Representing Multimodal Behaviors with Mean Location for Pedestrian Trajectory Prediction

Liushuai Shi, Le Wang, *Senior Member, IEEE,* Chengjiang Long, *Member, IEEE,* Sanping Zhou, *Member, IEEE,* Wei Tang, *Member, IEEE,* Nanning Zheng, *Fellow, IEEE,* and Gang Hua, *Fellow, IEEE*

Abstract—Representing multimodal behaviors is a critical challenge for pedestrian trajectory prediction. Previous methods commonly represent this multimodality with multiple latent variables repeatedly sampled from a latent space, encountering difficulties in interpretable trajectory prediction. Moreover, the latent space is usually built by encoding global interaction into future trajectory, which inevitably introduces superfluous interactions and thus leads to performance reduction. To tackle these issues, we propose a novel Interpretable Multimodality Predictor (IMP) for pedestrian trajectory prediction, whose core is to represent a specific mode by its mean location. We model the distribution of mean location as a Gaussian Mixture Model (GMM) conditioned on sparse spatio-temporal features, and sample multiple mean locations from the decoupled components of GMM to encourage multimodality. Our IMP brings four-fold benefits: 1) Interpretable prediction to provide semantics about the motion behavior of a specific mode; 2) Friendly visualization to present multimodal behaviors; 3) Well theoretical feasibility to estimate the distribution of mean locations supported by the central-limit theorem; 4) Effective sparse spatio-temporal features to reduce superfluous interactions and model temporal continuity of interaction. Extensive experiments validate that our IMP not only outperforms state-of-the-art methods but also can achieve a controllable prediction by customizing the corresponding mean location.

Index Terms—Pedestrian Trajectory Prediction, Multimodal Trajectory Prediction, Central-limit Theorem.

1 Introduction

Given the observed trajectories of a pedestrian and its neighbors, pedestrian trajectory prediction is to predict a sequence of the future locations of the pedestrian. This task plays a critical role in various vision applications, such as autonomous vehicles [1], [2], surveillance systems [3], [4], and other motion prediction tasks [5], [6].

One key challenge of pedestrian trajectory prediction is inherent multimodality incurred by the multiple possibilities of future behavior. In other words, given an observed trajectory, there are multimodal behaviors represented by diverse future trajectories [7], [8] that a pedestrian could take. For example, a pedestrian may go straight, turn left/right, or keep still. This motivates the community to address the multimodal prediction task.

Previous methods commonly embed multimodal behaviors into a latent space by the Conditional Variational Autoencoder (CVAE) framework [9], [10] conditioned on individual temporal dependencies and complex spatial interactions. Especially, global interaction [7], [10], [11] is usually employed to model spatial interaction from all neighbors of a pedestrian at each time step. After that, a latent variable is sampled from the latent space to represent a specific mode. Hence, multiple latent variables sampled repeatedly from the latent space can represent multimodal behaviors, and further achieving multimodal prediction. The upper branch of Figure 1 illustrates this process. Despite the advances of these latent methods, they still suffer from the following two limitations.

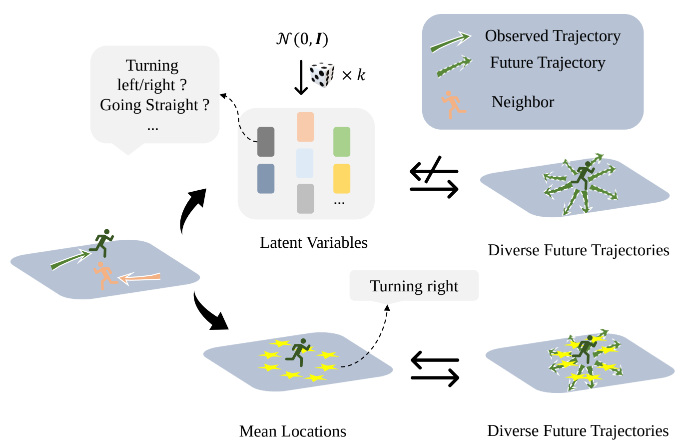


Fig. 1. Contrastive illustration between previous latent-based methods (upper branch) and our proposed method (lower branch). Latent-based methods present multimodal behaviors by multiple latent variables sampled from a prior distribution, while ours presents multimodal behaviors via the mean locations of the full trajectories.

First, representing multimodality by an inscrutable latent space lacks interpretability. This causes two disadvantages in practice. On the one hand, we cannot understand the distribution of multimodal motion behaviors based on the inscrutable latent space. On the other hand, it is hard to obtain a controllable prediction because it is unknown how the latent variable, randomly sampled from the latent space, encodes the multimodal behaviors of a pedestrian.

- *Liushuai Shi, Le Wang, Sanping Zhou, and Nanning Zheng are with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China. E-mail: shiliushuai@stu.xjtu.edu.cn, {lewang,spzhou,nnzheng}@mail.xjtu.edu.cn. (Corresponding author: Le Wang.)*
- *Chengjiang Long is with Meta Reality Labs (formerly Facebook Reality Labs), Burlingame, CA 94010, USA. E-mail: cjfykx@gmail.com.*
- *Wei Tang is with the Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA. E-mail: tangw@uic.edu.*
- *Gang Hua is with Wormpex AI Research, Bellevue, WA 98004, USA. E-mail: ganghua@gmail.com.*

For instance, a robot needs to understand how a pedestrian will turn left to avoid an accident, but it is impossible to sample trajectories *specific to* this mode from the uninterpretable latent space. To handle these problems, prior methods need to repeatedly sample trajectories of multimodal behaviors to fill the distribution of multimodal future trajectories and search for trajectories of the desired mode. However, it is unstable and unfriendly for tasks that testing time is short, such as accident avoidance.

Second, prior methods usually model the global interaction [7], [9], [11], [12] at each time step, assuming that a pedestrian interacts with all the neighbors due to the efficient computation. As the upper branch of Figure 2 illustrates, the pedestrian interacts with all the neighbors at each time step. However, as shown in the lower branch of Figure 2, a pedestrian hardly interacts with all others spatially. Hence, the global interaction can introduce superfluous interactions that disturb the trajectory prediction. In addition, the global interaction at each step is time-independent, and thus it is not suitable to model the temporal continuous interaction.

To address the above issues, we attempt to explore a simple yet effective representation of a pedestrian's mode. We identify two necessary criteria. 1) The representation should connect to the physical world. That is, humans could understand the pedestrian's behavior, such as turning left/right or going straight, given the representation. This can further enhance the interpretability of multimodal future trajectories and contribute to achieving a controllable prediction. 2) The representation should account for the spatio-temporal relationships between pedestrians. In other words, the representation should capture the temporal dependence between the observed trajectory and multimodal future behaviors manifested by diverse future trajectories. In addition, the representation also should model the complex spatial interaction between a pedestrian and its neighbors, which contributes to the predicted trajectories abide by the social traffic rules, such as avoiding traffic collisions. Driven by these analyses, we propose a novel Interpretable Multimodality Predictor (IMP) for pedestrian trajectory prediction. Our IMP jointly employs an interpretable intention representation and a social interaction representation to represent the trajectory of each pedestrian.

Concretely, the interpretable intention representation models the future behavior mode in the physical world by the mean location of a full trajectory. The full trajectory includes both the observed trajectory and the corresponding future trajectory. Meanwhile, we extract sparse spatio-temporal features as the social interaction representation in the feature space to model the spatio-temporal relationships between pedestrians. According to the central-limit theorem, we model the distribution of mean locations via an explicit Gaussian Mixture Model (GMM) conditioned on sparse spatio-temporal features. Because only one future trajectory (ground truth) is observed for one pedestrian during training, we predict diverse future trajectories greedily by a teacher-forcing strategy. Specifically, the current full trajectory is converted into its mean location to represent the current mode. Then, the mean location of current mode is directly encoded and then concatenated with the sparse spatio-temporal features to predict the single future trajectory in the training phase. In the inference phase, we sample diverse mean locations from separate components of the GMM to predict diverse future trajectories. Sampling in this way can ensure that the model treats each mode fairly and thus improve the diversity of predicted trajectories to cover multimodal behaviors.

The social interaction representation in the feature space is regarded as the extraction of sparse spatio-temporal features.

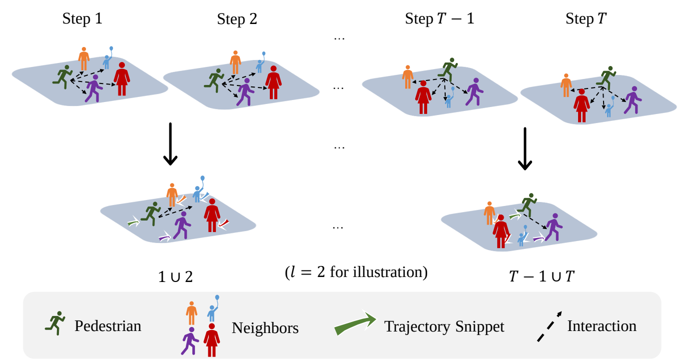


Fig. 2. Comparison between global interaction (upper branch) and our proposed sparse interaction (lower branch). Global interaction assumes a pedestrian interacts with all neighbors at each time step, while our proposed sparse interaction assumes a pedestrian adaptively interacts with partial neighbors at each trajectory snippet.

Specifically, a snippet-level embedding divides the observed trajectory with length T into multiple non-overlapping trajectory snippets with length l and then extracts the embedding of each trajectory snippet. As shown in Figure 2, the trajectory snippet integrates multiple continuous trajectory points temporally. Thus, it can model the temporal continuity of interaction by unifying similar interactions at multiple time steps into a single interaction in a trajectory snippet. Subsequently, a sparse spatial interaction is built to drop superfluous neighbors at each trajectory snippet. For instance, the pedestrian interacts with its partial neighbors at each trajectory snippet, as illustrated in the lower branch of Figure 2. Moreover, we capture the snippet-level temporal dependencies among the snippets, reducing the computation complexity from $\mathcal{O}(T^2)$ to $\mathcal{O}(T^2/l^2)$ and maintaining the prediction accuracy.

Four-fold benefits are brought by our IMP in pedestrian trajectory prediction: 1) Interpretable multimodal motion behaviors. The learned GMM connects the multimodal motion behaviors to the physical world instead of an inscrutable space. Thus, the mean location can provide semantic information (motion behavior) of a specific mode. As illustrated in Figure 1, the mean location marked by the right yellow star could indicate the pedestrian will turn right. Furthermore, predicting the future trajectory of the turning right mode via the mean location provides the rationale behind the prediction. 2) Friendly visualization. The mean location can be visualized to reflect the distribution of multimodal behaviors in the 2D coordinate, without the post-processing of trajectory prediction. Thus, it can accelerate intelligent systems such as autopilot to understand the pedestrian's multimodal behaviors. 3) Well theoretic feasibility. The mean location follows a normal distribution approximately according to the central-limit theorem, supporting the feasibility of estimating the distribution of the mean locations. 4) Effective spatio-temporal feature extraction. The sparse spatio-temporal features could reduce the superfluous interactions and model the temporal continuity of interaction. It contributes to achieving a better performance as shown in Table 4.

Extensive experiments on ETH [13], UCY [14], Stanford Drones Dataset (SDD) [15], nuScenes [16], and Argoverse [17] show that our IMP outperforms the state-of-the-art methods. Besides, the ablation study and visualization results validate the effectiveness of the proposed interpretable intention representation and social interaction representation. What's more, our method is able to achieve a controllable prediction by customizing the corresponding mean location, which is very critical to understanding how the pedestrian moves in some emergency situations.

In summary, the contributions of this paper are summarized below.

- We propose a novel interpretable multimodality predictor for pedestrian trajectory prediction, with advantages in interpretable motion behaviors, friendly visualization, theoretical feasibility, and controllable prediction.
- We propose to extract the sparse spatio-temporal features to reduce the superfluous interactions and model the temporal continuity of interaction. It is also beneficial to reduce the time complexity and maintain/improve the accuracy in temporal dependence capturing.
- Extensive experiments on two benchmarks demonstrate the efficacy of our proposed method against existing state-of-the-art methods.

This paper extends our previous conference paper [18], and the new major contributions include:

- A simple yet effective interpretable intention representation, *i.e.*, mean location, is proposed to represent multimodal behaviors, thus enabling the prediction of diverse future trajectories.
- A snippet-level embedding is added to extend the sparse interaction proposed in the conference paper from each time step to each trajectory snippet, thus modeling the temporal continuity of interaction and reducing the time complexity in capturing temporal dependence.
- More technical details about the proposed method are presented.
- More experiments (including comparisons on the Stanford Drone Dataset and more ablation studies) are carried out to evaluate the effectiveness of proposed method.

The rest of the paper is organized as follows. Section 2 briefly reviews related work in pedestrian trajectory prediction. Subsequently, we present the technical details of the proposed method in Section 3. Experimental results are presented in Section 4. Finally, we conclude this paper in Section 5.

2 Related Work

We briefly review related work in spatio-temporal feature extraction, multimodal trajectory prediction, and self-attention for pedestrian trajectory prediction.

2.1 Spatio-Temporal Feature Extraction

Prior works capture temporal dependence on the observed individual trajectory and model the spatial interaction by integrating neighbors' motion to obtain spatio-temporal features. Many works [7], [19], [20] employ the Recurrent Neural Networks (RNN) [21] or its variants such as LSTM [22] and GRU [23] to capture the temporal dependence of trajectory. Correspondingly, the local [19] and global [7] pooling mechanisms are leveraged to model spatial interaction. The local one integrates hidden states of neighbors within a certain radius, while the global one integrates hidden states of all neighbors involved in a scene. Due to the inefficiency of recurrent architectures, the temporal convolutional networks (TCNs) [12], [24] and the self-attention mechanism [11], [25] are employed to capture temporal dependence in an efficient parallel computation manner.

Since the graph structure can better describe the trajectory scene, another track of works models the spatial interaction using the graph. Social-BiGAT [8] employs the Graph Attention Network (GAT) [26] on the hidden representation of pedestrians to model spatial interaction. To better represent the interaction between pedestrians, Social-STGCNN [12] directly models the trajectory as a graph, where the edges weighted by the pedestrian relative distance represent interactions between pedestrians. EvolveGraph [27] builds a dynamic interaction graph to represent multiple possible interaction types by its edge. A multi-class edge classification task is conducted to recognize the interaction types of two pedestrians. Specially, "no edge" is one interaction type that implies no interaction between pedestrians.

Sun *et al.* [28] indicate there are strong interactions between some distant pedestrian pairs, hence inviting sociologists to manually divide the pedestrians into different groups according to specific physical rules and sociological actions. Motivated by the success of Transformer [25], some works [11], [29] employ the Transformer architecture to extract spatio-temporal features. In addition, several works [30], [31], [32], [33], [34] leverage the visual features of the scene to improve the spatio-temporal features. This paper aims to represent multimodal behaviors and predict diverse future trajectories without using visual features, like most works.

Prior methods model the global interaction [7], [9], [11], [12] at each time step, thus inevitably introducing superfluous spatial interactions from non-interactive neighbors. Moreover, global interaction is time-independent, and thus cannot model the temporal continuity of interaction. In contrast, our sparse sptio-temporal features build a sparse interaction at each snippet to reduce superfluous interactions and model the temporal continuity of interaction. In addition, capturing the temporal dependence among snippets has lower computation complexity than prior methods among time steps.

2.2 Multimodal Trajectory Prediction

Given the observed trajectory of a pedestrian, there are multiple reasonable future trajectories that the pedestrian could take. Hence, pedestrian trajectory prediction is inherently a multimodal trajectory prediction task [7], [30]. Many works [10], [29], [30], [35] encode the future trajectories into a CVAE-based latent space and then sample multiple latent variables to represent multimodal behaviors. Then, the multimodal future trajectories are predicted by decoding these sampled latent variables. Specifically, PECNet [10] treats multimodal behaviors as multimodal future destinations and encodes the destinations into a standard Gaussian distribution based on CVAE. SGAN [7] and SoPhie [32] replace CVAE with a generative adversarial network (GAN) for multimodal trajectory prediction. STAR [11] adds random noise sampled from a prior distribution onto the learned spatio-temporal features to obtain the multimodal future trajectories. In contrast, our IMP represents a specific mode by the mean location of the full trajectory. It has shown its advantages in interpretable mutlimodal motion behaviors, friendly visualization, theoretical feasibility, and controllable prediction.

TNT [36] and DenseTNT [37] regard multimodal behaviors as multimodal future destinations, and they focus on vehicle trajectory prediction (VTP). Unlike pedestrian trajectory prediction, VTP

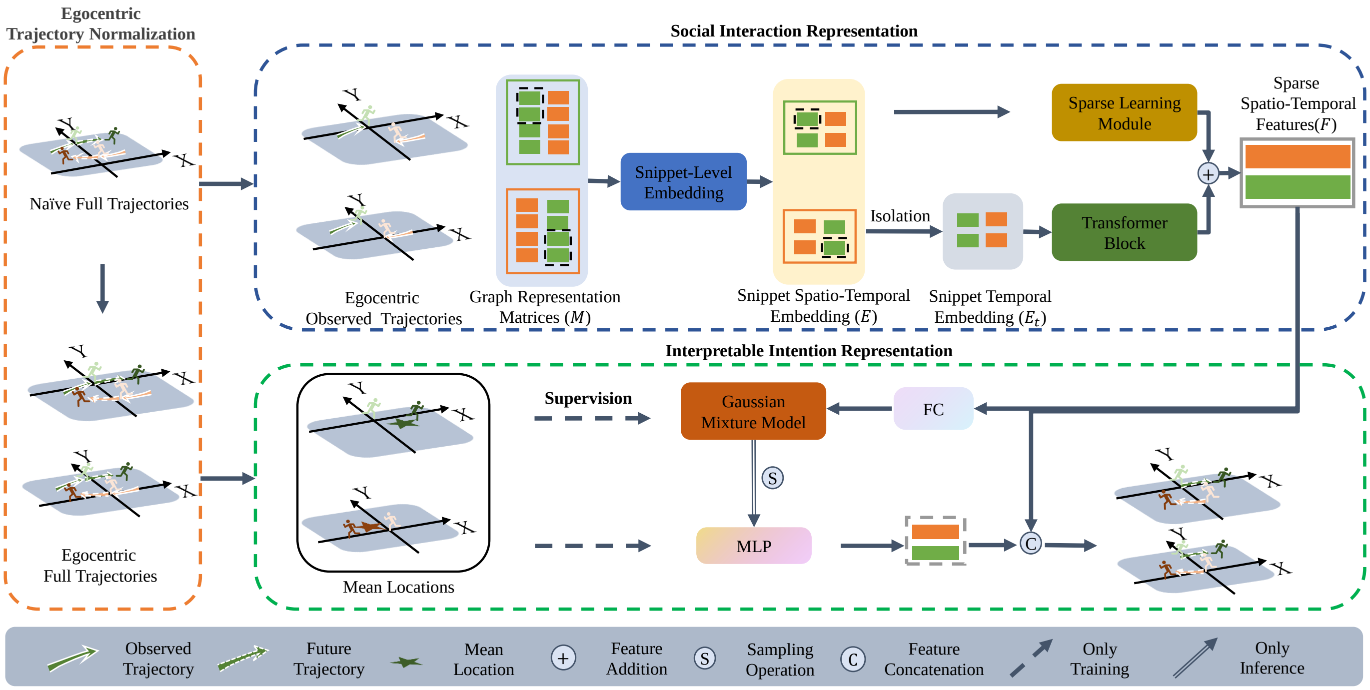


Fig. 3. The framework of our proposed IMP. The naive full trajectories are first normalized by trajectory translation and then fed into the next two parallel branches. The upper branch models the social interaction representation, where the egocentric observed trajectories are first represented by the graph representation matrices and then the snippet-level embedding module embeds the snippet of graph representation matrices to obtain the snippet spatio-temporal embedding E and the isolated snippet temporal embedding E_t. Next, E and E_t are fed into the sparse learning module to extract sparse interaction features F_s. E_t is fed into a standard Transformer block to capture snippet-level temporal dependence features F_t. Fusing F_s and F_t, the sparse spatio-temporal features F are generated as the social interaction representation to represent observed information. The lower branch models the proposed interpretable intention representation. It converts the egocentric full trajectory to its mean location with a specific mode. Supervised by the mean location, a Gaussian Mixture Model (GMM) is estimated based on F to obtain the distribution of the mean locations. Meanwhile, we encode the mean location and then concatenate with F to predict the future trajectory. In inference, we sample multiple mean locations from the GMM to predict diverse future trajectories, which can cover multimodal behaviors.

could use the HD map, which contains multiple helpful traffic elements (*e.g.*, lane and traffic sign), to restrict the movement of traffic agents. Thus, they directly sample abundant destinations according to the lanes. Then, regression and scoring are performed on the sampled destinations to optimize and filter the trajectories. However, the only provided information in pedestrian trajectory prediction is the trajectory, resulting in weak physical constraints for moving pedestrians. Thus, pedestrians have much larger moving flexibility compared with vehicles. It is challenging to model the distribution of multimodal motion behaviors in the physical world without the physical constraints in such a flexible scene. In contrast, the mean location is an average of trajectory, which can model the multimodal motion behaviors into a Gaussian distribution in the physical world. Also, the destination/middle point is an exact position, *i.e.*, the last or middle point. It requires the model to sample the destination/point with high accuracy, leading to greater difficulty in sampling. In contrast, the mean location as an average of trajectory smooths the future state, being a coarse position. The coarse position provides a higher error-tolerant rate than the exact position in sampling.

2.3 Central-Limit Theorem

The central-limit theorem (CLT) is an essential concept in statistics. It proves that the summation or mean of independent random variables tends to follow a normal distribution, even the original random variables are not normally distributed.

Classic CLT is built on independent and identically distributed (i.i.d.) random variables, while many works [38], [39], [40], [41] have evaluated the effectiveness of CLT on dependent random variables. For example, the CLT also works on a mixing sequence, meaning the data-generating process is asymptotically independent. Namely, the random variables temporally far apart from one another are nearly independent. As for the trajectory similar to a time series sequence, it is natural that two distant trajectory points tend to be independent, such as the beginning point and the destination. Moreover, the experimental results show that the mean location modeled as the Gaussian distribution works well. Thus, we believe that the CLT for a mixing sequence is worthy of further study in pedestrian trajectory prediction to provide statistical interpretability.

2.4 Self-attention

The core idea of Transformer [25], *i.e.*, self-attention, has successfully exhibited its advantage over RNNs [22], [23] on a series of sequence modeling tasks in natural language processing, such as text generation [42] and machine translation [43]. Self-attention decouples the attention into the query, key, and value, which can capture long-range dependencies and take advantage of parallel computation compared with RNNs. To describe the relationship between every pair of elements in the input sequence, self-attention computes attention scores by a matrix multiplication between the query and key.

To reduce the computational complexity of Transformer, sparse Transformer [44] is proposed to reduce the length of the sequence by dropping the elements at a longer distance. Unlike them, our proposed sparse interaction employs a sparse attention mechanism, which can reduce superfluous interactions in a learnable style instead of manually setting the distance.

3 Proposed Method

3.1 Problem Definition

Pedestrian trajectory prediction aims to predict future location coordinates of pedestrians based on the observed trajectory. We follow existing methods [11], [45] that assume the spatial trajectory coordinates (2D-Cartesian) of pedestrian are preprocessed by the tracking algorithm at each time step. The trajectory coordinate of pedestrian n at the time step t is denoted by (x_t^n, y_t^n). We observe a trajectory from time step 1 to T, and predict the next trajectory from time step $T+1$ to $T+q$. Note that the model is required to predict diverse future trajectories to cover multimodal behaviors, while only a single real future trajectory (ground truth) is provided in the dataset to train the model.

3.2 Method Overview

We introduce our proposed Interpretabl Multimodality Predictor (IMP) for pedestrian trajectory prediction, which consists of one trajectory preprocessing and two parallel branches, as illustrated in Figure 3. Firstly, the trajectory is normalized to reduce trajectory variance and improve trajectory prediction. Here, we use the egocentric trajectory normalization [46], [47] commonly used in vehicle trajectory prediction to obtain the egocentric trajectory, which is fed into the next two parallel branches. The upper branch illustrates the process of building the social interaction representation. Since spatial interaction is continuous in temporal, a snippet-level embedding module proceeds to embed the snippet of egocentric observed trajectories represented by the graph representation matrices (M) and produces snippet spatio-temporal embedding E on the egocentric trajectory representation. Subsequently, a sparse learning module models the sparse interaction features F_s on E to alleviate superfluous interactions. Meanwhile, isolated from the embedding of neighbors in E, the snippet temporal embedding E_t is fed into a standard Transformer block to capture the snippet-level temporal dependence features F_t. Afterward, a feature fusion operates on F_s and F_t to obtain the sparse spatio-temporal features F.

The lower branch models the interpretable intention representation conditioned on the sparse spatio-temporal feature F and predicts diverse future trajectories to cover multimodal behaviors. As illustrated in Figure 3, it first converts the egocentric full trajectory of a specific mode into its mean location. After that, we cluster the mean locations via a Gaussian Mixture Model (GMM), which are estimated via a fully connected layer (FC) on F supervised by the mean location. Subsequently, we predict the multimodal future trajectories greedily with the teacher-forcing strategy [48] due to the given single future trajectory (ground truth). Specifically, the mean location in the current scene is directly encoded and then concatenated with F to predict the single future trajectory in the training phase. While in the inference phase, multiple mean locations are sampled from the separated components of the GMM to predict diverse future trajectories, and thus cover multimodal behaviors.

3.3 Trajectory Normalization

Trajectory normalization [10], [11] is capable of reducing trajectory variance and improving prediction performance. Here, we employ the egocentric trajectory normalization [46], [47], [49], [50] commonly used in vehicle trajectory prediction to normalize the input trajectory.

Given the naive full trajectory $X \in \mathbb{R}^{N\times(T+q)\times D}$ of N pedestrians in the scene, where T denoted the length of observed trajectory, q is the length of future trajectory and D denotes the dimension of trajectory coordinate, we center the T trajectory point of X for each pedestrian in the coordinate system to obtain the end-observed-centered trajectory $\bar{X} \in \mathbb{R}^{N\times(T+q)\times D}$, which is generated by a trajectory subtraction operation. Specifically, the trajectory points of pedestrian n at time step $t \in \{1, ..., T+q\}$ subtract the trajectory point at the time step T as follows:

$$\bar{X}_n^t = X_n^t - X_n^T, \tag{1}$$

where X_n^t and X_n^T are the trajectory points at time steps t and T, respectively. $\bar{X}_n^t$ is the end-observed-centered trajectory point of pedestrian n at time step t. The end-observed-centered trajectory $\bar{X}_n \in \mathbb{R}^{(T+q)\times D}$ of pedestrian n is generated by stacking $\{\bar{X}_n^t\}_{t=1}^{T+q}$. Hence, $\bar{X}$ is generated by stacking $\{\bar{X}_n\}_{n=1}^{N}$.

As the translation destroys the relative positions between a pedestrian and its neighbors, we calculate the relative displacement between a pedestrian and its neighbors to store the relative positions. From the view of the pedestrian n, the relative displacement $\Delta_{n|j}^t$ between pedestrian n and its neighbor j at time step t is calculated by a trajectory subtraction as below:

$$\Delta_{n|j}^t = X_n^t - X_j^t, \tag{2}$$

where X_n^t and X_j^t are trajectory points of pedestrian n and neighbor j, respectively, at the time step t. The relative displacement $\Delta_{n|j} \in \mathbb{R}^{(T+q)\times D}$ between pedestrian n and its neighbor j can be obtained by stacking $\{\Delta_{n|j}^t\}_{t=1}^{T+q}$. Accordingly, the relative displacement $\Delta_n \in \mathbb{R}^{N\times(T+q)\times D}$ between pedestrian n and its N neighbors is generated by stacking $\{\Delta_{n|j}\}_{j=1}^{N}$. Note that the pedestrian self belongs to one of its neighbors for computational convenience. Hence, the relative displacement $\Delta \in \mathbb{R}^{N\times N\times(T+q)\times D}$ for each pedestrian is gained by stacking $\{\Delta_n\}_{n=1}^{N}$.

After that, we use a trajectory addition operation between the relative displacement and end-observed-centered trajectory to restore the relative positions. To be specific, the end-observed-centered trajectory $\bar{X}_n$ of pedestrian n adds the relative displacement $\Delta_{n|j}$ to restore the relative position of neighbor j as:

$$\hat{X}_{n|j} = \bar{X}_n + \Delta_{n|j}, \tag{3}$$

where $\hat{X}_{n|j} \in \mathbb{R}^{(T+q)\times D}$ is the egocentric full trajectory of neighbor j refer to pedestrian n. Accordingly, the egocentric full trajectory $\hat{X}_n \in \mathbb{R}^{N\times(T+q)\times D}$ of pedestrian n is obtained by stacking $\{\hat{X}_{n|j}\}_{j=1}^{N}$.

By stacking $\{\hat{X}_n\}_{n=1}^{N}$, the egocentric full trajectory $\mathbf{X} \in \mathbb{R}^{N\times N\times(T+q)\times D}$ is generated to represent the trajectory scene. The egocentric observed trajectory $\mathbf{X}_{\text{obs}} \in \mathbb{R}^{N\times N\times T\times D}$ is produced by deleting the future part of $\mathbf{X}$. Note that the whole computation including the stacking operation can be processed parallel to reduce time consumption.

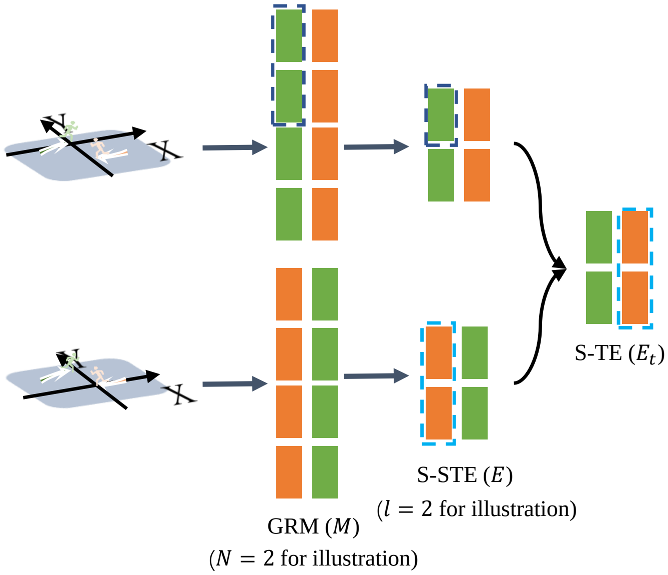


Fig. 4. Illustration of snippet-level embedding. It embeds the non-overlapped snippets on the egocentric observed trajectory to model the temporal continuity of interaction.

3.4 Snippet-level Embedding

Since the interaction is a continuous process, we employ a snippet-level embedding module on egocentric observed trajectory $\mathbf{X}_{\text{obs}}$. It unifies the similar interactions over multiple continuous time steps into a single interaction at a snippet and produces the snippet spatio-temporal embedding E.

As illustrated in Figure 4, we present the value of $\mathbf{X}_{\text{obs}}$ with the graph representation matrices (GRM) $\{M_n\}_{n=1}^{N} \in \mathbb{R}^{N\times T\times D}$, in which each element is a D-dimension trajectory coordinate, the row represents the trajectory coordinate of each neighbor at a time step and the column represents the trajectory coordinate of a neighbor at each time step. Based on the assumption that the direction of a trajectory will not change too abruptly, we divide an observed trajectory sequence with length T into multiple non-overlapped trajectory snippets with length l marked by the blue dotted rectangular in M_n. To embed the snippet, a 1D convolution kernel o with size k, stride s, and zero padding is operated on the column of each M_n to obtain the snippet-level spatio-temporal embedding E_n for pedestrian n, where $k = s = l$ due to the non-overlapped snippets. An example with $l = 2$ and $N = 2$ is illustrated in Figure 4. The process is described as

$$E_n = M_n \otimes o + b_m, \tag{4}$$

where $M_n \in \mathbb{R}^{N\times T\times D}$, $E_n \in \mathbb{R}^{N\times L\times D_e}$, and $L = T/l$. $\otimes$ is the convolutional operation with the learnable kernel o. b_m is a learnable bias following o. By stacking $\{E_1, .., E_N\}$, we obtain the final snippet spatio-temporal embedding (S-STE) $E \in \mathbb{R}^{N\times N\times L\times D}$.

Afterward, we isolate the embedding of neighbors on E by stacking the first column of $\{E_n\}_{n=1}^{N}$ to obtain the snippet-level temporal embedding (S-TE) $E_t \in \mathbb{R}^{N\times 1\times L\times D_e}$, as shown in Figure 4. E and E_t are fed into the next sparse learning module to model sparse spatial interaction. Thanks to the snippet-level embedding, the sparse spatial interaction could model the temporal continuity of interaction. Meanwhile, E_t is used to capture the snippet-level temporal dependence by a standard Transformer [25] block. The snippet-level embedding also can reduce the computation complexity of capturing temporal dependence from $\mathcal{O}(T^2)$ to $\mathcal{O}((T/l)^2)$ and maintain or even improve prediction performance, as discussed in Section 4.2.2.

3.5 Sparse Spatio-temporal Feature Extraction

We extract the sparse spatio-temporal features F to model our social interaction representation by building sparse spatial interaction features F_s and capturing snippet-level temporal dependence features F_t. Concretely, F_s is modeled by our sparse learning module with sparse cross attention, while F_t is captured by the Transformer [25] block with standard self-attention. Finally, F is obtained by a feature fusion between F_s and F_t.

3.5.1 *Sparse Learning Module*

The snippet-level embedding encapsulates the continuous interaction into a snippet and thus can model the temporal continuity of interaction. Here, our sparse learning module (SLM) aims to reduce superfluous spatial interactions generated from the non-interactive neighbors at each snippet. It inputs the snippet-level spatio-temporal embedding E and temporal embedding E_t, and outputs the corresponding sparse spatial interaction F_s. The core design of SLM is like a dictionary lookup. Considering the pedestrian as query and its neighbors as keys, the goal of SLM is to find the interactive keys and drop the superfluous keys out. This relationship between the query and its keys is represented by a sparse attention matrix, which is generated by the designed sparse attention learning block, as illustrated in Figure 5. In the sparse attention matrix, the superfluous keys are quantified to zero, while the interactive ones are quantified to interaction weights.

To build the dictionary lookup, we first employ a linear transformation on E and E_t to obtain the keys and query, respectively. Then, both the keys and query are decomposed into H subspaces by splitting the feature dimension into H equal parts. In subspace $h \in \{1, ..., H\}$, the cross-attention mechanism is used to compute the global attention $A_h \in \mathbb{R}^{N\times L\times 1\times N}$. By stacking $\{A_h\}_{h=1}^{H}$, the multi-head global attention $A \in \mathbb{R}^{N\times L\times H\times N}$ is obtained to represent the feature similarities between the query and keys, measured by the dot-product of the pair-wise query and key. The process is as follows:

$$\begin{aligned} Q &= \text{splitting}(\phi(E_t, W^Q)), \\ K &= \text{splitting}(\phi(E, W^K)), \\ A &= \text{Softmax}(\frac{QK^T}{\sqrt{d}}), \end{aligned} \tag{5}$$

where $\phi(\cdot,\cdot)$ denotes linear transformation. $W_Q \in \mathbb{R}^{D_e\times D_Q}$ and $W_K \in \mathbb{R}^{D_e\times D_K}$ are weights of the linear transformation. $Q = \{Q_h\}_{h=1}^{H}$ and $K = \{K_h\}_{h=1}^{H}$ are the query and key in each subspace, respectively. $\sqrt{d} = \sqrt{D_Q}$ is a scaled factor [25] in ensuring numerical stability. $\{A_h\}_{h=1}^{H} = \text{Softmax}(\{Q_h K_h^T\}_{h=1}^{H}/\sqrt{d})$.

Since A represents the attention between a query and its all keys, the superfluous attention from the superfluous keys could disturb the trajectory prediction. Thus, a sparse attention learning block is designed to learn a sparse attention matrix, as illustrated at the right of Figure 5. It first receives the multi-head global attention A to measure whether there is an interaction or not by considering various feature similarities comprehensively in H subspaces. Namely, assume $\mathbf{a} = \{a_h\}_{h=1}^{H}$ represents the feature similarities between a query and a specific key in H subspaces.

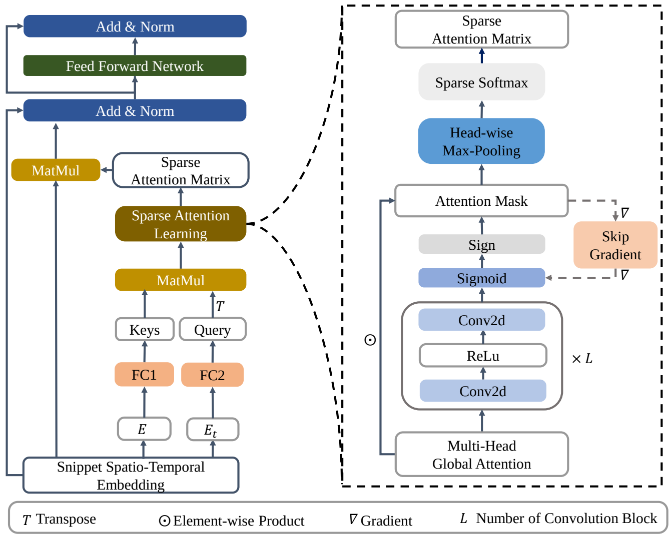


Fig. 5. Illustration of our proposed sparse learning module. The multi-head global attention is first learned to represent the feature similarity. The subsequent sparse attention learning is used to generate the sparse attention matrix, which drops the non-interactive neighbors out and quantifies the interactive.

A set of parameters $\mathbf{w} = \{w_h\}_{h=1}^{H}$ are learned to judge whether the query interacts with the key. To achieve this, we consider the head (subspace) dimension of A as the channel and cascade L non-linear convolution blocks with 1×1 kernel on A to obtain the fused feature similarities F_A, as shown in Figure 5. After that, a logistic regression, *i.e.*, sigmoid function, is used to generate an attention mask as follows:

$$\begin{aligned} F_A &= A \otimes \mathcal{K} + b_a, \\ R &= \mathbb{I}\left\{\sigma\left(F_A\right) \geq 0.5\right\}, \end{aligned} \tag{6}$$

where $\otimes$ is a convolutional operation. $\mathcal{K}$ is the learnable 1×1 convolution kernel. b_a is a learnable bias following $\mathcal{K}$. $\mathbb{I}\{\cdot\}$ is the indicator function, which outputs 1 if the corresponding inequality holds, otherwise 0. σ is the sigmoid function. R is the attention mask represented by 0/1 matrices. Since a pedestrian could interact with a specific neighbor in a subspace but no interaction in another subspace, the last convolution block outputs the feature with 1 channel, *i.e.*, $F_A \in \mathbb{R}^{N \times L \times 1 \times N}$ and $R \in \mathbb{R}^{N \times L \times 1 \times N}$, to avoid inconsistent interactions in various subspaces.

Since personal motion strongly influences the future trajectory prediction, we assume each pedestrian is self-interactive. Namely, we amend R to $\hat{R}$ by assigning the elements in the main diagonal of R to 1. Accordingly, an element-wise matrix multiplication between $\hat{R}$ and A is used to generate a sparse attention $\bar{A} \in \mathbb{R}^{N \times L \times H \times N}$ by broadcasting the shape of $\hat{R}$ same as A.

Similar to the above consistent interactions in various subspaces of R, a max-pooling mechanism along the head dimension is operated on $\bar{A}$ to build the consistent sparse attention $\widetilde{A} \in \mathbb{R}^{N \times L \times 1 \times N}$, whose benefit is evaluated empirically in Section 4.2.3. The interactive keys and superfluous keys are quantified to zero and non-zero values in $\widetilde{A}$, respectively. Subsequently, we need to normalize the values of interactive keys to obtain the quantitative weights. Unfortunately, the naive self-attention [25] uses the "Softmax" on the attention score to obtain probability representation, which leads to a side-effect: the sparse matrix will be back to a dense matrix again because Softmax outputs non-zero values for zero inputs. It leads to the sparse interaction will be back to a global interaction again. Hence, a sparse-softmax function is employed to prevent this problem as follows:

$$\hat{a}_i = \frac{\exp(\widetilde{a}_i) \cdot r_i}{\sum_j^{\mathcal{D}} \exp(\widetilde{a}_j) \cdot r_j + \epsilon}, \tag{7}$$

where $\widetilde{a}_i$ is the i-th element at the normalized dimension of $\widetilde{A}$. r_i is the i-th element at the normalized dimension of $\hat{R}$. $\cdot$ is an element-wise multiplication operation. $\mathcal{D}$ is the number of normalized dimension of $\widetilde{A}$ and $\hat{R}$. ϵ is a neglectable small constant to ensure numerical stability. $\hat{a}_i$ is the i-th element of the final sparse attention matrix $\hat{A}$.

After above processes, the final sparse attention matrix $\hat{A}$ can represent the results of the above dictionary lookup. Each row of $\hat{A}$ is regarded as the interaction between a pedestrian and its neighbors in a specific snippet, where the zero value is a mask to drop superfluous key out, while the non-zero value is an interactive weight to quantify the interaction.

Afterward, we extract the sparse interaction feature on the sparse attention matrix $\hat{A}$. Based on the Transformer [25] architecture as illustrated in Figure 5, we make a weighted summation among the embedding of interactive keys in E into E_t. The subsequent feed-forward network incorporated with residual connection is used to increase non-linearity and extract deep interactive features as follows:

$$\begin{aligned} \hat{F}_s &= \mathrm{LN}(\hat{A}E\mathbf{W_s} + E_t), \\ F_s &= \mathrm{LN}(\mathrm{FFN}(\hat{F}_s) + \hat{F}_s), \end{aligned} \tag{8}$$

where $\mathbf{W_s} \in \mathbb{R}^{D_e \times D_g}$ is the learnable parameter matrix. LN is the layer normalization. FFN is the feed-forward network implemented by two fully-connected layers with a RELU activation function. F_s is the final interactive feature of our proposed sparse spatial interaction.

Due to the non-differentiable sign function, *i.e.*, Eq. (4), we customize the backpropagation by a skip gradient [51] strategy illustrated in Figure 5 to ensure the whole network can be trained end-to-end. Specifically, we skip the process to compute the gradient of the sigh function and propagate the gradient from the back layer to the front layer, *i.e.*, from the operation of element-wise multiplication to the sigmoid function.

3.5.2 Temporal Dependence Learning

The future trajectory is not only influenced by the spatial interaction from its neighbors but also by the temporal dependence on the observed personal trajectory. Focusing on the temporal dependence, we input the snippet temporal embedding E_t into a standard Transformer block to capture the snippet-level temporal dependence features F_t. Note that E_t is first added to the position encoding [25] due to the sequential trajectory in temporal dependence.

After obtaining the sparse spatial interaction features F_s and the snippet-level temporal dependence features F_t, we fuse them to gain the sparse spatio-temporal features F by a feature addition, which is regared as the social interaction representation.

3.6 Interpretable Intention Representation

Our IMP builds the interpretable intention representation of multimodal behaviors conditioned on the extracted sparse spatio-temporal features F. It contains two steps: representing multimodal behaviors by mean location and clustering mean location via a

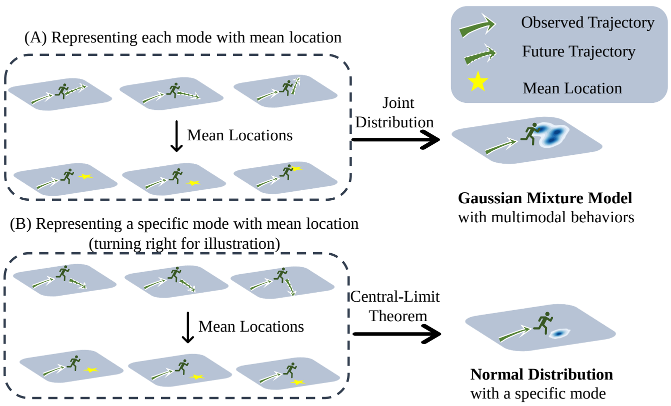


Fig. 6. Illustration of mean location. The observed trajectory and the future trajectory are concatenated and then converted into its mean location to represent multimodal behaviors. According to the central-limit theorem, the mean locations of a specific mode follow a normal distribution. A GMM is used to model multiple modes jointly.

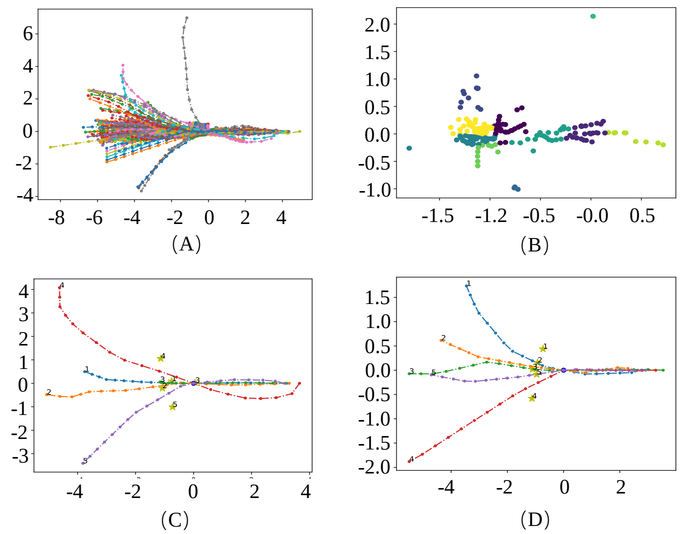


Fig. 7. Visualization of the mean location from training data. (A) is the sampled full trajectories with similar going straight observed trajectory. (B) is the clustered distribution of the mean locations from (A) by the Expectation-Maximization (EM) algorithm. (C) and (D) are the full trajectories and its mean location sampled from (A).

GMM according to the central-limit theorem. The detailed process is described immediately below.

Representation by Mean Location. Due to the the inherent multimodality about the pedestrian's motion, the pedestrian performs multimodal behaviors in future, such as turning left/right, or going straight, given the similar observed trajectory as illustrated in Figure 6 (A). We represent these multimodal behaviors by their mean location of the full trajectory, *i.e.* the green observed trajectory concatenated with the red dotted future trajectory.

We use the training data in ETH [13] and UCY [14] to give an intuitive evaluation about the mean location representing motion behavior. As shown in Figure 7, (A) shows the selected full trajectories with similar observed trajectories, *i.e.*, going straight. Note that the full trajectories are first shifted to the origin and then rotated to align the positive direction to the negative X-axis in the 2D cartesian coordinate system. (A) indicates the future behaviors are multimodal, such as going straight, turning left and right, conditioned on similar observed trajectories. We sample full trajectories from (A) to illustrate that the mean location (yellow star) follows the motion tendency of the future trajectory as shown in (C) and (D). (B) shows the clustered mixture distribution of mean locations calculated from the full trajectories in (A) by the Expectation-Maximization (EM) algorithm, where different colors represent different motion behaviors.

Thus, the mean location generated from the full trajectory can provide semantic information (motion behavior) to interpret future behavior. Predicting the future trajectory of the a specific mode, such as turning left, via the right mean location provides the rationale behind the prediction. Furthermore, the mean location achieves a controllable prediction. Due to the 2D coordinate, we can customize the mean location in coordinate system to represent desired mode. Such as, setting the mean location right the pedestrian and then predicting corresponding future trajectory could understand how the pedestrian turns right in future as illustrated in Figure 8.

Mean Location Distribution. Since the mean location comes from the full trajectory, we cannot acquire it directly in inference time. Thus, a Gaussian Mixture Model (GMM) is estimated on the extracted sparse spatio-temporal features F to represent the distribution of mean location supported by the central-limit theorem.

For a specific mode, the pedestrian performs various motion behaviors due to the randomness of the pedestrian's motion, such as turning right at various angles or distances, as illustrated in Figure 6 (B). Thus, there are multiple full trajectories $\{\mathbf{Y}_i\}_{i=1}^{m} = \{\mathbf{y}_i^t\}_{t=1}^{T+q}$ with the specific mode given the similar observed trajectory, where m is the number of full trajectories. Then, we convert these full trajectories into its mean locations as

$$\bar{\mathbf{y}}_i = \frac{\sum_{t=1}^{T+q} \mathbf{y}_i^t}{T+q}, \tag{9}$$

where $T+q$ is the length of the full trajectory, and $\bar{\mathbf{y}}_i$ is the mean location of the full trajectory $\mathbf{Y}_i$.

Once that, $\bar{\mathbf{y}}_i$ has well theoretical feasibility to estimate its distribution according to the central-limit theorem described in Theorem 1.

Theorem 1 (Central-limit theorem). Let $\{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_n\}$ be random samples drawn from a population with an overall mean $\boldsymbol{\mu}$ and finite variance $\boldsymbol{\sigma}^2$. If $\bar{\mathbf{X}}_n$ is the sample mean of n samples, the distribution of $\bar{\mathbf{X}}_n$ approximately obeys a normal distribution with the mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma}^2/n$.

The classic central-limit theorem (CLT) is built on independent and identically distributed (i.i.d.) random variables, while the trajectory is a time series sequence owing to dependence. Despite that, many works [38], [39], [40], [41] have verified that the CLT still works for the dependent sequence in practice. For example, the mixing sequence similar to the trajectory is asymptotically independent in data generation. Namely, the data points in sequence temporally far apart from one another are nearly independent. Naturally, two distant trajectory points are independent, such as the beginning point and the destination. We refer to a lecture[1] which introduces details about CLT on a mixing sequences. In this way, the CLT built on the i.i.d. cases will still be applicable.

Let us assume that a specific mode is a mixing sequence consisting of $m(T+q)$ samples of some continuous two-dimensional

1. https://www.stat.cmu.edu/~cshalizi/754/2006/notes/lecture-27.pdf

variables $\{\mathbf{y}_i^t | t \in \{1, ..., T+q\}, i \in \{1, ..., m\}\}$, namely the trajectory points of the full trajectory. These samples follow a conditional distribution $p(\mathbf{y}|\boldsymbol{c})$ with the expectation $\boldsymbol{u}$ and the variance $\boldsymbol{\Sigma}$, where $\mathbf{c}$ is the prior information, *i.e.*, observed trajectory. According to the CLT, the mean location of the samples follows the normal distribution $\mathcal{N}(\boldsymbol{u}, \bar{\boldsymbol{\Sigma}})$ approximately, where $\bar{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}/[m(T+q)]$. Due to the multiple behaviors, *e.g.*, turning left/right and going straight, there are multiple normal distributions $\{\mathcal{N}(\boldsymbol{\mu}_i, \bar{\boldsymbol{\Sigma}}_i)\}_{i=1}^{K}$ to model the mean location of multimodal behaviors, where K is the number of modes. Hence, a GMM $q(\overline{\mathbf{y}}|c)$ is used to model the distribution of the multimodal behaviors jointly as below:

$$q(\overline{\mathbf{y}}|c) = \sum_{k=1}^{K} \alpha_k \mathcal{N}(\boldsymbol{\mu}_k, \overline{\boldsymbol{\Sigma}}_k) \tag{10}$$

where K is the number of Gaussian components, *i.e.*, multiple modes. α_k, $\boldsymbol{\mu}$, and $\overline{\boldsymbol{\Sigma}}_k$ are the probability, mean and covariance matrix of the k-th Gaussian component, respectively.

Then, we estimate the GMM conditioned on the prior information. Since the future trajectory is influenced by the temporal dependence and spatial information together, the sparse spatio-temporal features F are considered as the prior information to estimate the parameters of the GMM by a fully connected layer (FC). Due to the single provided real future trajectory (ground truth) in each training iteration, we use the mean location $\bar{y}$ of $T+q$ samples generated from the full trajectory of the current iteration to estimate the mean location of the population. Therefore, a negative log-likelihood loss function $\mathcal{L}_{\text{NLL}}$ is leveraged to optimize the GMM iteratively supervised by $\bar{y}$ as follows:

$$\mathcal{L}_{\text{NLL}} = \frac{-\sum_{n=1}^{N} \log(\mathbb{P}(\bar{y}| \sum_{k=1}^{K} \hat{\alpha}_k \mathcal{N}(\hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k)))}{N}, \tag{11}$$

where $\hat{\alpha}_k$, $\hat{\boldsymbol{\mu}}_k$, and $\hat{\boldsymbol{\Sigma}}$ are the estimated probability, mean, and covariance matrix of the k-th Gaussian component, respectively. N is the number of pedestrians.

After obtaining the GMM, we can sample multiple mean locations to represent the multimodal behaviors and then predict the diverse future trajectories via the sampled mean locations to cover multimodal behaviors. Thanks to the mean location, the prediction process is intepretable and controllable as described in Section 4.4. Furthermore, it could reduce the "stress" of the model compared with estimating the specific mode via the full trajectory and thus achieve better performance, as discussed in Section 4.2.1.

3.7 Multimodal Trajectory Prediction

After obtaining the social interaction representation and interpretable intention representation, the final process predicts diverse future trajectories to cover multimodal behaviors. Due to the single provided future trajectory (ground truth), the model will collapse into "mean mode" and thus fail to cover multimodal behaviors if we directly learn multiple future trajectories supervised by the single ground truth [7]. Thus, we predict them greedily with the teacher-forcing strategy in the training phase, while the diverse future trajectories are predicted in the inference phase.

Greedy Prediction in Training Phase. In the training phase, we employ the teacher-forcing strategy to avoid weak capacity of model in early training stage. Namely, we directly encode the mean location $\bar{y}$ of the current iteration instead of sampling one from the GMM to gain the feature of mean location. Then, we concatenate the feature of mean location and sparse spatio-temporal features F to obtain the predicted trajectory $\hat{\mathbf{Y}}$ of a specific mode via a multilayer perceptron (MLP). The loss function supervised by the ground truth $\mathbf{Y}$ is shown by

$$\mathcal{L}_{\text{REG}} = \frac{\sum_{n=1}^{N} ||\mathbf{Y} - \hat{\mathbf{Y}}||_2^2}{N}, \tag{12}$$

where N is the number of pedestrians.

The whole network can be trained in an end-to-end way by minimizing the total loss $\mathcal{L}$ as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{NLL}} + \lambda_2 \mathcal{L}_{\text{REG}}, \tag{13}$$

where the λ_1 and λ_2 are used to balance the total loss $\mathcal{L}$.

Multimodal Prediction in Inference Step. In the inference phase, we disentangle the GMM into K separated Gaussian distributions $\{\mathcal{N}(\hat{\boldsymbol{u}}_k, \hat{\boldsymbol{\Sigma}}_k)\}_{k=1}^{K}$ and sample multiple mean locations from each separated component to predict diverse future trajectories. Compared with sampling from the GMM, sampling from the disentangled distributions can ensure the model plays each mode fairly and thus improve the diversity of predicted trajectories.

4 Experiments and Discussions

In this section, we evaluate the pedestrian trajectory prediction performance of our proposed IMP, and carry out detailed ablation studies to explore the performance contribution of each component in IMP. Meanwhile, we compare our method with existing state-of-the-art methods on two standard benchmark datasets.

4.1 Experimental Setting

Evaluation Datasets. We evaluate our method on ETH [13], UCY [14], Stanford Drones Dataset (SDD) [15], nuScenes [16], and Argoverse [17]. ETH [13] and UCY [14] are the most widely used benchmarks for pedestrian trajectory prediction. They contain four unique traffic scenes, where ETH includes ETH and HOTEL scenes, and UCY includes UNIV and ZARA scenes. There are $1,536$ individual pedestrians with challenging interactive scenes, such as pedestrian crossing, group walking, and collision avoidance. Following prior works [10], [11], we divide ETU and UCY into five subsets, where ETH includes ETH and HOTEL subsets, and UCY includes UNIV, ZARA1, and ZARA2 subsets. We use the leave-one-out [10] strategy to execute our method, *i.e.*, training on four subsets and testing on the resting one. The trajectories in ETH-UCY are recorded in the world coordinate system with meter as a unit. We use the egocentric trajectory normalization [46], [47] to normalize the trajectory on ETH-UCY.

SDD [15] is a large-scale benchmark for pedestrian trajectory prediction from a bird's eye view. It collects multi-agent trajectories (*e.g.*, pedestrians, bicyclists, skateboarders, cars, buses, and golf carts) on a university campus. Over $11,000$ individual pedestrians generate more than $185,000$ interactions among pedestrians and $40,000$ interactions between pedestrians and scenes. We use standard training and testing splits as in prior works [10], [45]. The trajectories in SDD are recorded in the pixel coordinate system with pixel as a unit. The last point of trajectory is translated into the origin to normalize the trajectory on SDD.

nuScenes [16] and Argoverse [17] are two large-scale autonomous driving datasets focusing on vehicle trajectory prediction. nuScenes [16] contains $1,000$ driving scenes and the corresponding HD semantic maps with 11 semantic classes sampled at 2Hz. Argoverse [17] consists of 333K driving sequences sampled at

TABLE 1
Ablation study about interpretable intention representation on ETH-UCY in ADE/FDE metrics. The lower the better.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
LV_1	0.57/1.10	0.25/0.45	0.59/1.19	0.40/0.82	0.32/0.67	0.42/0.84
LV_2	0.56/1.05	0.26/0.47	0.58/1.18	0.40/0.84	0.32/0.67	0.42/0.84
Ours	**0.29/0.47**	**0.12/0.18**	**0.29/0.51**	**0.20/0.35**	**0.15/0.27**	**0.21/0.35**
SR_1	0.38/0.69	0.14/0.23	0.34/0.62	0.26/0.49	0.19/0.35	0.26/0.47
SR_2	0.31/0.52	0.14/0.21	**0.29**/0.52	**0.20**/0.36	0.16/0.30	0.22/0.38
Ours	**0.29/0.47**	**0.12/0.18**	**0.29/0.51**	**0.20/0.35**	**0.15/0.27**	**0.21/0.35**
$K = 1$	0.35/0.60	0.15/0.22	0.36/0.64	0.24/0.44	0.20/0.39	0.26/0.45
$K = 2$	0.31/0.53	0.13/0.20	0.30/0.54	**0.20/0.35**	0.16/0.29	0.22/0.38
$K = 4$	0.30/0.50	**0.12/0.18**	0.30/0.54	0.20/0.36	0.16/0.29	**0.21**/0.37
$K = 5$	0.30/0.48	0.13/**0.18**	0.30/0.53	0.20/0.36	0.15/0.28	**0.21**/0.36
$K = 20$	0.29/0.49	**0.12/0.18**	**0.29/0.51**	0.20/0.36	**0.15/0.27**	**0.21**/0.36
Ours ($K = 10$)	**0.29/0.47**	**0.12/0.18**	**0.29/0.51**	**0.20/0.35**	**0.15/0.27**	**0.21/0.35**
MEAN	0.14/0.30	0.06/0.11	0.12/0.29	0.09/0.22	0.07/0.17	0.09/0.21

10Hz in dense traffic, where each sequence contains one target vehicle for prediction. As this paper focuses on the pedestrian trajectory prediction, the map information is deleted to match the setting of pedestrian trajectory prediction in a flexible scene.

We observe trajectory of 8 time steps (3.2 seconds) and predict future trajectory of 12 time steps (4.8 seconds) both on ETH-UCY and SDD like existing methods. On nuScenes, we observe the trajectory of 8 time steps (4 seconds) and predict the next trajectory of 12 time steps (6 seconds). On Argoverse, we observe 2 time steps (2 seconds) trajectory and predict the subsequent trajectory of 3 time steps (3 seconds).

Evaluation Metrics. We follow the common metrics of prior works [7], [19] to evaluate the trajectory prediction performance. They are

- Average Displacement Error (ADE): Average L_2 distance between the predicted trajectory points and ground-truth future trajectory points.
- Final Displacement Error (FDE): L_2 distance between the destination of the predicted trajectory and the final destination of the ground-truth future trajectory point.

To evaluate the predicted multimodal future trajectories, we compute the ADE and FDE on 20 predicted future trajectories and report the minimum ADE and FDE to compare with the existing methods fairly.

Experimental Settings. The trajectories in training set are flipped to augment the training data [10], [45]. We set the snippet length $l = 4$ empirically and the snippet-level embedding dimension $D_e = 128$. The dimensions of D_Q and D_K in the sparse learning module are equal to 128. The number of subspaces H is 8. In the sparse attention learning block, we stack two 1×1 convolution blocks, whose input channels and output channels at each layer are set to $(8, 16)$, $(16, 16)$, $(16, 16)$, $(16, 1)$, respectively. The hidden dimension of the feed-forward layer is 256. We use two Transformer [25] blocks to model the snippet-level temporal dependence. The hidden and output dimensions of the MLP used to encode the mean location are 64 and 128, respectively. The number K of the GMM components is 10 and 1 on ETH-UCY and SDD, repectively. λ_1 and λ_2 are set to 1 in the total loss $\mathcal{L}$. Our method is trained on ETH-UCY using the AdamW optimizer for 150 epochs with data batch size 128. The initial learning rate is set to 0.001, decaying by a factor of 0.5 with an interval of 40 epochs. On SDD, the initial learning rate is 0.01 for 500 epochs with batch size 512. We use the NORMALIZE operation in Pytorch [52] to scale the embedding of large pixel values on SDD.

4.2 Ablation Study

We conduct a series of ablative experiments to evaluate the performance contribution of each component of our proposed IMP, where each component is replaced by the corresponding counterpart or removed while keeping the others unchanged.

4.2.1 *Interpretable Intention Representation*

The major contribution of our proposed method reflects on the proposed interpretable intention representation, which represents a specific mode by the mean location. We first replace it with a previously commonly used latent variable to evaluate its effectiveness. Then, multiple strategies for building the interpretable intention representation are leveraged to investigate the effectiveness of the mean location of the full trajectory. Furthermore, a hyper-parameter calibration is conducted to measure the impact of the number of modes. Finally, we provide an empirical argument to indicate whether the mean location is important for pedestrian trajectory prediction or not.

Comparison with Latent Variable-based Methods. We replace our proposed mean location with the latent variable to evaluate its effectiveness. Similar to [10], we conduct an experiment LV_1 to encode the future trajectory into a high-dimensional standard Gaussian distribution, *i.e.*, the latent space, by CVAE [53]. A sample (latent variable) is drawn from the latent space to decode the future trajectory conditioned on the sparse spatio-temporal features F. Repeatedly, multiple samples are drawn to predict the multimodal future trajectories in the inference phase. We also conduct an experiment LV_2 to encode the mean location of full trajectory into a high-dimensional standard Gaussian distribution similar to LV_1. As shown in the first block of Table 1, our method outperforms LV_1 and LV_2 by a large margin. It validates the efficacy of our proposed IMP to represent multimodal behaviors, which is the major contribution of our method.

Comparison with Other Interpretable Intention Representations. Our proposed IMP is dedicated to explicitly generating an interpretable representation to reveal the future motion behaviors of pedestrians. Besides the mean location of full trajectory, we employ two variants SR_1 and SR_2 to replace the mean location of full trajectory. SR_1 and SR_2 denote the last point (destination) and the middle point of full trajectory, respectively. Here, SR_2 is the $\lceil T/2 \rceil$-th trajectory point, assuming the length of the full trajectory is T. Since SR_1 and SR_2 carry the motion behavior of full or future trajectory, they also can build the intrepretable representation for multimodal behaviors. As shown in the second block of Table 1, the mean location (Ours) achieves the best performance, indicating its effectiveness in representing future behaviors. The results also reveal that the mean location (Ours) and the middle point (SR_2) are better than the destination (SR_1) to represent the trajectory. The reason could be that the destination or middle point is an exact point, it requires the model samples the destination/point with high accuracy, leading to greater difficulty in sampling. In contrast, the mean location is the mean of observed and future trajectory, being a coarse position to represent multimodal motion behaviors. In this case, the coarse position provides a higher error-tolerant rate than the exact position in sampling. Besides, the destination is far from the observed value, leading to higher uncertainty than the mean location or middle point. The higher uncertainty suffers from difficult destination modeling, leading to poor performance even for FDE. In addition, the mean location can reflect the global motion tendency, while the destination/middle point can only focus on the local one.

Impact of the Number K of Modes. The components of the GMM are considered as the multimodal behaviors in our method. We conduct an experiment to analyze the impact of the number K of GMM components on prediction performance. Specifically, we set K to $1, 2, 4, 5, 10, 20$, respectively. As shown in the third block of Table 1, $K = 1$ (single mode) performs worst both on ADE and FDE, while $K = 10$ achieves the best performance. All variants sample 20 mean locations from each GMM component and report the minimum ADE and FDE to make a fair comparison. The results indicate that it needs to balance the number of GMM components and the number of sampled mean locations.

Empirically Argumentation about the Mean Location. We present the theoretical support of the mean location in Theorem 1, which is leveraged to guide the distribution estimation of a specific mode. Here, we provide an empirical argumentation to indicate the significance of the mean location for pedestrian trajectory prediction. Concretely, we conduct an experiment MEAN to predict the future trajectory conditioned on the real mean location, by providing the mean location of the full trajectory in advance. We employ two simple encoders (MLP) to encode the observed trajectory and real mean location. Then, we combine the encoded two features and employ a decoder (MLP) to obtain the future trajectory prediction. Note that this prediction is determinate because we generate a single future trajectory to measure the ADE and FDE instead of selecting the best trajectory from 20 predicted trajectories. The results in the fourth block of Table 1 show that MEAN achieves a stunning performance, indicating that the mean location is crucial for pedestrian trajectory prediction.

Ablation Study on nuScenes and Argoverse. Moreover, we evaluate our major contribution (mean location) in covering the possible future motion behaviors of vehicles on nuScenes/Argoverse validation set. The map information is removed to match the setting of pedestrian trajectory prediction in a flexible scene. We employ a special case of sparse graph learning, *i.e.*, assuming a pedestrian does not interact with anyone, to accelerate the training process. All the inputs, including observed trajectory, mean location or its variants, are encoded by a two-layer MLP. As shown in Table 2, the experimental results validate the effectiveness of mean location. Specifically, the first block evaluates the effectiveness of mean location compared with the latent-based method, where LV_1 embeds the mean location into a latent space (*i.e.*, a high-dimensional Gaussian space), while LV_2 embeds the destination into a latent space. Both of LV_1 and LV_2 are implemented based on the framework of PECNet [10]. The second block evaluates the effectiveness of mean location compared with the two variants, where SR_1 and SR_2 denote the last point (destination) and the middle point of the future trajectory, respectively. The third block shows the impact of the number K of modes.

TABLE 2
Ablation study about mean location on Argoverse and nuScenes validation set in ADE/FDE metrics. The lower the better.

Method	nuScenes		Argoverse	
	ADE	FDE	ADE	FDE
LV_1	0.98	2.23	3.04	5.10
LV_2	0.97	2.00	2.98	4.65
Ours	0.50	1.02	1.27	1.86
SR_1	0.57	1.17	1.29	1.89
SR_2	0.54	1.10	1.28	1.87
Ours	**0.50**	**1.02**	**1.27**	**1.86**
$K = 1$	0.55	1.17	1.42	2.15
$K = 2$	**0.50**	**1.02**	**1.27**	**1.86**
$K = 4$	0.53	1.10	1.32	1.95
$K = 5$	0.53	1.09	1.34	2.00
$K = 10$	0.59	1.23	1.55	2.37
$K = 20$	0.69	1.47	1.79	2.80

4.2.2 Snippet-level Embedding

Our snippet-level embedding divides the observed trajectory with length T into multiple non-overlapped snippets with length l and obtains the trajectory embedding on each snippet. In this way, it reduces the computation complexity of Transformer from $\mathcal{O}(T^2)$ to $\mathcal{O}(T^2/l^2)$, as indicated in capturing snippet-level temporal dependence. We conduct an experiment to evaluate its effectiveness in reducing computation complexity while maintaining or even improving the performance. We set the snippet length l to 1, 2, 4, and 8, respectively. $l = 1$ indicates that previous methods extract spatio-temporal features on each time step. On the contrary, $l = 8$ denotes that the observed trajectory is divided into one snippet with the length as same as the observed trajectory and directly uses MLP to capture temporal dependence.

As given in Table 3, snippet-level embedding with $l = 1$ is inferior to the ones with lengths $l = 2$, 4, or 8. It indicates that it is beneficial to introduce temporal dependence into spatial interaction for trajectory prediction, where spatial interaction becomes continuous. To balance the prediction accuracy and time complexity, we choose $l = 4$ in our implementation.

4.2.3 Sparse Learning Module

Our sparse learning module is mainly to reduce the superfluous interactions by learning a sparse attention matrix. To evaluate its

TABLE 3
Ablation study about snippet-level embedding on ETH-UCY in ADE/FDE metrics. The lower the better.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
$l = 1$	0.30/0.50	0.13/0.20	0.30/0.53	0.20/0.36	**0.15/0.27**	**0.21**/0.37
$l = 2$	0.30/0.48	0.13/0.19	0.30/0.53	**0.19/0.34**	**0.15**/0.28	**0.21**/0.36
$l = 8$	**0.29**/0.48	**0.11/0.17**	0.30/0.53	0.21/0.37	0.16/0.29	**0.21**/0.36
Ours ($l = 4$)	**0.29/0.47**	0.12/0.18	**0.29/0.51**	0.20/0.35	**0.15/0.27**	**0.21/0.35**

TABLE 4
Ablation study about sparse learning module on ETH-UCY in ADE/FDE metrics. The lower the better.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
Category Interaction	0.31/0.51	0.13/0.18	**0.28**/0.51	**0.20**/0.35	0.16/0.28	**0.21**/0.36
Full Interaction	0.30/0.51	0.13/0.21	0.30/0.52	0.21/0.38	0.17/0.30	0.22/0.38
Distance Interaction	0.32/0.54	0.14/0.21	0.29/0.51	**0.20**/0.35	0.16/0.28	0.22/0.37
Ours	**0.29/0.47**	**0.12/0.18**	**0.29/0.51**	**0.20/0.35**	**0.15/0.27**	**0.21/0.35**

TABLE 5
Multimodal trajectory prediction comparison with state-of-the-art methods on ETH-UCY in ADE/FDE metrics. The lower the better.

Model	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
SGAN [7]	CVPR2018	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 [32]	CVPR2019	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.51/1.15
PITF [31]	CVPR2019	0.73/1.65	0.30/0.59	0.60/1.27	0.38/0.81	0.31/0.68	0.46/1.00
GAT [8]	NeurIPS2019	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 [8]	NeurIPS2019	0.69/1.29	0.49/1.01	0.55/1.32	0.30/0.62	0.36/0.75	0.48/1.00
STGAT [54]	ICCV2019	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 [12]	CVPR2020	0.64/1.11	0.49/0.85	0.44/0.79	0.34/0.53	0.30/0.48	0.44/0.75
SGCN [18]	CVPR2021	0.63/1.03	0.32/0.55	0.37/0.70	0.29/0.53	0.25/0.45	0.37/0.65
PECNet [10]	ECCV2020	0.54/0.87	0.18/0.24	0.35/0.60	0.22/0.39	0.17/0.30	0.29/0.48
STAR [11]	ECCV2020	0.36/0.65	0.17/0.36	0.31/0.62	0.26/0.55	0.22/0.46	0.26/0.53
PCCSNet [45]	ICCV2021	**0.28**/0.54	**0.11**/0.19	**0.29**/0.60	0.21/0.44	**0.15**/0.34	**0.21**/0.42
IMP (Ours)	-	0.29//**0.47**	0.12/**0.18**	**0.29/0.51**	**0.20/0.35**	**0.15/0.27**	**0.21/0.35**

TABLE 6
Multimodal trajectory prediction comparison with state-of-the-art methods on SDD in ADE/FDE metrics. The lower the better.

Model	Venue/Year	ADE	FDE
Sophie [32]	CVPR2019	16.27	29.38
SGAN [7]	CVPR2018	27.23	41.44
Desire [30]	CVRP2018	19.25	34.05
CF-VAE [32]	CVPR2019	12.60	22.30
SimAug [55]	ECCV2020	10.27	19.71
PECNet [10]	ECCV2020	9.96	15.88
PCCSNet [45]	ICCV2021	**8.62**	16.16
IMP (Ours)	-	8.98	**15.54**

TABLE 7
Comparison of multimodal trajectory prediction between our method and two reproduced methods on nuScenes and Argoverse validation set in ADE/FDE metrics. The lower the better.

Method	nuScenes		Argoverse	
	ADE	FDE	ADE	FDE
STAR-V [11]	1.19	2.89	2.62	4.33
PECNet-V [10]	0.97	2.00	2.98	4.65
HOME-V [56]	0.57	1.17	1.29	1.89
DenseTNT-V [37]	0.81	1.77	2.09	3.32
Ours	**0.50**	**1.02**	**1.27**	**1.86**

effectiveness, we employ the full interaction implemented by a standard self-attention [57], distance-based interaction implemented by a distance-weighted graph, and category interaction [27] implemented by a classification task to replace our space learning module while keeping others fixed. As demonstrated in Table 4, our method achieves the best performance on average. The reason could be that our method removes some superfluous interactions that disturb the model's prediction. For the same performance in ADE with Category Interaction, we speculate that the dynamic graph enforces the interaction into numerable categories, which is also a way to reduce the interaction, similar to removing superfluous interaction. In contrast, our sparse learning module not only judges whether two pedestrians interact with each other but also quantifies the interaction, not disturbing the number of categories of interaction.

In addition, we evaluate the effectiveness of max-pooling used to build a consistent sparse attention matrix in reducing superfluous interactions. As discussed above, the interaction could be inconsistent in various subspaces. Therefore, we generate the mask matrix R with a single head and use a max-pooling mechanism on global attention A to build consistent sparse attention. To verify the contribution of this operation, we generate R with various heads as the same as A. Then, an element-wise multiplication between A and R generates a sparse attention matrix with multiple heads. As the results in Table 4 show, it leads to performance degradation in ADE/FDE from $0.21/0.35$ to $0.22/0.38$ on average. This indicates the effectiveness of max-pooling in building consistent interaction.

4.3 Comparison with State-of-the-Art Methods

This section will compare our method with state-of-the-art multimodal trajectory prediction methods on ETH [13], UCY [14], SDD [15], nuScenes [16], and Argoverse [17].

ETH-UCY. Table 5 presents the comparison results of our method with state-of-the-art methods in ADE and FDE metrics. Our method significantly outperforms all the competing methods. Specifically, our method improves the performance of the previous best method PCCSNet [45] from 0.42 to 0.35 on FDE while obtaining the same ADE. PCCSNet [45] saves the hidden state of the full trajectory in memory and then selects the top ranked hidden states to represent multimodal behaviors. Note that the memory space can be regarded as a high-dimensional latent space.

Moreover, our method improves the average ADE/FDE scores of the latent variable-based methods STAR [11] and PECNet [10] from $0.26/0.53$ and $0.29/0.48$ to $0.21/0.35$. They are the previous second and third best methods, respectively. All previous methods embed the multimodal behaviors into a latent space forcibly and thus the representation of a specific mode is not interpretable.

In contrast, our method employs the simple yet interpretable intention representation to represent multimodal behaviors. It can reduce the "stress" of converting the full trajectory into a latent space. Besides, modeling the mean location can avoid the model fitting the trivial detail of trajectory and enable a better convergence supported by the central-limit theorem.

Note that our method employs sparse interaction to reduce superfluous interactions; STAR [11] and PECNet [10] leverage global interaction that is suffered by superfluous interactions, while PCCSNet [45] gives up interaction. The results also reveal that our proposed sparse interaction outperforms global interaction and the case without any interaction. This proves that it is beneficial to preserve effective interaction and meanwhile remove the superfluous interactions to facilitate trajectory prediction.

SDD. We further evaluate our method on the commonly used large-scale dataset SDD. As the results in Table 6 show, our method improves the previous best latent variable-based method PECNet [10] from $9.96/15.88$ to $8.93/15.46$ in ADE/FDE. It indicates the effectiveness of our proposed IMP against latent variable-based methods.

Compared with PCCSNet [45], our method improves the FDE from 16.16 to 15.46 and achieves a comparable ADE. The underlying reason for the minor decline in ADE is that PCCSNet prioritizes ADE to calculate the prediction performance, while our method balances the ADE and FDE to achieve an overall performance. Nevertheless, the results still validate the effectiveness of our method.

nuScenes and Argoverse. We reproduce and compare related methods, *i.e.*, latent-based methods (PECNet [10] and STAR [11]) and sampled-based methods (HOME [56] and DenseTNT [37]), with our method on nuScenes [16] and Argoverse [17]. PECNet embeds the destination into a latent space to model the multimodality, while STAR directly samples latent variables in a standard latent space to generate diverse trajectories. HOME and DenseTNT sample multiple goals and then score and select goals to model the multimodality. As this paper focuses on pedestrian trajectory prediction in a flexible motion scene (*i.e.*, without the HD map), we remove the physical information about the HD map to make a fair comparison. The future trajectory is predicted only from the trajectory information. We reproduce four variants, STAR-V, PECNet-V, HOME-V, and DenseTNT-V, referring to the pipeline of STAR, PECNet, HOME, and DenseTNT, to evaluate their prediction in a flexible scene.

For HOME-V, we use a Gaussian Mixture Model (GMM) to model the heatmap of the goal and then sample multiple goals from each component of GMM to predict multimodal future trajectories. For DenseTNT-V, we employ a GMM to model the distribution of goals and then sample multiple goals from the GMM as the sparse goals. After that, we score the sampled sparse goals and select the top-K goals. Subsequently, we generate dense goals referring to the midpoint of the region squared by the maximum and minimum of the X and Y coordinates of the top-K sparse goals. Finally, the final trajectories are predicted by those dense goals. All features of the reproduced method are obtained by a two-layer MLP. We employ a special case of sparse graph learning, *i.e.*, assuming a pedestrian does not interact with anyone, to accelerate the training process. Similar to the experiments on ETH-UCY and SDD, 20 future trajectories are predicted to represent the multi-modality of future motion state, and the minimum ADE and FDE are reported to fairly compare our method with the reproduced methods.

As the experimental results in Table 7 shown, our method outperforms all reproduced methods, indicating the effectiveness of the mean location in a flexible scene. Furthermore, we find HOME-V is superior to DenseTNT-V, indicating the dense goal cannot provide effective information in a flexible scene, *i.e.*, without the map information.

4.4 Visualization Results

We conduct qualitative analyses of our method on interpretable intention representation, sparse spatial interaction, and best-predicted trajectory.

Interpretable Intention Representation. Our method predicts the multimodal behaviors based on our proposed simple yet interpretable intention representation, *i.e.*, the mean location. Each mean location corresponds to a predicted future trajectory. Since the mean location is a 2D vector, it can be easily visualized on the image. As shown in the first row of Figure 8, the predicted trajectories exhibit obvious multimodality, and their distribution presents a "tree" structure. The results meet the typical motion patterns of the pedestrian, such as turning left/right and going straight. For the mean location marked by the yellow star, the distribution of the mean location is consistent with the diverse predicted trajectories. Namely, it indicates that the mean location is beneficial to improving the interpretability of prediction by providing the rationale behind it, which is very crucial for safety-critical applications such as autonomous driving.

Moreover, we find the mean location could achieve an interesting controllable prediction. As shown in the second row

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2023.3268110

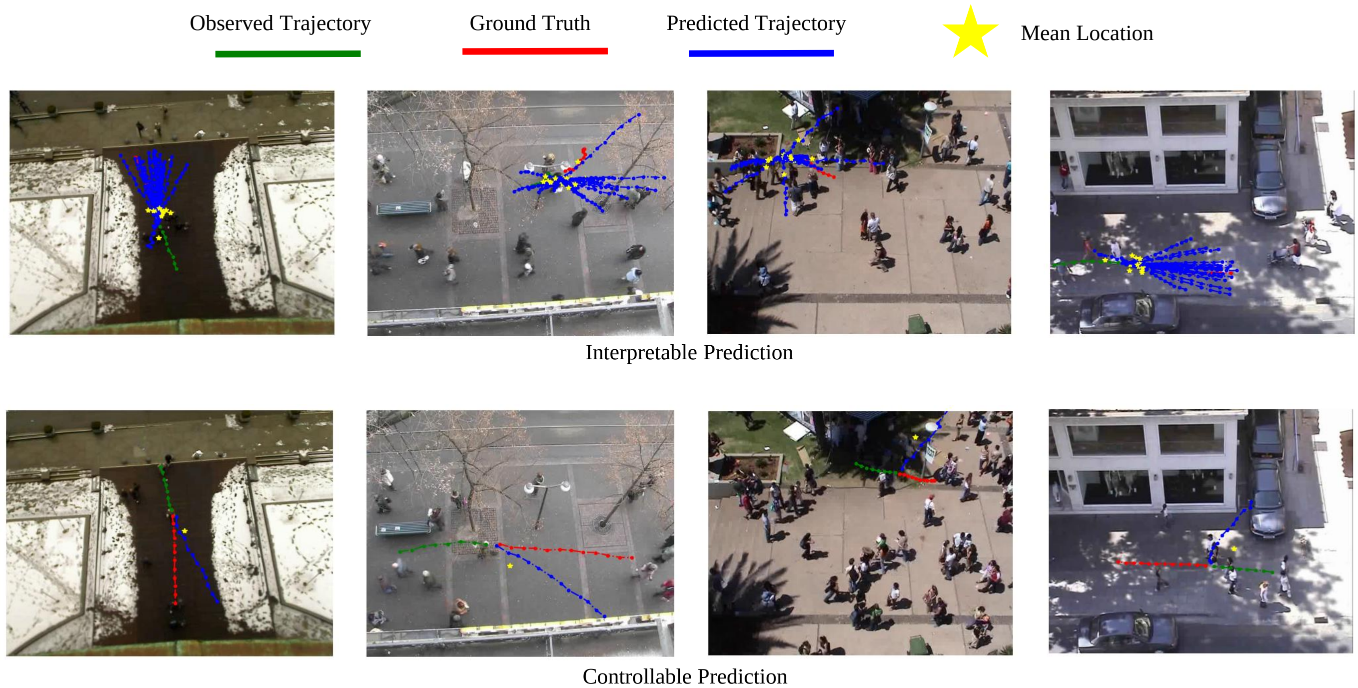


Fig. 8. Visualization on the proposed interpratable intention representation. The first row presents the diverse predicted future trajectories and their corresponding mean locations. The second row presents the controllable predicted future trajectory conditioned on the customized mean location.

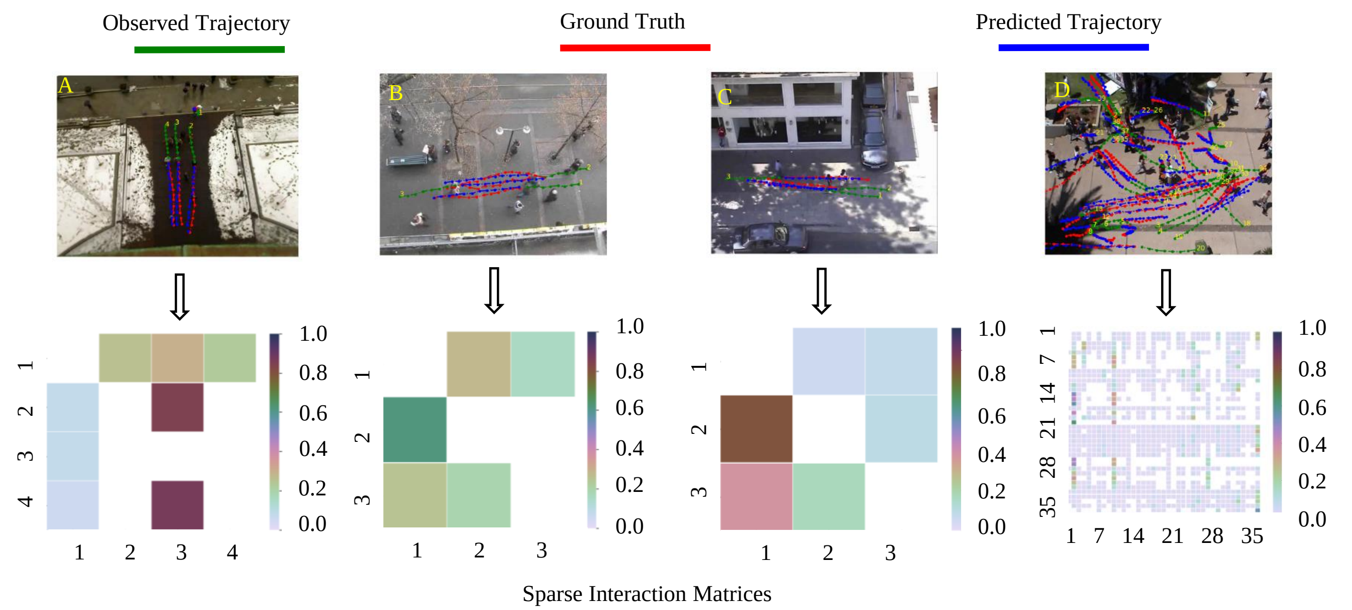


Fig. 9. Visualization on spatial sparse interaction. To highlight the interactive neighbors, we neglect the self-interaction, *i.e.*, zeroing the diagonal elements of sparse interaction matrices. The first row presents the interactive scenes, where the trajectory with the minimum FDE is selected from multimodal behaviors as the predicted trajectory. The second row presents the corresponding interactive matrices, where the white color masks the non-interactive neighbors. The color bar shows the weights of interaction. Some pedestrians are unmarked because there is no record in the dataset.

of Figure 8, we customize the mean location and then predict the future trajectory. The mean location (yellow star) is sampled randomly around the pedestrian. We can see that the predicted trajectory always follows the direction of the yellow star. In this case, the autopilot can set the mean location at the desired location to understand how a pedestrian walks to the tagged location, which is crucial to avoiding a collision. In addition, the autopilot can only take care of the interesting mode that affects driving by setting the mean location at an interesting region while neglecting other modes to reduce the computation consumption.

Sparse Spatial Interaction. We randomly select some interactive scenes from each subset of ETH-UCY to visualize the sparse spatial interaction. As illustrated in Figure 9, the first row represents the interactive scenes, where the trajectory with the minimum FDE is chosen from multimodal behaviors as the predicted trajectory. The second row represents the corresponding interaction matrices.

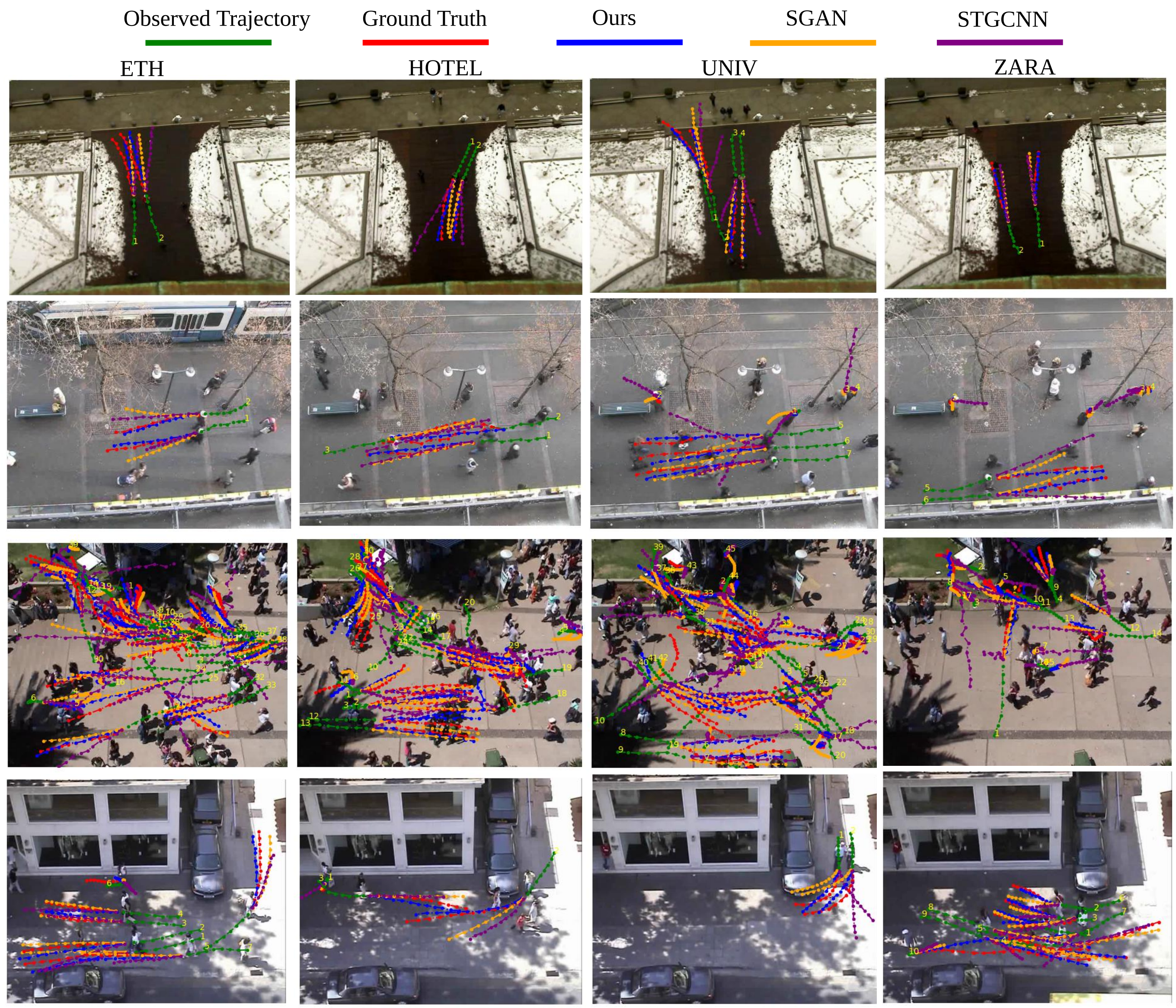


Fig. 10. Visualization of the best-predicted trajectory. The predicted trajectory with minimum FDE is selected as the best-predicted trajectory. Some pedestrians are unmarked because there is no record in the dataset.

Each matrix column represents interaction between a pedestrian to its neighbors, and the non-interactive neighbors are masked by white color. The results show that our sparse learning module can capture some interpretable interactive neighbors in different interactive scenes.

Specifically, in scene A, both pedestrians 2 and 4 interact with pedestrian 3, while there is no interaction between pedestrians 2 and 4. It makes sense because pedestrian 3 lies in the middle of pedestrians 2 and 4, and thus leads to pedestrians 2 and 4 do not influence each other. Scene B shows a common interactive scene where pedestrians 2 and 3 meet face to face, but only one of them takes a detour to avoid a collision. It reflects on the corresponding interaction matrix, where pedestrian 3 interacts with pedestrian 2 representing the behavior to avoid a collision, while pedestrian 2 does not interact with pedestrian 3 indicating going straight. Scene C shows the global interaction, where all pedestrians participate in future trajectory prediction. Scene D illustrates the dense interactions, where many pedestrians do not interact with all their neighbors despite the interaction density.

Best-predicted Trajectory. We visualize the best-predicted trajectory and compare it with two state-of-the-art methods Social-STGCNN [12] and SGAN [7]. The trajectory with the minimum FDE is chosen as the best-predicted trajectory. The visualized scenes in Figure 10 include various motion patterns such as going straight, turning left/right, avoiding collision, and walking with the dense crowd. The results show that our method has a better tendency along with the ground truth. The reason is that our estimated mean locations are adequate to cover multimodal behaviors, and our proposed sparse interaction is beneficial to refining the distribution of mean locations by reducing superfluous interactions.

5 Conclusion

This paper presents a simple yet effective pedestrian trajectory prediction method, benefiting from our newly proposed Interpretable Multimodality Predictor (IMP). It jointly models an interpretable intention representation to represent multimodal behaviors and a social interaction representation to extract the spatio-temporal features between pedestrians. The experimental results on two benchmark datasets demonstrated the effectiveness of the proposed IMP in improving prediction performance and interpretable prediction by providing the rationale behind the trajectory prediction. What's more, the mean location achieves a controllable prediction by customizing the mean location in an interesting region. Moreover, sparse interaction can further improve prediction performance by reducing superfluous interactions.

We believe the explicit interpretable intention representation, *i.e.*, mean location, has the potential to integrate multiple tasks, such as trajectory prediction and object tracking. For example, trajectory prediction can provide possible future locations by the proposed representation and thus speed up object tracking by searching in a local region instead of a global one. Moreover, although the mean location is not necessarily a waypoint an agent (such as a vehicle) can traverse, it is possible to employ the mean location in traffic scenes constrained by map information. We will explore these potential directions in our future work.

Acknowledgments

This work was supported in part by National Key R&D Program of China under Grant 2021YFB1714700, NSFC under Grants 62088102 and 62106192, Natural Science Foundation of Shaanxi Province under Grants 2022JC-41 and 2021JQ-054, China Postdoctoral Science Foundation under Grant 2020M683490, and Fundamental Research Funds for the Central Universities under Grants XTR042021005 and XTR072022001.

References

[1] X. Huang, P. Wang, X. Cheng, D. Zhou, Q. Geng, and R. Yang, "The apolloscape open dataset for autonomous driving and its application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2702–2719, 2020.

[2] Y. Luo, P. Cai, A. Bera, D. Hsu, W. S. Lee, and D. Manocha, "PORCA: Modeling and planning for autonomous driving among many pedestrians," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3418–3425, 2018.

[3] J. Wang and Y. He, "Motion prediction in visual object tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2020, pp. 10 374–10 379.

[4] D. Stadler and J. Beyerer, "Improving multiple pedestrian tracking by track management and occlusion handling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10 953–10 962.

[5] X. Shu, L. Zhang, G.-J. Qi, W. Liu, and J. Tang, "Spatiotemporal co-attention recurrent neural networks for human-skeleton motion prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3300–3315, 2022.

[6] H. Akolkar, S.-H. Ieng, and R. Benosman, "Real-time high speed motion prediction using fast aperture-robust event-driven visual flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 361–372, 2022.

[7] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2255–2264.

[8] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-BiGAT: Multimodal trajectory forecasting using bicycle-gan and graph attention networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 137–146.

[9] B. Ivanovic and M. Pavone, "The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 2375–2384.

[10] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 759–776.

[11] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 507–523.

[12] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14 424–14 432.

[13] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2009, pp. 261–268.

[14] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," in *Comput. Graphics Forum*, 2007, pp. 655–664.

[15] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 549–565.

[16] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 621–11 631.

[17] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan *et al.*, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8748–8757.

[18] L. Shi, L. Wang, C. Long, S. Zhou, M. Zhou, Z. Niu, and G. Hua, "SGCN: Sparse graph convolution network for pedestrian trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8994–9003.

[19] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 961–971.

[20] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "TrafficPredict: Trajectory prediction for heterogeneous traffic-agents," in *Proc. AAAI. Conf. Artif. Intell.*, 2019, pp. 6120–6127.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comp.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2342–2350.

[23] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2067–2075.

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

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[26] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018.

[27] J. Li, F. Yang, M. Tomizuka, and C. Choi, "EvolveGraph: Multi-agent trajectory prediction with dynamic relational reasoning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 19 783–19 794.

[28] J. Sun, Q. Jiang, and C. Lu, "Recursive social behavior graph for trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 660–669.

[29] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9813–9823.

[30] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "DESIRE: Distant future prediction in dynamic scenes with interacting agents," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 336–345.

[31] J. Liang, L. Jiang, J. C. Niebles, A. G. Hauptmann, and L. Fei-Fei, "Peeking into the future: Predicting future person activities and locations in videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5725–5734.

[32] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "SoPhie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1349–1358.

[33] J. Liang, L. Jiang, K. Murphy, T. Yu, and A. Hauptmann, "The garden of forking paths: Towards multi-future trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 508–10 518.

[34] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15 213–15 222.

[35] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 683–700.

[36] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid *et al.*, "TNT: Target-driven trajectory prediction," *arXiv preprint arXiv:2008.08294*, 2020.

[37] J. Gu, C. Sun, and H. Zhao, "DenseTNT: End-to-end trajectory prediction from dense goal sets," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15 303–15 312.

[38] P. Diananda, "The central limit theorem for m-dependent variables," in *Math. Proc. Cambridge. Philos. Soc.*, vol. 51, no. 1. Cambridge University Press, 1955, pp. 92–95.

[39] W. Hoeffding and H. Robbins, "The central limit theorem for dependent random variables," *Duke Math. J.*, vol. 15, no. 3, pp. 773–780, 1948.

[40] I. Ibragimov, "A central limit theorem for a class of dependent random variables," *Theory Probab. Appl.*, vol. 8, no. 1, pp. 83–89, 1963.

[41] S. A. Utev, "On the central limit theorem for φ-mixing arrays of random variables," *Theory Probab. Appl.*, vol. 35, no. 1, pp. 131–139, 1991.

[42] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5753–5763.

[43] X. Pan, M. Wang, L. Wu, and L. Li, "Contrastive learning for many-to-many multilingual neural machine translation," in *Proc. Annu. Meet. Assoc. Comput Linguist.*, 2021, pp. 244–258.

[44] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," *arXiv preprint arXiv:1904.10509*, 2019.

[45] J. Sun, Y. Li, H.-S. Fang, and C. Lu, "Three steps to multimodal trajectory prediction: Modality clustering, classification and synthesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13 250–13 259.

[46] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding hd maps and agent dynamics from vectorized representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 525–11 533.

[47] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 1468–1476.

[48] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Comput.*, vol. 1, no. 2, pp. 270–280, 1989.

[49] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu, "HiVT: Hierarchical vector transformer for multi-agent motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8823–8833.

[50] S. Assaad, C. Downey, R. Al-Rfou, N. Nayakanti, and B. Sapp, "VN-Transformer: Rotation-equivariant attention for vector neurons," *arXiv preprint arXiv:2206.04176*, 2022.

[51] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013.

[52] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[53] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3483–3491.

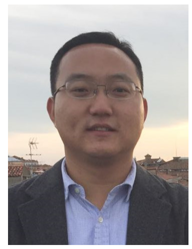
[54] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, "STGAT: Modeling spatial-temporal interactions for human trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6271–6280.

[55] J. Liang, L. Jiang, and A. Hauptmann, "SimAug: Learning robust representations from simulation for trajectory prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 275–292.

[56] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "HOME: Heatmap output for future motion estimation," in *Proc. IEEE Int. Intell. Transp. Syst. Conf.*, 2021, pp. 500–507.

[57] H. Bai, S. Cai, N. Ye, D. Hsu, and W. S. Lee, "Intention-aware online POMDP planning for autonomous driving in a crowd," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2015, pp. 454–460.

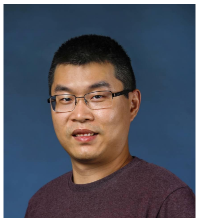
Liushuai Shi received the B.E degree in Software Engineering from Zhengzhou University, Zhengzhou, China, in 2019 and the M.S. degree in Software Engineering from Xi'an Jiaotong University, Xi'an, China, in 2022. He is currently a Ph.D. student with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University. His research interests include computer vision and deep learning. He has published two papers in CVPR and AAAI.



Le Wang (Senior Member, IEEE) received the B.S. and Ph.D. degrees in Control Science and Engineering from Xi'an Jiaotong University, Xi'an, China, in 2008 and 2014, respectively. From 2013 to 2014, he was a visiting Ph.D. student with Stevens Institute of Technology, Hoboken, New Jersey, USA. From 2016 to 2017, he was a visiting scholar with Northwestern University, Evanston, Illinois, USA. He is currently a Professor with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University, Xi'an, China. His research interests include computer vision, pattern recognition, and machine learning. He is an associate editor of Pattern Recognition Letters. He is an area chair of CVPR'2022, ICME'2022 and ICPR'2022, and a senior program committee member of AAAI'2022. He holds 13 China patents and has 17 more China patents pending. He is the author of more than 70 peer reviewed publications in prestigious international journals and conferences.

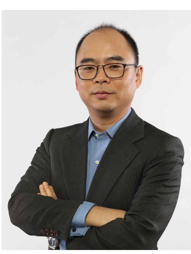
Chengjiang Long (Member, IEEE) is currently a Research Scientist at Meta Reality Labs (formerly Facebook Reality Labs) at the Burlingame CA office. Prior joining Meta (formerly Facebook, Inc.), he worked as a Principal Scientist/Tech Leader in JD Tech R&D Center at Silicon Valley (a part of JD.COM) from June 2020 to Dec 2021, and worked as a Computer Vision Researcher/Senior RD Engineer at Kitware from February 2016 to April 2020. He was an Adjunct Professor at Rensselaer Polytechnic Institute (RPI) from Jan 2018 to May 2018. He received the M.S. degree in Computer Science from Wuhan University in 2011 and a B.S degree in Computer Science and Technology from Wuhan University in 2009. He got his Ph.D. degree in Computer Science from Stevens Institute of Technology in 2015. During his Ph.D. study, he worked at NEC Labs America and GE Global Research as a research intern in 2013 and 2015, respectively. To date, he has published over 65 papers in prestigious international journals and conferences, and owns 1 patent. He is also the reviewer for more than 20 top international journals and conferences. His research interests involve various areas of Computer Vision, Computer Graphics, Multimedia, Machine Learning, and Artificial Intelligence. He is a member of IEEE and AAAI.

Sanping Zhou (Member, IEEE) received the Ph.D. degree in control science and engineering from Xian Jiaotong University, Xi'an, China, in 2020. From 2018 to 2019, he was a Visiting Ph.D. Student with the Robotics Institute, Carnegie Mellon University. He is currently an Assistant Professor with the Institute of Artificial Intelligence and Robotics, Xian Jiaotong University. His research interests include machine learning, deep learning, and computer vision, with a focus on person re-identification, salient object detection, medical image segmentation, image classification, and visual tracking.



Wei Tang (Member, IEEE) received his Ph.D. degree in Electrical Engineering from Northwestern University, Evanston, Illinois, USA in 2019. He received the B.E. and M.E. degrees from Beihang University, Beijing, China, in 2012 and 2015 respectively. He is currently an Assistant Professor in the Department of Computer Science at the University of Illinois at Chicago. His research interests include computer vision, pattern recognition and machine learning.

Nanning Zheng (Fellow, IEEE) graduated in 1975 from the Department of Electrical Engineering, Xi'an Jiaotong University (XJTU), received the M.E. degree in Information and Control Engineering from Xi'an Jiaotong University in 1981, and a Ph.D. degree in Electrical Engineering from Keio University in 1985. He is currently a Professor and the Director with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University. His research interests include computer vision, pattern recognition, computational intelligence, and hardware implementation of intelligent systems. Since 2000, he has been the Chinese representative on the Governing Board of the International Association for Pattern Recognition. He became a member of the Chinese Academy Engineering in 1999. He is a fellow of the IEEE.



Gang Hua (Fellow, IEEE) was enrolled in the Special Class for the Gifted Young of Xi'an Jiaotong University (XJTU), Xi'an, China, in 1994 and received the B.S. degree in Automatic Control Engineering from XJTU in 1999. He received the M.S. degree in Control Science and Engineering in 2002 from XJTU, and the Ph.D. degree in Electrical Engineering and Computer Science at Northwestern University, Evanston, Illinois, USA, in 2006. He is currently the Vice President and Chief Scientist of Wormpex AI Research. Before that, he served in various roles at Microsoft (2015-18) as the Science/Technical Adviser to the CVP of the Computer Vision Group, Director of Computer Vision Science Team in Redmond and Taipei ATL, and Principal Researcher/Research Manager at Microsoft Research. He was an Associate Professor at Stevens Institute of Technology (2011-15). During 2014-15, he took an on leave and worked on the Amazon-Go project. He was an Visiting Researcher (2011-14) and a Research Staff Member (2010-11) at IBM Research T. J. Watson Center, a Senior Researcher (2009-10) at Nokia Research Center Hollywood, and a Scientist (2006-09) at Microsoft Live labs Research. He is an associate editor of TPAMI, TIP, TCSVT, CVIU, IEEE Multimedia, TVCJ and MVA. He also served as the Lead Guest Editor on two special issues in TPAMI and IJCV, respectively. He is a general chair of ICCV'2025. He is a program chair of CVPR'2019&2022. He is an area chair of CVPR'2015&2017, ICCV'2011&2017, ICIP'2012&2013&2016, ICASSP'2012&2013, and ACM MM 2011&2012&2015&2017. He is the author of more than 200 peer reviewed publications in prestigious international journals and conferences. He holds 19 US patents and has 15 more US patents pending. He is the recipient of the 2015 IAPR Young Biometrics Investigator Award for his contribution on Unconstrained Face Recognition from Images and Videos, and a recipient of the 2013 Google Research Faculty Award. He is an IEEE Fellow, an IAPR Fellow, and an ACM Distinguished Scientist.