# Disentangled Multi-Relational Graph Convolutional Network for Pedestrian Trajectory Prediction

**Inhwan Bae and Hae-Gon Jeon**
Gwangju Institute of Science and Technology (GIST)
inhwanbae@gm.gist.ac.kr and haegonj@gist.ac.kr

## Abstract

Pedestrian trajectory prediction is one of the important tasks required for autonomous navigation and social robots in human environments. Previous studies focused on estimating social forces among individual pedestrians. However, they did not consider the social forces of groups on pedestrians, which results in over-collision avoidance problems. To address this problem, we present a Disentangled Multi-Relational Graph Convolutional Network (DMRGCN) for socially entangled pedestrian trajectory prediction. We first introduce a novel disentangled multi-scale aggregation to better represent social interactions, among pedestrians on a weighted graph. For the aggregation, we construct the multi-relational weighted graphs based on distances and relative displacements among pedestrians. In the prediction step, we propose a global temporal aggregation to alleviate accumulated errors for pedestrians changing their directions. Finally, we apply DropEdge into our DMRGCN to avoid the over-fitting issue on relatively small pedestrian trajectory datasets. Through the effective incorporation of the three parts within an end-to-end framework, DMRGCN achieves state-of-the-art performances on a variety of challenging trajectory prediction benchmarks.

## 1 Introduction

Pedestrian trajectory prediction attempts to predict the paths of persons based on their previous steps, and is an important part of autonomous navigation and social robot platforms. It assumes that a person walks toward a destination, while taking interactions with other people into account. To be specific, pedestrians move together with their companions by selecting an optimal route to avoid collisions, and tend to follow the footsteps of their surrounding flows. Pioneering works in (Helbing and Molnar 1995; Mehran, Oyama, and Shah 2009) have modeled these human-human interactions as social forces. However, the social force relies on parametric models, which have a difficult time generalizing complex interpersonal relations.

The adoption of convolutional neural networks (CNNs) has recently alleviated the generalization issue. A work in (Alahi et al. 2016) introduced a Social-LSTM that predicts human trajectories using recurrent neural networks (RNNs) with social pooling to model the interactions of neighboring pedestrians. In (Gupta et al. 2018), a generative adversarial networks (GANs)-based encoder-decoder framework was proposed with a global pooling mechanism to learn the social norm. However, we observe that important features needed to learn the trajectories sometimes leak through the heuristic pooling methods.

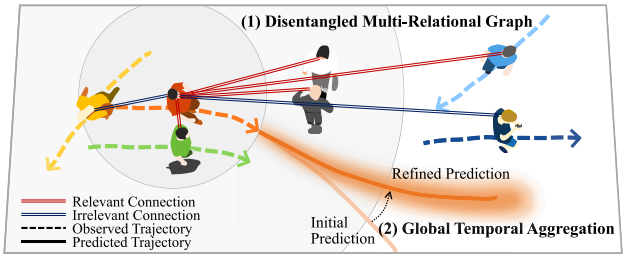


Figure 1: An illustration of the DMRGCN. The social interactions and temporal relations of each pedestrian are represented by pedestrian graphs. Our disentangled multi-relational graph establishes relevant connections between pedestrians well. Our global temporal aggregation then learns to correct the initial prediction, which is associated with over-avoidance.

An alternative approach to capture human interactions with relevant neighbors involves learning a graph representation, which consists of a set of nodes and edges, (Kosaraju et al. 2019; Mohamed et al. 2020). In pedestrian trajectory prediction, the node represents each pedestrian in a scene and the edges correspond to distances from other pedestrians. The use of a graph convolutional network (GCN) enables it to better learn the physical and the social interactions among pedestrians. Although the GCN-based works have shown promising performance for predicting further trajectories, most of them suffer from two limitations: (1) only simple social relationship like collision avoidance is aggregated; (2) an inevitable error in final destination occurs because modeling social norms is not suitable for determining the end-points of pedestrians in the last frame.

In this paper, we present a Disentangled Multi-Relational Graph Convolutional Network (DMRGCN), a GCN-based

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
Corresponding author: Hae-Gon Jeon

socially entangled pedestrian trajectory prediction in Figure 1. Our DMRGCN consists of three parts: disentangled multi-scale aggregation with a multi-relational graph, global temporal aggregation and DropEdge application to pedestrian trajectory prediction.

In contrast to previous GCN-based approaches (Mohamed et al. 2020), DMRGCN models sophisticated social relationships. This is made possible through the disentangled multi-scale aggregation on a multi-relational graph, which represents distances and relative displacements among pedestrians. This is inspired by action recognition (Liu et al. 2020) and aims to remove redundant dependencies of the graphs. We also compensate for the final destination error induced by over-avoidance, using our global temporal aggregation. We extract a motion feature at an intermediate layer of a time prediction CNN, and then sum up the feature with its output. Lastly, we apply DropEdge (Rong et al. 2020) into our DMRGCN, which randomly removes a certain number of edges from the input graph at each training epoch, to avoid the over-fitting issue on relatively small pedestrian trajectory datasets. Through the effective incorporation of these designs, our model outperforms previous models of trajectory prediction accuracy.

## 2 Related Work

In this section, we will briefly review a variety of methods used for pedestrian trajectory prediction.

**Pedestrian Trajectory Prediction.** Pioneering works (Helbing and Molnar 1995; Pellegrini et al. 2009; Mehran, Oyama, and Shah 2009; Yamaguchi et al. 2011) used handcrafted energy potentials based on social forces. Since then, the state-of-the-art in pedestrian trajectory prediction has advanced with the introduction of both CNNs and RNNs.

Social-LSTM (Alahi et al. 2016) proposes a RNN model with social pooling which aggregates neighbor pedestrians on a grid. Social-Attention (Vemula, Muelling, and Oh 2018), SR-LSTM (Zhang et al. 2019) and SFT (Fernando et al. 2018) extend the capability of social pooling with new pooling schemes, which use not only people inside the grid but also all pedestrians, with weighted importance computed by attention modules. A work in (Shi et al. 2020) aggregated the relationship of a pedestrian with others using an attention module, and predicted the coordinates that the pedestrian would take next using a Gaussian mixture model. Social-GAN (Gupta et al. 2018) introduced a generative model to predict a socially-acceptable path. Its generator recursively forecasts a trajectory in a multi-modal way, and the discriminator classifies whether the predicted paths are real or fake. SoPhie (Sadeghian et al. 2019) captures both human-human and human-environment interactions by introducing a physical and a social attention, respectively. In (Sun, Zhao, and He 2020), a reciprocal learning that enforces consistency between forward and backward path prediction showed reasonable performances as well.

For more information, we redirect the readers to (Bartoli et al. 2018; Rehder and Kloeden 2015; Rehder et al. 2018; Liang et al. 2019) for conditioned trajectory prediction, which is outside the scope of this paper.

**Graph neural network-based approach.** With the success of graph neural networks in various applications for modeling relations such as node classification (Kipf and Welling 2017; Hamilton, Ying, and Leskovec 2017; Xu et al. 2019; Veličković et al. 2018; Li et al. 2019b) and action recognition (Yan, Xiong, and Lin 2018; Li et al. 2019c; Shi et al. 2019b,a; Li et al. 2019a; Liu et al. 2020), it became clear that social relations can be also represented on graphs, which makes pedestrian trajectory prediction more tractable (Huang et al. 2019; Kosaraju et al. 2019; Mohamed et al. 2020).

In (Veličković et al. 2018), a graph attention network (GAT) was applied in a self-attention-based architecture by implicitly assigning the importance of each node. For applications to pedestrian trajectory prediction, graph structures were directly used to better learn physical and social interactions between pedestrians in (Huang et al. 2019). Social-BiGAT (Kosaraju et al. 2019) incorporated GAT into BicycleGAN (Zhu et al. 2017) to adjust the latent vectors for each pedestrian, removing unnecessary degradation of the path prediction. In (Liang et al. 2020), GAT is used on the 2D grid graph for multiple plausible destination predictions.

As a recent progress, Social-STGCNN (Mohamed et al. 2020) aggregated motion information using GCN on a spatio-temporal graph made by stacking graphs at each time domain, whose edges are weighed by a displacement-based kernel function. The temporal prediction is then carried out at once by a temporal convolutional network (TCN) (Bai, Kolter, and Koltun 2018). Compared with Social-STGCNN, our DMRGCN also employs a GCN mechanism, but it can learn sophisticated social relations between pedestrians using multi-scale aggregation. In addition, it is highly efficient for reducing the accumulated error over long sequences by the design of the global temporal aggregation.

## 3 Disentangled Multi-Relational Graph Convolutional Network

We present an end-to-end pedestrian trajectory prediction method. Our contributions are threefold: (1) a disentangled multi-scale aggregation to clearly distinguish between relevant pedestrians; (2) a multi-relational GCN to extract sophisticated social interaction in a scene; (3) a global temporal aggregation to compensate accumulated errors from over-avoidance. An overview of the proposed model is illustrated in Figure 2.

### 3.1 Preliminaries

**Problem Definition.** The pedestrian trajectory prediction problem involves determining future position sequences from observed position sequences for all people in a scene. Assuming that there are $N$ pedestrians in a scene, and the corresponding positions of each pedestrian $n \in \{1, ..., N\}$ at a specific time $t$ can be denoted as $p_t^n = (x_t^n, y_t^n)$. Using the observed time frame $T_{obs}$ and the total sequence frame $T_{pred}$, the full sequence of a pedestrian can be denoted as $S_{1:T_{pred}}^n = \{p_t^n \in \mathbb{R}^2 | t \in \mathbb{N}, 1 \leq t \leq T_{pred}\}$. We assume that the predicted coordinates $(\hat{x}_t^n, \hat{y}_t^n)$ are random variables.

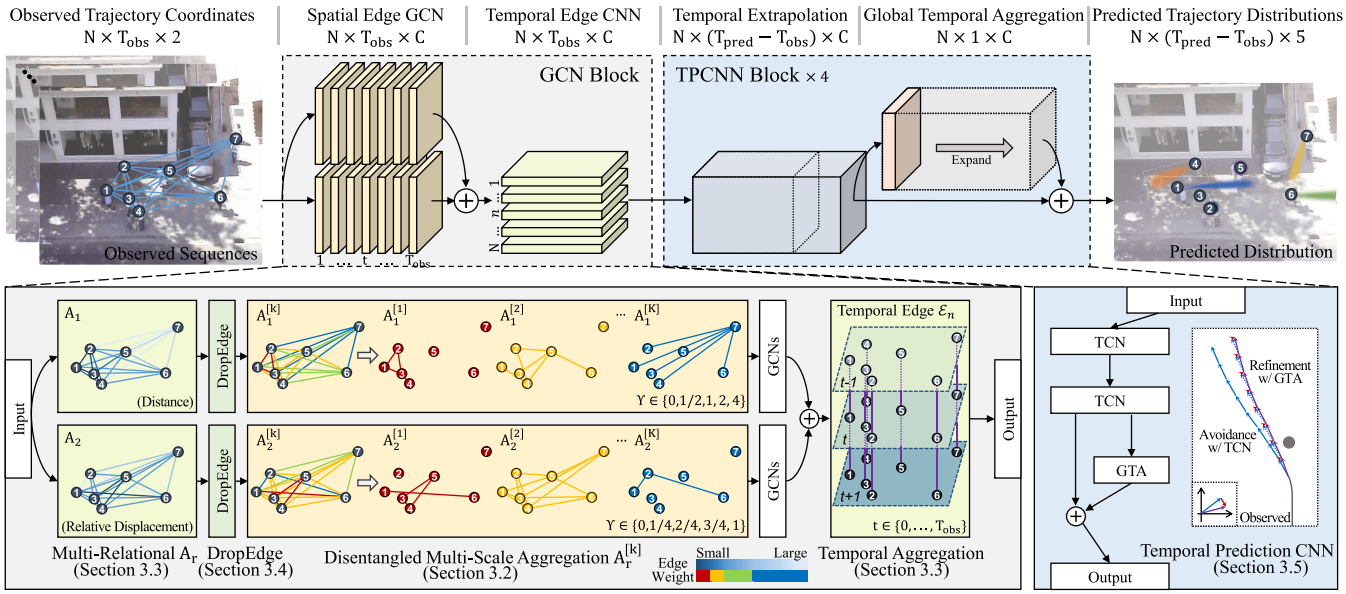


Figure 2: An overview of our DMRGCN model. Starting with 2-dimensional coordinates of $N$ pedestrians for $T_{obs}$ frames, we construct multi-relational graphs and then apply the disentangled multi-scale aggregation, followed by GCNs, on spatio-temporal pedestrian graphs. We then extrapolate future trajectories with TCNs and refine it with the GTA module.

With a given sequence $S_{1:T_{obs}}$ for all pedestrians, the probabilistic model is learned to estimate $S_{T_{obs}+1:T_{pred}}$ that the pedestrians travel after the last observed frame.

**Graph Convolutional Networks.** The Graph $\mathcal{G}$ is represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of $N$ nodes and $\mathcal{E}$ is the set of edges that represents connections between nodes. The spatio-temporal graph is represented by $\mathcal{G} = \{\mathcal{G}_1, ..., \mathcal{G}_{T_{obs}}\}$ as the set of attributes of the spatial graph $\mathcal{G}_t = \{(\mathcal{V}_t, \mathcal{E}_t)\}$ for all observed times $t \in \{1, ...T_{obs}\}$. The node feature $H = \{h_t^n \in \mathbb{R}^C | n, t \in \mathbb{N}, 1 \leq n \leq N, 1 \leq t \leq T_{obs}\}$ is a set of the pedestrian position $p_t^n$ at a specific time $t$ with the number of channel $C$. Its adjacency matrix $A = \{a_t^{i,j} \in \mathbb{R} | i, j, t \in \mathbb{N}, 1 \leq i, j \leq N, 1 \leq t \leq T_{obs}\}$ represents the physical relationships between pedestrians $i$ and $j$. The layer-wise GCN feature is updated as below:

$$H^{(l+1)} = \sigma(\hat{A} H^{(l)} W^{(l)}), \tag{1}$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ is a normalized form of $\tilde{A}$, and $\tilde{A} = A + I$ is a relation graph with an added self-loop. $\tilde{D}$ denotes the diagonal node degree matrix from $\tilde{A}$, $\sigma(\cdot)$ is a nonlinear activation function (PReLU in our implementation), and $W$ indicates a layer-wise learnable weight matrix.

For a multi-relational graph $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$ originally used to indicate relationships between entities (Marcheggiani and Titov 2017; Shi et al. 2019b; Li et al. 2019c), additional term $\mathcal{R}$ is used to represent the $R$ relations of edges. With these relations, an adjacency matrix expresses the relation information as $A_r = \{a^{i,j,r} \in \mathbb{R} | i, j, r \in \mathbb{N}, 1 \leq i, j \leq N, 1 \leq r \leq R\}$. The layer-wise feature update rule for multi-relational GCN is defined as

$$H^{(l+1)} = \sigma\left(\sum_{r=1}^{R} \hat{A}_r H^{(l)} W_r^{(l)}\right), \tag{2}$$

where $l$ is an index of the layers.

## 3.2 Disentangled Multi-Scale Aggregation

**Over-smoothing and biased weighting problems.** Even though graph-based approaches can represent arbitrary structures well, two problems limit their applicability for pedestrian trajectory prediction. First of all, they suffer from over-smoothing problems on node features. When constructing pedestrian graphs for crowded environments, the features are smoothed by the aggregation of lots of nodes.

Another problem comes from high-order social relations. In (Li et al. 2019c), multi-scale aggregation using k-hop neighborhoods was used for feature aggregation, where $k$-order polynomials of adjacency matrix $A^k$ for $k \in \{1, ..., K\}$. The GCN feature update rule with multi-scale aggregation is defined as

$$H^{(l+1)} = \sigma\left(\sum_{k=1}^{K} \hat{A}^k H^{(l)} W_k^{(l)}\right), \tag{3}$$

where $\hat{A}^k$ indicates the normalized term of the polynomial adjacency matrix $\tilde{A}^k$. In this way, rich representations can be learned by creating relations between both far and near neighbors with separate weights.

We tried to directly adapt this idea for pedestrian trajectory prediction, but there is a problem due to inherent differences in the pedestrian graph. The weight bias problem occurs when powering the adjacency matrix for multi-scale aggregation operations. The edge weight increases exponentially when there are lots of strong connections on the k-hop neighborhoods. As shown in Figure 3(Left) and (Top right), the group1 with large number of members has strong connections between them. The weights are only biased among

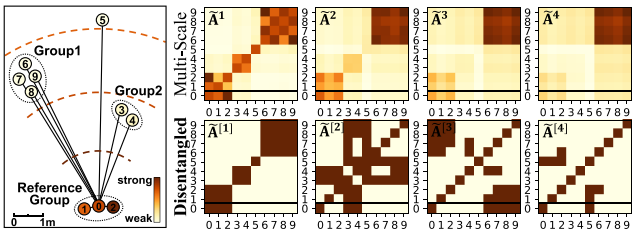


Figure 3: An illustration of the multi-scale aggregation method and our proposed disentangled multi-scale aggregation. The adjacency matrix is constructed using the inverse $L_2$ distance between nodes suggested in (Mohamed et al. 2020), so that strong weights are formed between close pedestrians. (Left) An example with 10 pedestrians. (Top right) Through adjacency powering, the connection is biased toward the group1. (Bottom right) Our novel disentangled multi-scale aggregation shows that all neighbors are strongly connected with a set of sub-graphs.

members in the group1, even though the reference group should consider both the group1 and group2 in practical.

In (Liu et al. 2020), a disentangled multi-scale aggregation for unweighted sparse graph was proposed using the concept of shortest distance to generalize an adjacency matrix to further neighborhoods. However, this is not suitable for pedestrian trajectory prediction, where all pedestrian nodes are connected for only 1-hop ($A^{[1]} = \mathbb{1}$ and $A^{[k]} = \mathbb{0}$ for $k \in \{2, ...\}$), because the pedestrian graph is a complete graph (Mohamed et al. 2020).

**Disentangling pedestrian interactions.** We present a novel disentangled multi-scale aggregation of social relations on a weighted graph. To do this, we first make a set of graphs with edges weighted by distance scales between pedestrians. We define the adjacency matrix $\tilde{A}^{[k]}$ according to the distance scale $k$ as

$$[\tilde{A}^{[k]}]_{i,j} = \begin{cases} 1 & \text{if } \Upsilon[k] \leq a_{i,j} < \Upsilon[k+1], \\ 1 & \text{if } i = j, \\ 0 & \text{otherwise}, \end{cases} \tag{4}$$

where, we set the scale to $\Upsilon \in \{0, 0.5, 1, 2, 4\}$.

Thus, $\tilde{A}^{[k]}$ can be expressed as a set of unweighted sub-graphs. Substituting the polynomial term $\tilde{A}^k$ with the disentangled adjacency matrix $\tilde{A}^{[k]}$, Equation (3) can be reformulated as:

$$H^{(l+1)} = \sigma\left(\sum_{k=1}^{K} \tilde{D}^{[k]-\frac{1}{2}} \tilde{A}^{[k]} \tilde{D}^{[k]-\frac{1}{2}} H^{(l)} W_k^{(l)}\right), \tag{5}$$

where, $\tilde{D}^{[k]}$ is the degree matrix of $\tilde{A}^{[k]}$.

As shown in the Figure 3(Bottom right), if the distance between pedestrians is within $\Upsilon[k] \leq a_{i,j} < \Upsilon[k+1]$, a strong connection is formed, even though they are located far away. With this representation, the disentangled multi-scale aggregation enables social interactions to be learned for both far and near pedestrians.

In addition, by disentangling neighbors according to distances, the over-smoothing problem is also alleviated. The aggregation across all edges obviously results in overwhelming amounts of irrelevant information, especially in crowded scenes. By contrast, disentangling a node on the sub-graph avoids the over-smoothing problem by adaptively aggregating information with respect to the scale ranges.

### 3.3 Spatio-Temporal Graph Aggregation

**Multi-relational GCN for pedestrian graph.** In this paper, one of our contributions is to apply the concept of a multi-relational graph to pedestrian trajectory prediction. Unlike the previous work (Mohamed et al. 2020) which constructed a graph while only considering the relative displacement between pedestrians, we use both Euclidean distance and their relative displacement. We observe that a model will suffer by avoiding either companions or persons walking behind back when it learns to only consider either the distance or the relative displacement.

Therefore, our model is coordinated to learn complementary features to combine the distance and the relative displacement information. We introduce a multi-relational graph with two types of relations $\mathcal{R} = \{(distance), (relative\_displacement)\}$. As a unified formula for the multi-relational graph and the disentangled multi-scale aggregation, we derive our model as below:

$$H^{(l+1)} = \sigma\left(\sum_{r=1}^{R}\sum_{k=1}^{K} \hat{A}_r^{[k]} H^{(l)} W_{r[k]}^{(l)}\right), \tag{6}$$

where $R$ is the number of relations, which is set to 2 in this work because we consider both the distance and the relative displacement edge. $\hat{A}_r^{[k]}$ is a normalized adjacency matrix as $\hat{A}_r^{[k]} = \tilde{D}_r^{[k]-\frac{1}{2}} \tilde{A}_r^{[k]} \tilde{D}_r^{[k]-\frac{1}{2}}$ with degree matrix $\tilde{D}_r^{[k]}$ of $\tilde{A}_r^{[k]}$. The scale set $\Upsilon$ for $\tilde{A}_r^{[k]}$ with respect to each relation is empirically determined and the detail will be described in Section 4.2.

**Spatio-temporal pedestrian graph.** We have described the aggregation method for the spatial part using a set of spatial graphs $\mathcal{G} = \{\mathcal{G}_t | t \in \mathbb{N}, 1 \leq t \leq T_{obs}\}$. As suggested in (Mohamed et al. 2020), we additionally make connections to define temporal edges between pedestrian nodes as $\mathcal{E}_n = \{e_{i,j} | i, j \in \mathbb{N}, 1 \leq i, j \leq T_{obs}, |i - j| \leq \lfloor \lambda/2 \rfloor\}$ for all $n \in \{1, ..., N\}$, where $\lambda$ is a user-defined parameter to control the number of neighbors. We implement it by passing them through one 2D convolution layer with $3 \times 1$ filter along a channel axis. We note that the temporal edge graphs are stacked over the channel axis of a feature map in order to represent time before the convolution operation.

### 3.4 DropEdge on Weighted Graph

Because there are limited datasets for pedestrian trajectory prediction, the state-of-the-art models often suffer from overfitting issue. As a solution to this problem, we utilize the DropEdge technique (Rong et al. 2020) which creates a sub-graph to randomly remove edges from an input graph in the training phase. Acting as a message passing reducer and data

| | ETH | HOTEL | UNIV | ZARA01 | ZARA02 | 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 | 1.09 / 2.35 | 0.79 / 1.76 | 0.67 / 1.40 | 0.47 / 1.00 | 0.56 / 1.17 | 0.72 / 1.54 |
| Social-GAN-P | 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 | 0.70 / 1.43 | 0.76 / 1.67 | 0.54 / 1.24 | 0.30 / 0.63 | 0.38 / 0.78 | 0.54 / 1.15 |
| PIF | 0.73 / 1.65 | 0.30 / 0.59 | 0.60 / 1.27 | 0.38 / 0.81 | 0.31 / 0.68 | 0.46 / 1.00 |
| Reciprocal Learning | 0.69 / 1.24 | 0.43 / 0.87 | 0.53 / 1.17 | **0.28** / 0.61 | 0.28 / 0.59 | 0.44 / 0.90 |
| STGAT | 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-BiGAT | 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 |
| **DMRGCN** | **0.60 / 1.09** | **0.21 / 0.30** | **0.35 / 0.63** | 0.29 / **0.47** | **0.25 / 0.41** | **0.34 / 0.58** |

Table 1: Comparison of our DMRGCN with other state-of-the-art methods (ADE/FDE). The results for the state-of-the-art methods are directly referred from (Sun, Zhao, and He 2020; Mohamed et al. 2020). Bold: Best, underline: Second best.

augmenter, DropEdge alleviates the overfitting problems of unweighted graph-based GCNs.

In order to apply DropEdge to our multi-relational GCN based on weighted graphs, we modify it by the element-wise multiplication of a binary matrix with an input adjacency matrix. To do this, we redefine the relational adjacency matrix $\tilde{A}_r$ which is a normalized term of $\hat{A}_r$ in Equation (2) as:

$$\tilde{A}_r^{(l)} = A_r \odot A_r'^{(l)} + I \quad \text{s.t.} \quad [A_r']_{i,j} = \begin{cases} 1 & \text{if } random(0,1) > p, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where $A_r'$ is the randomly-connected edge on the original node $\mathcal{V}$, $p$ is the dropping rate, and $\odot$ stands for element-wise multiplication. We will demonstrate the effectiveness of DropEdge for our model in Section 4.2.

### 3.5 Global Temporal Aggregation for Trajectory Prediction

**Time Prediction CNN.** Using the spatio-temporal feature followed by the multi-relational GCN in Section 3.3, we predict further trajectories with our designed time prediction CNN (TPCNN) module, instead of the RNN commonly used in early works (Alahi et al. 2016; Gupta et al. 2018). Our TPCNN module consists of two TCNs (Bai, Kolter, and Koltun 2018) as an inference module and one global temporal aggregation as a refinement module which will be described next. The TPCNN module predicts future time features $H_{T_{obs}+1:T_{pred}}$ to correctly extrapolate further trajectories by directly applying convolution operators to the spatio-temporal features $H_{1:T_{obs}}$ along with the time channel.

**Global Temporal Aggregation.** Most pedestrian trajectory prediction models have a common problem, in that prediction errors are accumulated as the sequences become longer. Particularly, this problem arises when a person turns around an obstacle or another person. In this work, we tackle this problem by proposing a novel global temporal aggregation (GTA) which learns to compensate for the accumulated error. GTA takes each pedestrian's trajectory as input, and outputs a single feature vector which is added to the initial prediction.

For time period $\mathcal{T} = T_{pred} - T_{obs}$, the GTA feature update rule is defined as:

$$h'_{t,n,c} = h_{t,n,c} + \sigma\left(\sum_{p=1}^{\mathcal{T}}\sum_{q=1}^{C}(h_{p,n,q} \times w_{p,q,c}) + b_{n,c}\right), \tag{8}$$

where $h$ is an element of the spatio-temporal feature $H$. $w$ and $b$ denote learnable kernel weights and biases, respectively. As illustrated in Figure 2(Blue box), the single feature vector is added to the hidden feature $h^n$ for each pedestrian to minimize the residual error between the initial prediction and the actual path.

### 3.6 Implementation Details

**Loss function.** Like previous works (Xu, Yang, and Du 2020; Mohamed et al. 2020; Shi et al. 2020), we use the bi-variate Gaussian probabilistic density function. Our model predicts the outputs of the position coordinates $\hat{p}_t^n \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_t^n, \hat{\boldsymbol{\sigma}}_t^n, \hat{\rho}_t^n)$ of a pedestrian $n$ at time $t$, where $\mathcal{N}(\cdot)$ is a multivariate normal distribution. $\hat{\boldsymbol{\mu}} = (\hat{\mu}_x, \hat{\mu}_y)$ are two mean variables in the $x, y$-axis motion, $\hat{\boldsymbol{\sigma}} = (\hat{\sigma}_x, \hat{\sigma}_y)$ are the corresponding standard deviations, and $\hat{\rho}$ is the correlation coefficient between the $x$ and $y$-axis motions. Using the predicted output, we minimize the loss function $\mathcal{L}$, as below:

$$\mathcal{L} = \sum_{n=1}^{N}\sum_{t=T_{obs}+1}^{T_{pred}} -\log\left[\frac{\exp(-0.5\Psi^T C_{\hat{p}_t^n}^{-1}\Psi)}{(2\pi)^{\frac{n}{2}}|C_{\hat{p}_t^n}|^{\frac{1}{2}}}\right]$$
$$\text{s.t. } \Psi = \begin{bmatrix} x_t^n - \hat{\mu}_{t,x}^n \\ y_t^n - \hat{\mu}_{t,y}^n \end{bmatrix}, C_{\hat{p}_t^n} = \begin{bmatrix} (\hat{\sigma}_{t,x}^n)^2 & \hat{\rho}\hat{\sigma}_{t,x}^n\hat{\sigma}_{t,y}^n \\ \hat{\rho}\hat{\sigma}_{t,x}^n\hat{\sigma}_{t,y}^n & (\hat{\sigma}_{t,y}^n)^2 \end{bmatrix}, \tag{9}$$

where $x_t^n$ and $y_t^n$ are ground-truth coordinates of the $x, y$-axis motion, respectively.

**Training procedure.** We construct a unified model with GCN followed by TPCNN for pedestrian trajectory prediction. We use one GCN and four TPCNN blocks, which shows the best results in ablation studies. Our model is trained for 256 epochs with the SGD optimizer. We use a mini-batch size of 128 with an initial learning rate $1e-4$ and decay of rate $0.8$ every 32 epochs. Data augmentation schemes such as random rotation, flip and scaling are utilized. The training is performed on a NVIDIA 2080Ti GPU, which usually takes 12 hours.

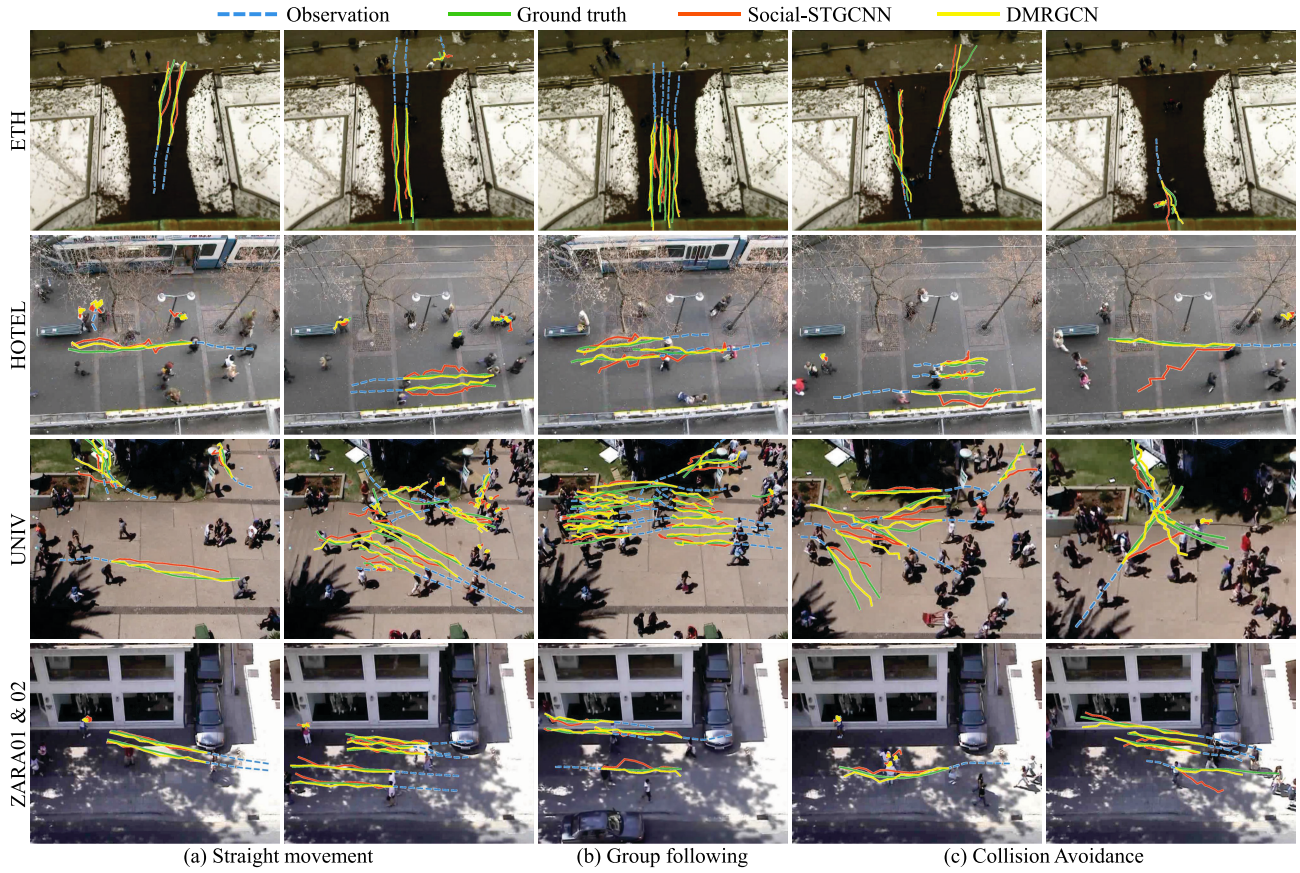


Figure 4: Examples of pedestrian trajectory prediction results. For the UNIV, the images are enlarged for better visualization. Our model is compared to Social-STGCNN, whose results are computed with a pre-trained network provided by the authors. To aid visualization, the trajectories with the best ADE are reported.

## 4 Experiments

In this section, we present our experimental results for two public datasets: ETH (Pellegrini et al. 2009) and UCY (Lerner, Chrysanthou, and Lischinski 2007) which contain pedestrian trajectories and various real-world human interactions. Both datasets include 5 subsets (ETH, HOTEL, UNIV, ZARA1 and ZARA2), and provide the locations of each pedestrian in 2D coordinates.

Following previous works (Alahi et al. 2016; Gupta et al. 2018; Kosaraju et al. 2019; Huang et al. 2019; Mohamed et al. 2020; Sun, Zhao, and He 2020), we adopt a leave-one-out evaluation strategy, in which four datasets are used for training and the remaining one is used for testing. Given pedestrian trajectories at 0.4 second time intervals, our model takes $T_{obs} = 8$ time steps (3.2 sec) as input, and predicts $T_{pred} - T_{obs} = 12$ time steps (4.8 sec).

In our evaluations, we use common quantitative measures to determine the accuracy of pedestrian trajectory, specifically, average displacement error (ADE) and final displacement error (FDE). All were reported in previous works (Alahi et al. 2016; Gupta et al. 2018; Mohamed et al. 2020; Sun, Zhao, and He 2020), and for quantitative evaluation we select the best predictions among 20 samples in $L_2$ norm, following the previous work (Gupta et al. 2018).

### 4.1 Comparison with State-of-the-art Methods

We compared our DMRGCN with these other state-of-the-art methods: Linear regression, Social-LSTM (Alahi et al. 2016), Social-GAN-P (Gupta et al. 2018), SoPhie (Sadeghian et al. 2019), PIF (Liang et al. 2019), Reciprocal Learning (Sun, Zhao, and He 2020), STGAT (Huang et al. 2019), and Social-BiGAT (Kosaraju et al. 2019), Social-STGCNN (Mohamed et al. 2020).

In Table 1, we report the results of the comparison between DMRGCN and the state-of-the-art works, using performance metrics ADE and FDE in meter scale. The results indicate DMRGCN provides the best performance on nearly all of the measures and datasets. Of particular note, DMRGCN shows huge performance improvements on UNIV and HOTEL, compared to other graph-based methods: STGAT, Social-BiGAT and Social-STGCNN. Since UNIV contains about 25 persons in a scene, they suffer from an over-smoothing problem. Our multi-relational graph representation effectively allows us to avoid this issue, while other graph-based methods can present complex social interactions on a single graph. Our DMRGCN also shows promising results on HOTEL which only contains a few people. We observe the disentangled multi-scale aggregation plays an important role in effectively detecting and avoiding

| Variant ID | Components | | | | | | | Performance |
|---|---|---|---|---|---|---|---|---|
| | DT | DP | AT | NG | NT | GT | DE | AVG |
| 1 | - | - | - | 0 | 4 | - | - | 0.39 / 0.74 |
| 2 | - | - | - | 0 | 4 | ✓ | - | 0.38 / 0.72 |
| 3 | ✓ | - | - | 1 | 4 | - | - | 0.38 / 0.71 |
| 4 | - | ✓ | - | 1 | 4 | - | - | 0.39 / 0.71 |
| 5 | ✓ | ✓ | - | 1 | 4 | - | - | 0.38 / 0.70 |
| 6 | ✓ | - | MS | 1 | 4 | - | - | 0.38 / 0.71 |
| 7 | - | ✓ | MS | 1 | 4 | - | - | 0.38 / 0.70 |
| 8 | ✓ | ✓ | MS | 1 | 4 | - | - | 0.38 / 0.69 |
| 9 | ✓ | - | DMS | 1 | 4 | - | - | 0.39 / 0.69 |
| 10 | - | ✓ | DMS | 1 | 4 | - | - | 0.38 / 0.69 |
| 11 | ✓ | ✓ | DMS | 1 | 4 | - | - | 0.37 / 0.68 |
| 12 | ✓ | ✓ | DMS | 1 | 4 | - | 0.9 | 0.37 / 0.66 |
| 13 | ✓ | ✓ | DMS | 1 | 4 | - | 0.8 | 0.35 / 0.63 |
| 14 | ✓ | ✓ | DMS | 1 | 4 | - | 0.7 | 0.37 / 0.66 |
| 15 | ✓ | ✓ | DMS | 1 | 4 | - | 0.6 | 0.37 / 0.69 |
| 16 | ✓ | ✓ | DMS | 1 | 4 | - | 0.5 | 0.37 / 0.67 |
| 17 | ✓ | ✓ | DMS | 1 | 1 | - | 0.8 | 0.38 / 0.69 |
| 18 | ✓ | ✓ | DMS | 1 | 2 | - | 0.8 | 0.38 / 0.70 |
| 19 | ✓ | ✓ | DMS | 1 | 3 | - | 0.8 | 0.37 / 0.68 |
| 20 | ✓ | ✓ | DMS | 1 | 5 | - | 0.8 | 0.38 / 0.70 |
| 21 | ✓ | ✓ | DMS | 1 | 6 | - | 0.8 | 0.39 / 0.73 |
| 22 | ✓ | ✓ | DMS | 1 | 4 | ✓ | 0.8 | **0.34 / 0.58** |
| 23 | ✓ | ✓ | DMS | 2 | 4 | ✓ | 0.8 | 0.36 / 0.61 |

Table 2: Ablation study. DT, DP, AT, NG, NT, GT and DE respectively denote distance, relative displacement, aggregation type (none, MS, DMS), the number of GCN, the number of TPCNN, global temporal aggregation and DropEdge. MS and DMS denotes multi-scale aggregation and disentangled multi-scale aggregation. Note that we have used the same scale factors for both MS and DMS, so that the same number of graphs are used.

collisions.

Figure 4 shows several cases where there are differences between the DMRGCN predictions and those of Social-STGCNN, which is the second best method in Table 1. Both models predict final destinations well in Figure 4(a) and (b). However, Social-STGCNN generates noisy paths because of the influence of standing persons, and incorrect relations for pedestrians walking behind the reference person. In addition, Social-STGCNN suffers from inaccurate path generation. As shown in Figure 4(c), Social-STGCNN does not recover the paths after avoiding an approaching group. On the other hand, by effectively incorporating each component, DMRGCN achieves state-of-the-art results by handling a set of difficult situations.

### 4.2 Ablation Studies

An extensive ablation study was conducted to examine the effects of different components on DMRGCN performance. We summarize the results in Table 2.

We first demonstrate the effectiveness of the multi-relational graph based on both distance and relative displacement between pedestrians (variants 3 to 5). We then confirm that the use of the disentangled multi-scale aggregation improves the performances with respect to the FDE metric, compared to multi-scale aggregation (variants 6 to 11). In addition, DropEdge provides better FDE results in our DMRGCN. Particularly, when the dropping rate in DropEdge is 0.8, the performances for both ADE and FDE get better (variants 12 to 16). Lastly, we examine the performance of DMRGCN with respect to the number of GCN and TCN (variants 17 to 23). Note that the performance improvement plateaus when one GCN and four TCN with GTA are used.

| Variant ID | Components | | | Performance |
|---|---|---|---|---|
| | DT | DP | Disentangling scale set | AVG |
| 1 | ✓ | | $\Upsilon$={0, 1/2, 1} | **0.38** / 0.72 |
| 2 | ✓ | | $\Upsilon$={0, 1, 4} | **0.38** / 0.71 |
| 3 | ✓ | | $\Upsilon$={0, 1/2, 1, 2, 4} | 0.39 / **0.69** |
| 4 | ✓ | | $\Upsilon$={0, 1, 2, 3, 4} | 0.39 / 0.72 |
| 5 | ✓ | | $\Upsilon$={0, 1/2, 1, 2, 4, 8, 16} | **0.38** / 0.70 |
| 6 | | ✓ | $\Upsilon$={0, 1/4, 1/2} | 0.39 / 0.73 |
| 7 | | ✓ | $\Upsilon$={0, 1/2, 1} | **0.38** / 0.71 |
| 8 | | ✓ | $\Upsilon$={0, 1/8, 1/4, 1/2, 1} | **0.38** / 0.70 |
| 9 | | ✓ | $\Upsilon$={0, 1/4, 1/2, 3/4, 1} | **0.38** / **0.69** |
| 10 | | ✓ | $\Upsilon$={0, 1/4, 1/2, 3/4, 1, 5/4, 3/2} | **0.38** / 0.70 |

Table 3: Ablation study on scale sets $\Upsilon$. DT and DP denote distance and relative displacement, respectively. The experiments are performed with one GCN and four TPCNNs.

Interestingly, the effectiveness of GTA is revealed with GCN. While the performance improvement by GTA is small in the absence of GCN (variants 1 and 2), it is relatively beneficial in the presence of GCN (variants 13 and 22). It can be seen that GTA corrects initial path predictions, especially long-term routes, based on the social interactions from the GCN-based aggregation.

As another ablation study, we apply a variety of scale sets $\Upsilon$ in Section 3.3. In Table 3, we test the performance changes for the distance adjacency matrix (variants 1 to 5), and the relative displacement matrix (variants 6 to 10). As expected, a denser sampling in areas closer to a reference person produces more accurate path predictions in general. This experiment also shows that longer connections (variants 5 and 10) have a negligible impact on accuracy compared with variants 3 and 9 which are used in our implementation.

## 5 Conclusion

In this paper, we have presented a novel DMRGCN architecture for pedestrian trajectory prediction. We have introduced a multi-relational graph to learn various types of social interaction, and have proposed a novel disentangled multi-scale aggregation to represent complex social interactions with a set of sub-graphs. In addition, through the incorporation of the temporal convolutional network and our novel global temporal aggregation, we are able to correct errors resulting from over-avoidance. Our DMRGCN outperforms the state-of-the-art methods with public datasets.

Directions exist for improving DMRGCN. One is to integrate prior state information similar to RNN models into TCN for smoother path predictions. Another is to impose prior knowledge on scene configuration and geometry. Lastly, DMRGCN assumes a camera is static. Lifting this restriction by designing dynamic graph models is an important future challenge for autonomous vehicle applications.

## Acknowledgements

This work is in part supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST)), Vehicles AI Convergence Research & Development Program through the National IT Industry Promotion Agency of Korea(NIPA) funded by the Ministry of Science and ICT(No. S1602-20-1001), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2014-3-00077, AI National Strategy Project), and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2020R1C1C1012635).

## References

Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; and Savarese, S. 2016. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bai, S.; Kolter, J. Z.; and Koltun, V. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* .

Bartoli, F.; Lisanti, G.; Ballan, L.; and Del Bimbo, A. 2018. Context-aware trajectory prediction. In *Proceedings of International Conference on Pattern Recognition (ICPR)*.

Fernando, T.; Denman, S.; Sridharan, S.; and Fookes, C. 2018. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural Networks* 108: 466–478.

Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; and Alahi, A. 2018. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Helbing, D.; and Molnar, P. 1995. Social force model for pedestrian dynamics. *Physical review E* 51(5): 4282.

Huang, Y.; Bi, H.; Li, Z.; Mao, T.; and Wang, Z. 2019. STGAT: Modeling Spatial-Temporal Interactions for Human Trajectory Prediction. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Kosaraju, V.; Sadeghian, A.; Martín-Martín, R.; Reid, I.; Rezatofighi, H.; and Savarese, S. 2019. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Lerner, A.; Chrysanthou, Y.; and Lischinski, D. 2007. Crowds by example. *Computer Graphics Forum* 26(3): 655–664.

Li, B.; Li, X.; Zhang, Z.; and Wu, F. 2019a. Spatio-temporal graph routing for skeleton-based action recognition. In *Thirty-Third AAAI Conference on Artificial Intelligence*.

Li, G.; Muller, M.; Thabet, A.; and Ghanem, B. 2019b. Deepgcns: Can gcns go as deep as cnns? In *Proceedings of International Conference on Computer Vision (ICCV)*.

Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; and Tian, Q. 2019c. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liang, J.; Jiang, L.; Murphy, K.; Yu, T.; and Hauptmann, A. 2020. The garden of forking paths: Towards multi-future trajectory prediction. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liang, J.; Jiang, L.; Niebles, J. C.; Hauptmann, A. G.; and Fei-Fei, L. 2019. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liu, Z.; Zhang, H.; Chen, Z.; Wang, Z.; and Ouyang, W. 2020. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Marcheggiani, D.; and Titov, I. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mehran, R.; Oyama, A.; and Shah, M. 2009. Abnormal crowd behavior detection using social force model. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mohamed, A.; Qian, K.; Elhoseiny, M.; and Claudel, C. 2020. 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)*.

Pellegrini, S.; Ess, A.; Schindler, K.; and van Gool, L. 2009. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Rehder, E.; and Kloeden, H. 2015. Goal-directed pedestrian prediction. In *Proceedings of International Conference on Computer Vision Workshops*.

Rehder, E.; Wirth, F.; Lauer, M.; and Stiller, C. 2018. Pedestrian prediction by planning using deep neural networks. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*.

Rong, Y.; Huang, W.; Xu, T.; and Huang, J. 2020. Dropedge: Towards deep graph convolutional networks on node classi-

fication. In *International Conference on Learning Representations (ICLR)*.

Sadeghian, A.; Kosaraju, V.; Sadeghian, A.; Hirose, N.; Rezatofighi, H.; and Savarese, S. 2019. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shi, L.; Zhang, Y.; Cheng, J.; and Lu, H. 2019a. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shi, L.; Zhang, Y.; Cheng, J.; and Lu, H. 2019b. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shi, X.; Shao, X.; Fan, Z.; Jiang, R.; Zhang, H.; Guo, Z.; Wu, G.; Yuan, W.; and Shibasaki, R. 2020. Multimodal interaction-aware trajectory prediction in crowded space. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Sun, H.; Zhao, Z.; and He, Z. 2020. Reciprocal learning networks for human trajectory prediction. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations (ICLR)*.

Vemula, A.; Muelling, K.; and Oh, J. 2018. Social attention: Modeling attention in human crowds. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations (ICLR)*.

Xu, Y.; Yang, J.; and Du, S. 2020. CF-LSTM: Cascaded feature-based long short-term networks for predicting pedestrian trajectory. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Yamaguchi, K.; Berg, A. C.; Ortiz, L. E.; and Berg, T. L. 2011. Who are you with and where are you going? In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yan, S.; Xiong, Y.; and Lin, D. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. *Thirty-Second AAAI Conference on Artificial Intelligence* .

Zhang, P.; Ouyang, W.; Zhang, P.; Xue, J.; and Zheng, N. 2019. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhu, J.-Y.; Zhang, R.; Pathak, D.; Darrell, T.; Efros, A. A.; Wang, O.; and Shechtman, E. 2017. Toward multimodal image-to-image translation. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.