P1-11
17th International Conference on Machine Vision Applications (MVA)
Fully Online, July 25–27, 2021.

Relational Subgraph for Graph-based Path Prediction

Masaki Miyata
Chubu University
Aichi
saboias21@mprg.cs.chubu.ac.jp

Katsutoshi Shiraki
Chubu University
Aichi
siraki@mprg.cs.chubu.ac.jp

Hiroaki Minoura
Chubu University
Aichi
himi1208@mprg.cs.chubu.ac.jp

Tsubasa Hirakawa
Chubu University
Aichi
hirakawa@mprg.cs.chubu.ac.jp

Takayoshi Yamashita
Chubu University
Aichi
takayoshi@isc.chubu.ac.jp

Hironobu Fujiyoshi
Chubu University
Aichi
fujiyoshi@isc.chubu.ac.jp

Abstract

Path prediction methods using graph convolutional networks (GCNs) that represent pedestrians' relationships by graphs have been proposed. These GCN-based methods consider only the distance information for the relationship between pedestrians, and the visibility state and other relationships are not taken into account. In this paper, we propose a path prediction method that represents the detailed relationship between pedestrians by introducing relational subgraphs. Each subgraph is constructed on different relationships. The proposed method inputs these relational subgraphs and the distance graph into GCNs and we extract features. Then, the features are input to a temporal convolutional network, which outputs multivariate Gaussian parameters to predict the future path. The experimental results with ETH and UCY datasets show that the proposed method outperforms the conventional method using only the distance information.

1 Introduction

Path prediction, which predicts the future paths of moving objects such as pedestrians and vehicles, is a fundamental problem. In particular, predicting pedestrians' paths plays an important role in various applications such as autonomous driving and surveillance systems. For example, by predicting pedestrians' paths in autonomous driving and robotics application, vehicles can be controlled to avoid collision with them appropriately. In surveillance systems, the path of pedestrians can be predicted to identify unusual activities such as collisions. Consequently, path prediction is expected to be used in various fields and has been widely addressed [3, 4, 5, 7, 9]. For accurate path prediction, it is important to consider various auxiliary information such as the pedestrian's direction and posture, the behavior of other pedestrians, and the surrounding static environment. In particular, understanding the behavior of other pedestrians, i.e., considering the interactions among pedestrians is expected to realize path prediction that avoids collisions among pedestrians.

For these purposes, a path prediction method that considers the interactions between pedestrians in a graph convolutional network (GCN) has been proposed [7, 9]. In these methods, the edges in the graph structure are weighted with the distance information between pedestrians. The future path is predicted from the weighted edges and the nodes with each pedestrian's past positions. However, these methods consider only the distance information between the objects as interactions. Human relationships are not only distances and are also complex and diverse, such as destinations and groups. Therefore, considering interactions by only distance ignores the diversity and complexity of relationships.

In this paper, we propose a graph-based path prediction method that introduces the *relational subgraphs*. The relational subgraph represents several relationships between pedestrians, such as the direction of pedestrian movement and the distance between objects. We construct these graph structures and input them into GCNs to extract features. The extracted features are input to a temporal convolutional network (TCN) [2], which outputs the multivariate Gaussian parameters to predict future paths. The proposed relational subgraph enables path prediction to take into account diverse and complex relationships. In our evaluation experiment, we demonstrate the effect of introducing diverse and complex relationships.

2 Related work

Path prediction for pedestrians has been widely investigated in the last few years due to deep learning [3, 4, 5, 6, 7, 9].

The standard and primal approach is based on a long short-term memory (LSTM) [3, 4]. LSTM-based approach encodes the past pedestrian's path, i.e., coordinates, by LSTM. Since an LSTM network encodes only a single pedestrian path, these methods consider the relationship between pedestrians by spatially encoding features of the target and the other pedestrians [3] or directly pooling these features [4].

Apart from the above approaches, our method is based on graph representation and GCNs. The graph-based path prediction methods have been proposed [7, 9]. These methods represent the relationship between pedestrians as graph representation and consider the interactions. These methods define a fully connected spatio-temporal graph structure weighted by relative distance and predict using GCN [7, 9]. However, it is difficult for these methods to sufficiently represent the relationships because they utilize only distance information. Meanwhile, we propose using a subgraph in which only some nodes are connected to consider other relations.

Since the future path is unknown, there exist several possibilities for selecting a pedestrian path. For considering such diversity of future paths, the path prediction approach probabilistically predicts the future paths by sampling future paths from randomized latent variables have been proposed [4, 5]. We also predict future paths as probabilistic distributions. Our method estimates the multivariate Gaussian distribution parameters from our proposed network and predicts future paths by sampling path based on the estimated distribution.

3 Method

In this section, we introduce the proposed path prediction method. Our method introduces the relational subgraphs to consider the state and relationship of pedestrians and the distance graph used by conventional methods.

Figure 1 shows the network structure of the proposed method. The proposed method consists of two modules. Given N pedestrians, we construct two graphs: the distance graph and the relational subgraph. We create these graphs for each observation time step $t \in \{1, \dots, T_O\}$ and input them for the following modules. The first module is GCN, which extracts graph features from input graphs. In this module, we separately extract features from two input graphs. Then, these features are aggregated into one by point-wise convolution. The second module is a TCN that takes the extracted graph features and outputs multivariate Gaussian distribution parameters. Based on the obtained parameters, we predicts the future paths by sampling two-dimensional coordinates $\boldsymbol{p}_t^n = (x_t^n, y_t^n)$, where $t \in 1, \dots, T_P$ during prediction time step. Hereafter, we describe the details of the proposed method.

3.1 Distance graph

Let $G_t = (V_t, E_t)$ be a graph that represents the position and relationship between pedestrians, where $V_t = \{v_t | \forall i \in \{1, \dots, N\}\}$ and $E_t = \{e^{i,j} | \forall i, j \in \{1, \dots, N\}\}$ are nodes and edges, respectively. We create a graph for each time step t, and the node contains each pedestrian's location information at t. Edges represent the relationship between pedestrians and the strength of mutual influences. The distance graph used in existing works [7, 9], represents the influences by using distance. Figure 2(a) shows an example of the distance graph. When the nodes are connected, we use the inverse of the L2 distance for $e^{i,j}$. If they are disconnected, $e^{i,j} = 0$.

3.2 Relational Subgraph

The relationship and influence do not derive only from a distance. The other factor would affect the future path. In this paper, we introduce the relational subgraph to represent the detailed relationship, and we propose the following relational subgraphs. We show the examples of the relational subgraphs in Figs. 2(b-f).

Short subgraph connects edges whose distance is lower than the threshold ($1.2m$), which consider only near neighbors and suppress the influence from the distant pedestrian. The speed subgraph connects edges if the difference of movement in one frame is lower than the threshold ($0.5m$). The direction subgraph represents whether pedestrians move in the same direction or not. The edge is connected if the cosine similarity is larger than the threshold (0.5). The group subgraph represents the group behavior of several pedestrians. We define the group subgraph by the element product of three subgraphs: short, speed, and direction. The visibility status subgraph indicates whether the target is included in the viewing angle. We set the viewing angle of a person as 200 degrees. We connect edges of the pedestrians in the viewing range are targeted.

3.3 Graph Convolutional Network

Given the direction graph and the relational subgraph, we extract graph features by GCNs. The node feature v^i is computed by aggregating from itself v^i and

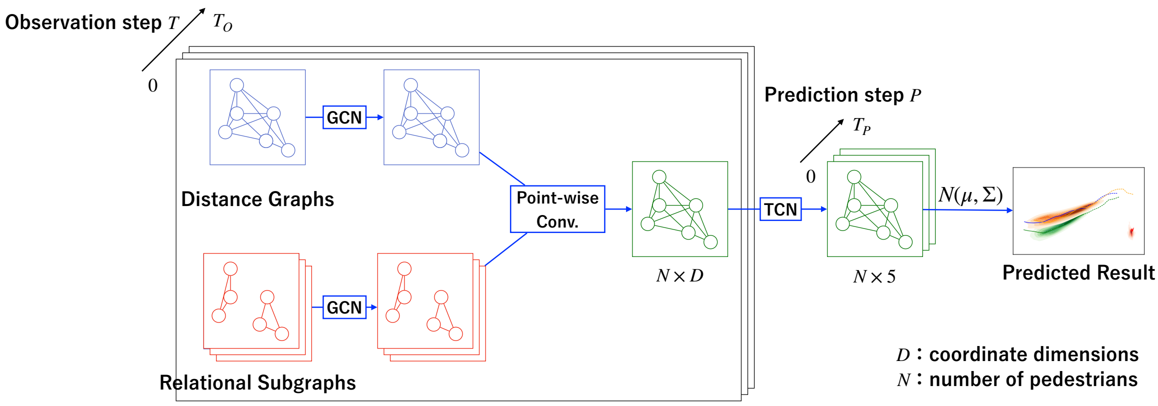

Figure 1. Network structure of the proposed method.

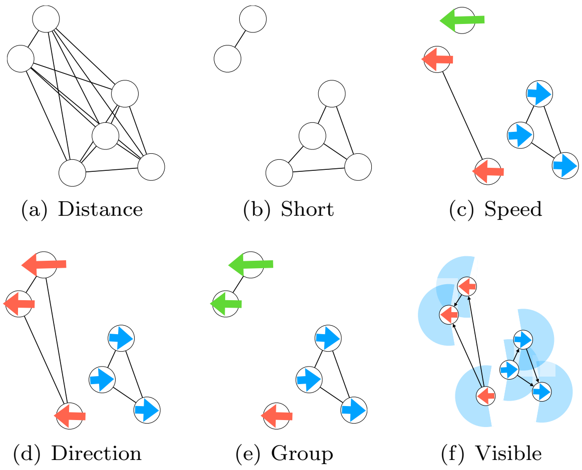

Figure 2. The structures of (a) distance graph and (b-f) relational subgraphs.

the neighbors $B(v^i)$ as follows:

$$\hat{v}^i = \sigma\left(\frac{1}{\Omega}\sum_{v^j \in B(v^i)} p\left(v^i, v^j\right) \cdot \mathbf{w}\left(v^i, v^j\right)\right), \tag{1}$$

where, σ is an activation function, $p(\cdot)$ is a sampling function for aggregating neighbouring node features, and $\mathbf{w}$ is trainable parameters. Note that $\frac{1}{\Omega}$ denotes the normalization term.

The extracted features from the distance graph and the relational subgraphs are aggregated as a single graph feature by point-wise convolution. We compute features for each time step separately.

3.4 Temporal Convolutional Network

We then predict the future path by the extracted features and TCN. We concatenate the extracted features in the time step direction and apply temporal convolution. The output of TCN is the parameters of a multivariate Gaussian distribution: mean μ_t^n, variance σ_t^n, and correlation ρ_t^n for each prediction step t and each pedestrian n. The future coordinates $\mathbf{p}_t^n$ is sampled by using the distribution by

$$\mathbf{p}_t^n \sim \mathcal{N}(\mu_t^n, \sigma_t^n, \rho_t^n). \tag{2}$$

To train the network, we compute a negative log-likelihood loss as follows:

$$L^n\left(\mathbf{W}\right) = -\sum_{t=1}^{T_P} \log\left(\mathbb{P}\left(\mathbf{p}_t^n | \mu_t^n, \sigma_t^n, \rho_t^n\right)\right), \tag{3}$$

where $\mathbf{W}$ is every trainable parameters in the network.

4 Experiments

In this experiment, we examine the effectiveness of the proposed method and each relational subgraph.

4.1 Experimental Settings

We use ETH [10] and UCY [11] datasets for our evaluation. Through our experiments, the observation time is 3.2 seconds ($T_O = 8$), and the prediction time is 4.8 seconds ($T_P = 12$). The GCN consists of a single convolutional layer, and the TCN has three convolutional layers. The kernel sizes of every convolutional layer in the TCN are $3 \times 3 \times T_P$. For network training, we used a stochastic gradient descent optimization whose learning rate is 0.01. The batch size is 128, and the number of epochs is 250.

We compare the performance with Social-STGCNN [9], which uses only the distance graph as an input. As we mentioned in the previous section, we use the following relational subgraphs as the proposed method: short, speed, direction, group, and visible state (visible). We compare these prediction performances. As evaluation metrics, we use the average displacement

Table 1. ADE and FDE metrics on ETH/UCY datasets

Graphs	eth	hotel	univ	zara1	zara2	average
	ADE/FDE	ADE/FDE	ADE/FDE	ADE/FDE	ADE/FDE	ADE/FDE
Distance only [9]	0.73/1.23	0.33/0.41	0.49/**0.82**	0.34/0.57	**0.30**/0.50	0.44/0.71
Short	0.72/1.20	0.35/0.41	**0.46/0.85**	**0.33**/0.56	0.32/0.52	0.44/0.71
Speed	0.68/1.21	0.24/0.34	0.57/0.95	0.40/0.67	0.35/0.54	0.45/0.75
Direction	0.73/1.10	0.23/0.31	0.50/0.90	0.37/0.58	0.32/0.50	0.43/0.68
Group	**0.63/0.98**	**0.20/0.27**	0.51/0.86	0.35/0.57	0.34/0.50	0.41/0.64
Visible	0.64/0.99	0.22/0.29	0.49/0.86	0.34/**0.51**	**0.30/0.48**	**0.40/0.63**

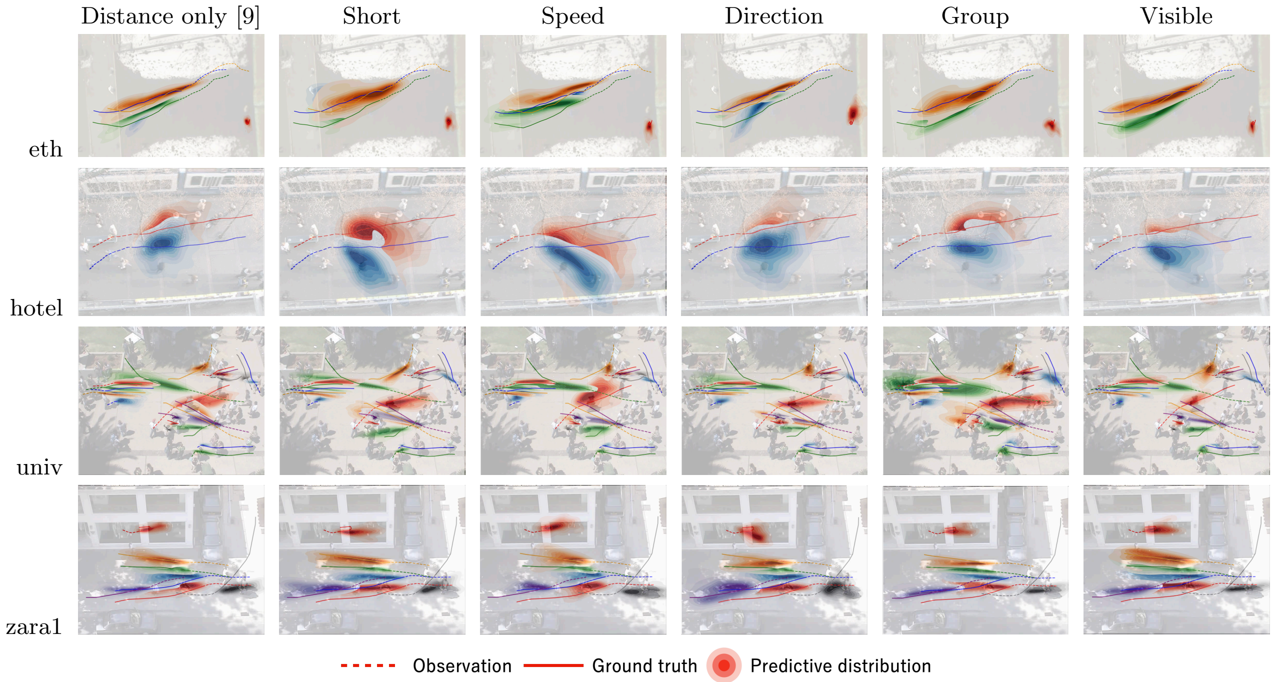

Figure 3. Visualization results of predicted paths on ETH and UCY datasets.

error (ADE) and the final displacement error (FDE). ADE is the average of the Euclidean distance between the predicted and true value at each prediction time. FDE is the Euclidean distance between the predicted and true value at the final prediction time.

4.2 Results

First, we show the quantitative results with ADE and FDE in Tab. 1. Except for univ, our method with group or visible achieved higher performances than the results of distance only [9]. These subgraphs successfully restrict the influence from unrelated pedestrian to decide own path.

In the results of univ, the short subgraph achieved a lower error. The univ data is a crowded scene, and there are many pedestrians. In such a case, restricting the distant pedestrians' connection suppresses unnecessary influence and could improve accuracy.

In the cases of speed and direction subgraphs, there were no improvements. Moreover, the short subgraph does not improve the accuracy except for univ. The reason for the inaccurate results is that the graph structure using multiple simple relations becomes complex, which hinders the improvement of accuracy.

4.3 Visualization Results

We show the visualization results of predicted paths in Fig. 3.

The top row in Fig. 3 shows the results of eth, where two pedestrians (orange and blue) move together. In the distance only [9], the blue pedestrian tries to avoid collision with the orange pedestrian, resulting in blue one contact with the green pedestrian. Also, the results of short, speed, direction subgraphs failed to predict correct paths. Meanwhile, the group subgraph assumes blue and orange pedestrians as the same group, and we can successfully predict their paths.

As we discussed above, univ is a crowded scene and one of the most challenging scenes in ETH/UCY datasets. The third row in Fig. 3 shows the results of univ. The predicted paths of the short subgraph are rather converged compared with another result.

In the bottom row in Fig. 3, pedestrians move in the same direction. The group subgraph assumes these pedestrians as the same group. As a result, orange pedestrian follows neighbor green, blue, and red pedestrians and failed to predict path. The visible subgraph can predict the path because we can ignore the effects from behind.

5 Conclusion

In this paper, we proposed a path prediction method using relational subgraphs to represent richer social interactions. Each subgraph represents a specific relationship between pedestrians and the surrounding objects, such as the direction of pedestrian movement and visibility state. The relational subgraph enables us to predict a more accurate future path. In the experimental results with ETH and UCY datasets, the proposed method outperform the conventional methods, and we showed the effectiveness of the relational subgraphs. Our future work includes automatic generation of relational subgraphs and improvement of the network.

References

[1] Kipf, Thomas N. and Welling, Max, “Semi-Supervised Classification with Graph Convolutional Networks,” International Conference on Learning Representations, 2017

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

[3] Alexandre Alahi and Kratarth Goel and Vignesh Ramanathan and Alexandre Robicquet and Li Fei-Fei and Silvio Savarese, “Social LSTM: Human Trajectory Prediction in Crowded Spaces,” Computer Vision and Pattern Recognition, pp. 961-971, 2016

[4] Agrim Gupta and Justin Johnson and Li Fei-Fei and Silvio Savarese and Alexandre Alahi, “Social gan: Socially acceptable trajectories with generative adversarial networks,” Computer Vision and Pattern Recognition, June 2018

[5] Sadeghian, Amir and Kosaraju, Vineet and Sadeghian, Ali and Hirose, Noriaki and Rezatofighi, Hamid and Savarese, Silvio, “Sophie: An attentive gan for predicting paths compliant to social and physical constraints,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1349–1358 2019.

[6] Liang, Junwei and Jiang, Lu and Niebles, Juan Carlos and Hauptmann, Alexander G and Fei-Fei, Li, “Peeking into the future: Predicting future person activities and locations in videos,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5725–5734, 2019

[7] Anirudh Vemula and Katharina Muelling and Jean Oh, “Social attention : Modeling attention in human crowds,” IEEE International Conference on Robotics and Automation, pp. 1–7, 2018

[8] Yu Bing and Yin Haoteng and Zhu Zhanxing, “Spatio-temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting,” Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2018

[9] Mohamed, Abduallah and Qian, Kun and Elhoseiny, Mohamed and Claudel, Christian, “Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 14424–14432, 2020

[10] Pellegrini, Stefano and Ess, Andreas and Schindler, Konrad and Van Gool, Luc, “You’ll never walk alone: Modeling social behavior for multi-target tracking,” IEEE International Conference on Computer Vision, pp. 261–268, 2009

[11] Lerner, Alon and Chrysanthou, Yiorgos and Lischinski, Dani, “Crowds by example,” Computer graphics forum, vol. 26, No. 3, pp. 655–664, 2007