pruning, potential control point candidates can be assigned to effectively feasible areas.

Endpoint Sampling. The final endpoint $\hat{e}$ is determined by summing the set of control points $\hat{C}$, which is sampled through the probabilistic process $\hat{e}^n = p^n_{T_{obs}} + \sum_{k=1}^{K} \hat{c}^n_k$. Following the previous study [6], we sample the $L = 20$ endpoints $\hat{E}^n = \{\hat{e}^n_l | l, n \in \mathbb{N}, 1 \leq l \leq L, 1 \leq n \leq N\}$ which represent multi-modality, and feed them into a trajectory refinement module in Section 2.4.

2.3 Multi-Relational Pedestrian Graph

While the GCN has the advantage of imposing physical constraints, conventional GCN-based models use a single relation edge, which makes capturing social relations limited. Due to this reason, the GCN-based approaches have gained less interest than those of the GAT-based approaches whose multi-head attention allows it. In this work, we fully take advantage of the GCN framework by overcoming the limitation through a multi-relational-based kernel function to produce each relational adjacent matrix as:

$$H' = \sigma\left[\mathrm{CNN}\Big(\sum\nolimits_{r=1}^{R} \hat{A}_r H^T W_r\Big)^T + H\right], \tag{3}$$

where R is the number of elements in a set of relations $\mathcal{R} = \{\mathit{Distance}, \mathit{Displacement}, 1/\mathit{Distance}, 1/\mathit{Displacement}\}$, $\hat{A}_r$ is the normalized term of $A_r = \{a^{i,j,r} \in \mathbb{R} | i, j, r \in \mathbb{N}, 1 \leq i, j \leq N, 1 \leq r \leq R\}$, and W_r is a learnable weight matrix. Through this, Our multi-relational graph deals with obstacles, stop and go motion, and group following in very challenging situations.

2.4 Trajectory Refinement

Guided Endpoint Sampling. To jointly train the control point prediction module and trajectory refinement module, we need to decouple them. We define a rule to limit the predicted endpoints at training time. We divide the predicted endpoints into a positive set Υ^+ and a negative set Υ^- using the adaptive threshold parameter Γ.

$$\begin{aligned} \Upsilon^{n+} &= \{\hat{e}_l^n \,|\, \|\hat{e}_l^n - e^n\| \leq \Gamma\} \\ \Upsilon^{n-} &= \{\hat{e}_l^n \,|\, \|\hat{e}_l^n - e^n\| > \Gamma\} \end{aligned} \tag{4}$$

$$\text{for} \quad \forall l \in \{1, ..., L\}, \quad \Gamma = \frac{\|p_{T_{obs}}^n - p_1^n\|}{T_{obs} \times \gamma},$$

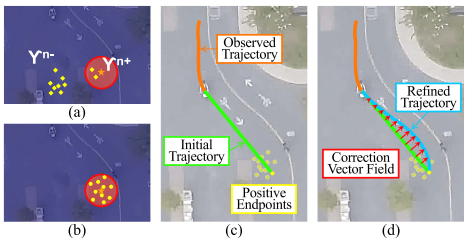

Fig. 3. An example of trajectory refinement. (a) Endpoint candidates are classified into the positive and negative set. (b) Guided endpoints are randomly sampled. (c) The initial trajectory is predicted by linearly interpolating between the observed trajectory and the endpoints. (d) After that, the trajectory is refined by adding a correction vector field in the initial trajectory.

To ensure that the model converges stably, we only back-propagate gradients for the positive sets using a valid mask Ψ. Here, the element of the valid mask $[\Psi]_{n,l}$ is defined as $[\Psi]_{n,l} = \psi_l^n = \{1 \;\text{ if }\; \hat{e}_l^n \in \Upsilon^{n+}, \;\text{ otherwise }\; 0\}$. During the initial training phase, the number of positive sets might be extremely small because the candidates are broadly spread. To address this issue, we additionally sample the guided endpoints within the range Γ of the ground-truth endpoint.

Initial Trajectory Prediction. The purpose of establishing an initial trajectory based on the guided endpoints is to make our trajectory refinement module tractable. The simplest way to do this is to connect these control points through linear interpolation. However, we observe that the use of a set of control points in the initial trajectory prediction acts as a hard constraint, even though it is helpful to infer accurate destinations of pedestrians. Therefore, we first generate a single initial trajectory $\tilde{S}_{T_{obs}+1:T_{pred}}$ for one endpoint as below:

$$\begin{aligned} \tilde{p}_{t,l}^n =& p_{T_{obs}}^n + \frac{\hat{e}_l^n - p_{T_{obs}}^n}{T_{pred} - T_{obs}} \times (t - T_{obs}) \\ & \tilde{S}_{T_{obs}+1:T_{pred},l} = \{\tilde{p}_{t,l}^n\} \\ \text{for} \quad & \forall t \in \{T_{obs}+1, ..., T_{pred}\}, \\ & \forall l \in \{1, ..., L\}, \quad \forall n \in \{1, ..., N\}. \end{aligned} \tag{5}$$

Graph Trajectory Refinement. As a next step, we present a novel refinement module to yield an accurate trajectory from the observed trajectory $S_{1:T_{obs}}$ and the initial trajectory $\tilde{S}_{T_{obs}+1:T_{pred}}$. By concatenating the two trajectories along with the time axis, we can aggregate the social interactions for all time frames using the MRGCN. For social interactions, the correction vector field is computed and the final refined trajectory can be obtained as below:

Table 1. Comparison of our Graph-TERN with other state-of-the-art methods on ETH/UCY dataset (ADE/FDE, Unit: meter). The mark † means that the common data-loader in [6] are used. **Bold**: Best, Underline: Second best.

Model	Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Linear Regression	-	1.33 / 2.94	0.39 / 0.72	0.82 / 1.59	0.62 / 1.21	0.77 / 1.48	0.79 / 1.59
Social-LSTM [1]	2016	1.09 / 2.35	0.79 / 1.76	0.67 / 1.40	0.47 / 1.00	0.56 / 1.17	0.72 / 1.54
Social-GAN [6]	2018	0.87 / 1.62	0.67 / 1.37	0.76 / 1.52	0.35 / 0.68	0.42 / 0.84	0.61 / 1.21
STGAT [7]	2019	0.65 / 1.12	0.35 / 0.66	0.52 / 1.10	0.34 / 0.69	0.29 / 0.60	0.43 / 0.83
Social-STGCNN [11]	2020	0.64 / 1.11	0.49 / 0.85	0.44 / 0.79	0.34 / 0.53	0.30 / 0.48	0.44 / 0.75
PECNet† [10]	2020	0.65 / 1.13	0.22 / 0.38	0.35 / 0.57	0.25 / 0.45	0.18 / 0.31	0.33 / 0.57
Trajectron++† [15]	2020	0.61 / 1.03	0.20 / 0.28	0.30 / 0.55	0.24 / 0.41	0.18 / 0.32	0.31 / 0.52
Causal-STGAT [5]	2021	0.60 / 0.98	0.30 / 0.54	0.52 / 1.10	0.32 / 0.64	0.28 / 0.58	0.40 / 0.77
TPNMS [9]	2021	0.52 / 0.89	0.22 / 0.39	0.55 / 1.13	0.35 / 0.70	0.27 / 0.56	0.38 / 0.73
Causal-STGCNN [5]	2021	0.64 / 1.00	0.38 / 0.45	0.49 / 0.81	0.34 / 0.53	0.32 / 0.49	0.43 / 0.66
SGCN [16]	2021	0.63 / 1.03	0.32 / 0.55	0.37 / 0.70	0.29 / 0.53	0.25 / 0.45	0.37 / 0.65
LBEBM† [13]	2021	0.62 / 1.16	0.19 / 0.35	0.37 / 0.67	0.23 / 0.43	0.19 / 0.36	0.32 / 0.59
DMRGCN [2]	2021	0.60 / 1.09	0.21 / 0.30	0.35 / 0.63	0.29 / 0.47	0.25 / 0.41	0.34 / 0.58
STT [12]	2022	0.54 / 1.10	0.24 / 0.46	0.57 / 1.15	0.45 / 0.94	0.36 / 0.77	0.43 / 0.88
Graph-TERN	-	**0.42 / 0.58**	**0.14 / 0.23**	**0.26 / 0.45**	**0.21 / 0.37**	**0.17 / 0.29**	**0.24 / 0.38**

$$\begin{aligned} \hat{p}^n_{t,l} &= \tilde{p}^n_{t,l} + f^n_{t,l} \\ \hat{S}_{T_{obs}+1:T_{pred},l} &= \{\hat{p}^n_{t,l}\} \cup \hat{E} \\ \text{for} \quad \forall t &\in \{T_{obs}+1, ..., T_{pred}-1\}, \\ \forall l &\in \{1, ..., L\}, \quad \forall n \in \{1, ..., N\}. \end{aligned} \tag{6}$$

Unlike existing methods which use social interactions based only on observations, our refinement module allows a more complex social relation because our MRGCN captures such relations even with the interpolated points and the endpoint.

2.5 Loss Functions

We maximize an expectation to train the control point prediction module. We sum the probabilistic density functions of all the predicted control point distributions and pedestrians. The loss function Θ_w is defined as:

$$\Theta_w = \sum_{n=1}^{N} \sum_{k=1}^{K} -\log \left[\sum_{m=1}^{M} \hat{\pi}^n_{m,k} \frac{\exp\left(-\frac{(\boldsymbol{c}^n_k - \hat{\boldsymbol{\mu}}^n_{m,k})^2}{2(\hat{\boldsymbol{\sigma}}^n_{m,k})^2}\right)}{\sqrt{2\pi}\ \hat{\boldsymbol{\sigma}}^n_{m,k}} \right] \tag{7}$$

In addition, we minimize the trajectory refinement loss Θ_r. The loss is based on a mean square error (MSE) of an average displacement between a refined trajectory and a ground truth trajectory, and is formulated as below:

$$\Theta_r = \sum_{n=1}^{N} \sum_{l=1}^{2L} \sum_{t=T_{obs}+1}^{T_{pred}-1} \psi^n_l \Big[\big(x^n_{t,l} - \hat{x}^n_{t,l}\big)^2 + \big(y^n_{t,l} - \hat{y}^n_{t,l}\big)^2 \Big] \tag{8}$$

Finally, the loss function Θ of the entire network is defined as a weighted sum of the control point loss and the refinement loss: $\Theta = \Theta_w + \lambda\Theta_r$, where λ is a scale factor between the control point prediction error and the trajectory refinement error, and is empirically set to $\lambda = 1$

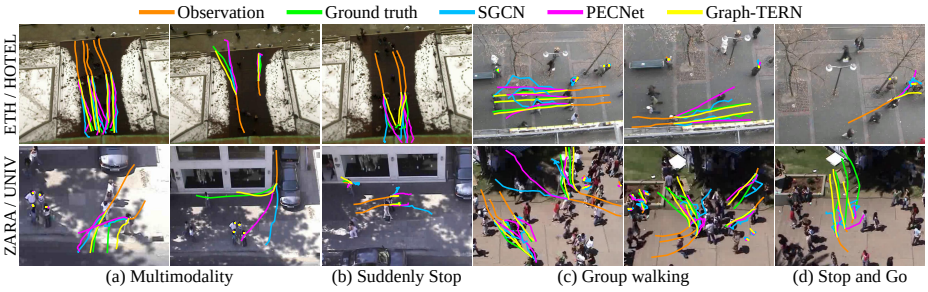

Fig. 4. Visualization of prediction results. We compare Graph-TERN, SGCN and PECNet, whose results are reproduced with the pre-trained network. To aid visualization, trajectories with the best ADE on 20 samples are reported.

3 Experiments

We evaluate our Graph-TERN using two real-world public datasets: ETH [14] and UCY [8]. In ETH and UCY datasets, there are five different scenes (ETH, HOTEL, UNIV, ZARA1, and ZARA2) with various complex social interactions such as collision avoidance, group movement, and people stopping. We follow the standard evaluation strategy.

We compared our Graph-TERN with other state-of-the-art works with two performance metrics: average displacement error (ADE) and final displacement error (FDE) in meter scale on both ETH and UCY datasets Table 1. The results indicate Graph-TERN provides the best performance on nearly all of the measures and datasets. Of particular note, in ETH and HOTEL set, there are many people who abruptly stop walking. Graph-TERN handles this case well by learning the probability of people stopping in the control point prediction as in Figure 2(b). For UNIV set with very crowded scenes, our multi-relational GCN synergizes well with the initial trajectory prediction and the refinement module by considering complex social relations. This infers accurate multimodal and stop-and-go predictions based on the well-estimated group movements in Figure 4.

4 Conclusion

We present a set of control points prediction and a refinement network for pedestrian trajectory prediction. The control point prediction allows the accurate computation of the final destinations of pedestrians and the refinement provides a socially acceptable trajectory. By incorporating a MRGCN, our model achieves SOTA results by modeling complex social interactions in real-world scenes[1].

[1] [Remarks] This paper is a summary presentation of the paper which has been published in AAAI'23 by request of the IW-FCV2023 program committee to share the research results.

References

1. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
2. Bae, I., Jeon, H.G.: Disentangled multi-relational graph convolutional network for pedestrian trajectory prediction. Proceedings of the AAAI Conference on Artificial Intelligence (AAAI) (2021)
3. Bae, I., Park, J.H., Jeon, H.G.: Learning pedestrian group representations for multi-modal trajectory prediction. In: Proceedings of European Conference on Computer Vision (ECCV) (2022)
4. Bae, I., Park, J.H., Jeon, H.G.: Non-probability sampling network for stochastic human trajectory prediction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
5. Chen, G., Li, J., Lu, J., Zhou, J.: Human trajectory prediction via counterfactual analysis. In: Proceedings of International Conference on Computer Vision (ICCV) (2021)
6. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
7. Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of International Conference on Computer Vision (ICCV) (2019)
8. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. Computer Graphics Forum **26**(3), 655–664 (2007)
9. Liang, R., Li, Y., Li, X., Tang, Y., Zhou, J., Zou, W.: Temporal pyramid network for pedestrian trajectory prediction with multi-supervision. Proceedings of the AAAI Conference on Artificial Intelligence (AAAI) (2021)
10. Mangalam, K., Girase, H., Agarwal, S., Lee, K.H., Adeli, E., Malik, J., Gaidon, A.: It is not the journey but the destination: Endpoint conditioned trajectory prediction. In: Proceedings of European Conference on Computer Vision (ECCV) (2020)
11. Mohamed, A., Qian, K., Elhoseiny, M., Claudel, C.: Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
12. Monti, A., Porrello, A., Calderara, S., Coscia, P., Ballan, L., Cucchiara, R.: How many observations are enough? knowledge distillation for trajectory forecasting. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
13. Pang, B., Zhao, T., Xie, X., Wu, Y.N.: Trajectory prediction with latent belief energy-based model. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
14. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: Proceedings of International Conference on Computer Vision (ICCV) (2009)
15. Salzmann, T., Ivanovic, B., Chakravarty, P., Pavone, M.: Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In: Proceedings of European Conference on Computer Vision (ECCV) (2020)
16. Shi, L., Wang, L., Long, C., Zhou, S., Zhou, M., Niu, Z., Hua, G.: Sgcn: Sparse graph convolution network for pedestrian trajectory prediction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)