Sparse Trajectory Prediction

Journal:	*Transactions on Pattern Analysis and Machine Intelligence*
Manuscript ID	TPAMI-2024-10-2789
Manuscript Type:	Regular (S1)
Keywords:	Pedestrian Trajectory Prediction, Sparse Interaction, Transformer

Sparse Trajectory Prediction

Liushuai Shi, *Student Member, IEEE,* Le Wang, *Senior Member, IEEE,* Sanping Zhou, *Member, IEEE,* Wei Tang, *Member, IEEE,* and Gang Hua, *Fellow, IEEE*

Abstract—Pedestrian trajectory prediction is crucial for ensuring safe decision-making in intelligent robotic systems. While this task demands real-time performance, previous works have primarily focused on improving prediction accuracy, often neglecting efficiency. Dense predictions with time-consuming post-clustering steps and global interactions with quadratic computational complexity result in a trade-off between accuracy and speed. In this paper, we propose a novel Sparse Trajectory Prediction (STP) model that aims to achieve both high accuracy and real-time speed by following an efficient principle: leveraging sparse structures to achieve global effects. STP instantiates this principle within a transformer-style encoder-decoder framework. In the encoder, STP introduces irregular interaction, which builds sparse interactions with dynamic interactive positions, reducing computational complexity from quadratic to linear while maintaining global interaction. In the decoder, STP applies an early-sparsity strategy to generate sparse motion modes that represent global motion behaviors. These modes are shared across all predictions, eliminating redundant computations. By harnessing the expressive power of transformers, STP maps these sparse motion modes into multimodal future trajectories, significantly improving prediction speed while ensuring accuracy. Experimental results on four commonly used datasets demonstrate that STP maximizes both accuracy and prediction speed, achieving state-of-the-art performance and significantly improving prediction speed by about $20\times$ - $60\times$ to satisfy the real-time demand.

Index Terms—Pedestrian Trajectory Prediction, Sparse Interaction, Transformer

1 Introduction

Pedestrian trajectory prediction is critical for ensuring safe and accurate decision-making in intelligent robotics. It serves as a bridge between the perception module upstream and the planning module downstream [1], [2] in various intelligent applications, such as autonomous vehicles [3], surveillance systems [4], [5], and other motion prediction tasks [6]. However, trajectory prediction is extremely challenging due to the intricate motion multimodality present in complex, interactive environments. It requires the predictor to extract social interaction features to forecast diverse motion behaviors represented by multiple socially acceptable future trajectories.

As a real-time demanding task, both accuracy and efficiency in trajectory prediction are crucial for making safe decisions in dynamic and unpredictable traffic scenarios. However, current research prioritizes accuracy at the expense of efficiency, creating a significant bottleneck for real-world deployment. The challenge arises from the multimodal nature of future behaviors, which has led to various methods that explore different mode spaces to capture diverse motion patterns, such as explicit Gaussian spaces [10], [11], [12], latent spaces [8], [13], and memory spaces [7], [14]. The multiple motion modes are sampled repeatedly from the generated space to account for the multimodal motion behaviors, but they suffer from inaccurate prediction due to similar sampling motion modes. To pursue higher accuracy, recent works [7], [8], [14] adopt dense prediction with late-sparsity strategies, where a large number of potential trajectories are generated first. A clustering algorithm (e.g., K-means) is then applied to eliminate redundant predictions, refining the output to the desired set. While this approach improves accuracy, it introduces significant inefficiencies, as generating and clustering a large volume of predictions becomes computationally expensive. As shown in Figure 1, these methods (marked by '+') create a trade-off between accuracy and computational speed, making them ill-suited for real-time applications. Furthermore, as the number of pedestrians increases, efficiency deteriorates even more, moving further away from real-time feasibility, as

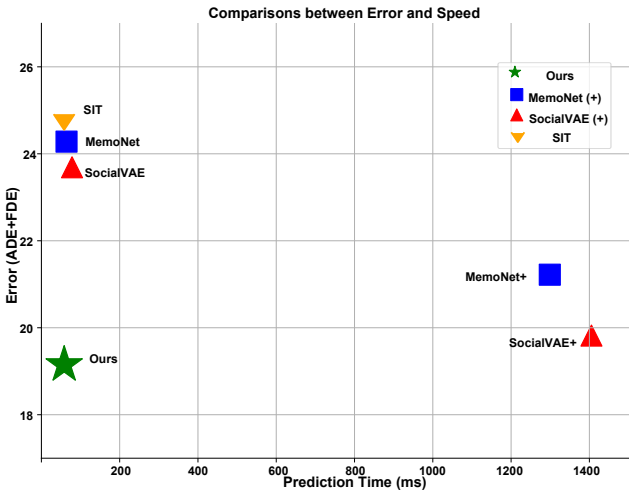


Fig. 1. The comparisons against the methods (MemoNet [7],SocialVAE [8]) with the late-sparsity (marked by '+') and the SIT [9] without late-sparsity. The late-sparsity leads to an obvious trade-off between accuracy and speed.

- *Liushuai Shi, Le Wang and Sanping Zhou are with the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and 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.)*
- *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 the Multimodal Experiences Lab, Dolby Laboratory, Bellevue, WA 98004, USA. E-mail: ganghua@gmail.com.*
- *Part of this work was done while Liushuai Shi was a visiting scholar at University of Illinois Chicago.*

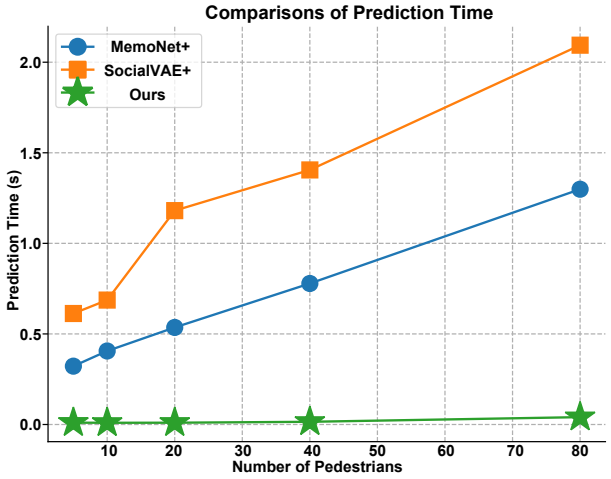


Fig. 2. The prediction time (seconds) changing with different number of pedestrians. All methods predict the trajectory with 12 time steps (4.2 seconds).

illustrated in Figure 2. More recently, interaction modeling has evolved from local interactions [15] to more comprehensive global interactions [8], [9], [11], [13], which introduce richer social behavior modeling but at the cost of quadratic computational complexity with respect to the number of neighbors. This further exacerbates the efficiency problem.

This paper introduces a novel framework, termed the Sparse Trajectory Prediction (STP), to strike a dual focus on accuracy and efficiency, delivering precise, real-time predictions, as demonstrated in Figure 1. At the heart of STP is a new efficiency principle: leveraging sparse structures to achieve global effects. STP applies this principle across two key modules of pedestrian trajectory prediction: social interaction and multimodal prediction, which are integrated within a unified transformer-based architecture.

Efficient Social Interaction. STP reduces the computational complexity of global interactions by exploiting the inherent sparsity in social interactions: a pedestrian mostly interacts with only a small subset of neighbors. A simple approach to building this sparse interaction is to fix the interaction distance [8]. However, this approach fails to model more challenging interactions (e.g., long-range interactions) and is difficult to generalize across various scenarios due to its reliance on the distance hyperparameter. This conflict between global and local interactions drives us to ask: Can we model global interaction in a sparse manner?

To address this, STP introduces an irregular interaction mechanism that learns multiple arbitrary interaction points, where the pedestrian engages with a sparse set of neighbors at each point individually, thereby improving computational efficiency. These sparse interactions are then fused around each point to achieve global interaction. Standard deformable attention [16] struggles in non-Euclidean trajectory spaces due to the irregularity of social fields. Therefore, STP employs an irregular sampling strategy that removes the constraints of Euclidean space. This approach enables STP to perform global interaction efficiently, benefiting from the reduced computational cost of sparse interactions.

Multimodal Prediction with Early-Sparsity. STP introduces an early-sparsity strategy for multimodal prediction. It treats late-sparsity as a dense-to-sparse process that extracts diverse results from dense candidates. However, this dense-sparse operation must be performed frequently for each prediction, leading to inefficiency. In contrast, our early-sparsity strategy is designed to represent global motion behaviors with a one-time dense-to-sparse process.

Specifically, early-sparsity involves a motion compressor that eliminates the need for repeated late-sparsity operations. The motion compressor condenses world knowledge from the training dataset into diverse, sparse motion modes independent of model training and inference. These motion modes are shared across all predictions, capturing global motion behaviors efficiently. Leveraging the expressive power of Transformers, these sparse motion modes are mapped into multimodal future trajectories for specific scenarios, significantly improving prediction speed while maintaining accuracy.

Unified Transformer Architecture. STP seamlessly integrates the irregular interaction and early-sparsity strategies in a unified transformer-style encoder-decoder architecture. The framework consists of three two components: *(1) a Motion Compressor*, which employs the early-sparsity to obtain the sparse motion modes by compressing the world knowledge, *(2) a Social-level Encoder*, which incorporates irregular interactions to reduce computational complexity by focusing on sparse yet effective social interactions, and *(3) a Trajectory-level Decoder*, which analyzes the relationships between the sparse motion modes to enhance diversity, and perceives encoded social interactions with the sparse motion modes to produce accurate, efficient multimodal predictions.

We evaluate STP on four commonly used datasets, *i.e.*, ETH-UCY-V1 [17], [18], ETH-UCY-V2 [17], [18], Stanford Drones Dataset (SDD) [19] and the SportVU NBA movement dataset [20]. The experimental results demonstrate the effectiveness of our proposed efficiency principle in achieving the dual focus on both accuracy and speed. Specifically, STP outperforms the state-of-the-art methods in terms of accuracy, including the late-sparsity methods. Additionally, STP offers superior efficiency, significantly improving prediction speed by about $20\times$ - $60\times$ against previous well-tuned state-of-the-art methods that rely on late-sparsity.

The contributions of this paper are summarized below.

- Both efficiency and accuracy in pedestrian trajectory prediction are crucial to downstream applications, but current research prioritizes accuracy at the expense of efficiency, creating a significant bottleneck for real-world deployment. We introduce a novel Sparse Trajectory Prediction model (STP) to achieve accurate, real-time performance by following a new efficient principle that leverages the sparse structures to achieve global effects.
- To reduce the complexity of social interaction, we propose a novel irregular interaction mechanism designed to perform global interaction in a sparse yet effective manner.
- Toward efficient multi-modal prediction, we propose a novel early-sparsity strategy to generate sparse motion modes, which are shared in all predictions to represent global motion behaviors following the proposed efficient principle.
- We integrate the irregular interaction and early-sparsity strategies in a unified Transformer architecture. It maps the sparse motion modes into specific multimodal future trajectories, effectively improving the prediction speed.
- Extensive experiments on four benchmarks demonstrate the efficiency and accuracy of our proposed method against existing state-of-the-art methods.

This paper extends our previous conference paper [21] to

achieve efficient and accurate prediction in two fundamental modules of pedestrian trajectory prediction, *i.e*, social interaction and multimodal prediction. The conference version employs the early-sparsity strategy to improve the prediction speed at the multimodal prediction modules. In complementary, this version proposes an irregular interaction to reduce the computational complexity of social interaction modules. Furthermore, the proposed early-sparsity and irregular interaction are unified as the proposed efficient principle. Finally, we present clearer motivation and more technical details about the proposed method and conduct more experiments on two extra datasets to evaluate the effectiveness of the 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

Research on pedestrian trajectory prediction is briefly categorized into two classes: prediction based on environment information (*e.g.*, semantic map) [22], [23], [24], [25], [26], [27], [28], [29], [30], [31] and prediction based on social interaction from neighbors. In this paper, we focus on the latter to achieve efficient prediction.

Considering the influence factors of human motion, the pedestrian trajectory predictor extracts social interactive features in both temporal and spatial dimensions and predicts diverse future trajectories to cover multimodal motion behaviors. This section briefly reviews related work in social interaction and multimodal trajectory prediction to present the current research status.

2.1 Social Interaction Extraction

Physical Models. Before deep learning, many works design specific physical models to forecast a deterministic future trajectory. Social force [32], motion velocity [33], and energy [34] are commonly used to model the motion behavior of pedestrians. Also, some works employ the statistical model, such as Gaussian processes [35], [36], to deal with the uncertainty of future trajectories. However, they are difficult to generalize into the complex motion patterns and spatial interactions.

Deep Learning Models. As deep learning develops in the community, most deep models in pedestrian trajectory prediction extract social interaction via fashion data-driven strategy. In temporal dimension, the recurrence-based methods [8], [13], [37] use the recurrent neural networks (RNNs) [38], [39], [40] to model the sequential motion dependence. Due to the inefficient recurrent structure, researchers refit many deep models, such as Multilayer perceptrons (MLPs) [9], [21], [41], temporal convolutional networks (TCNs) [10], [42] and temporal-attention models [11], [43] to capture the temporal features from observed trajectory.

In the spatial dimension, the pooling mechanism is first used to integrate spatial interaction in a local radius [15] or global scene [13]. Since the graph structure can better describe the trajectory scene, the graph-based methods [10], [11], [43], [44], [45], [46], [47] model the spatial interaction by graph neural networks (GCNs) [48], [49] and its variants [50]. Motivated by the success of Transformer [51], recent many works [9], [11], [24], [52], [53], [54] employ the self-attention mechanism to extract spatial features. Even [55] finetunes a large language model (LLM) to generate the social interactive features. However, the self-attention module in Transformer suffers from quadratic computation, and irrelevant interactions could influence the modeling of social interaction, increasing the risk of overfitting. Existing method [11] models the sparse attention to drop out the redundant interactions while it learns adaptive attention mask and thus still undergoes the quadratic computation. In contrast, our STP proposes an irregular interaction to engage with interested neighbors adaptively with a sparse style, thus reducing the computational complexity into a fixed window. What's more, prior transformer-based methods [52], [53] only focus on the social interaction extraction in spatial and temporal dimensions. In contrast, our STP unifies the pedestrian trajectory prediction modules, *i.e.*, social interaction, and multimodal trajectory prediction, into an encoder-decoder transformer architecture, achieving efficient and accurate prediction.

2.2 Multimodal Trajectory Prediction

Due to the motion multimodality [13], [22], pedestrians could take various motion behaviors represented by diverse future trajectories. There are two major generative strategies to deal with such multimodal prediction tasks. The former encoders the future trajectories into a latent space by a generative model, such as generative adversarial networks (GANs) [13], [24], conditional variational autoencoder (CVAE) [8], [41], [53], [56] and diffusion model [57]. They sample multiple latent variables in this space to decode multimodal future trajectories. Specifically, STAR [52] directly samples multiple latent variables and fuses them into social features to enforce the model outputting multimodal results. The latter assumes the trajectory points follow a Gaussian distribution [10], [11] or Gaussian Mixture Model (GMM) [12], [43] and estimate this distribution to obtain an explicit space, where the multimodal future trajectories are also obtained via multiple random samplings. In addition, some works [7], [14] employ the explicit memory bank to store multiple trajectory instances. SIT [9] attempts to build a hand-designed independent trajectory anchor to improve diversity. Due to the repeated random sampling, the sampled trajectories deviate from diversity, leading to inaccurate prediction. In addition, the memory banks suffer from the unbalance of the dataset, and the models suffer from strong bias when selecting the dominant trajectories, reducing diversity.

The late-sparsity strategy enhances prediction diversity to pursue accurate prediction further. PECNet [41] changes the variance of latent space in the sampling process. AgentFormer [53] penalizes the pairwise distance of predicted trajectories. However, a more effective method is using dense to sparse post-processing [7], [8], [14]. They first sample many trajectories and then cluster them into the desired number of trajectories. Unfortunately, this post-processing step suffers from the expensive prediction time and loses the probability of predicted trajectories. In contrast, STP employs an early-sparsity strategy to compress the world knowledge from the training data into sparse motion modes. By harnessing the expressive power of transformers, STP maps these sparse motion modes into multimodal future trajectories, significantly improving prediction speed while ensuring accuracy. MTR [58] is a concurrent work with our conference version [21] compressing the goal point to represent multimodal future trajectories. MTR focuses on vehicle trajectory prediction with rich HD maps and traffic elements (*e.g.*, lane and traffic sign) to restrict the movement of traffic agents. In contrast, pedestrian trajectory prediction focuses on the social interaction between pedestrians without strong rule constraints.

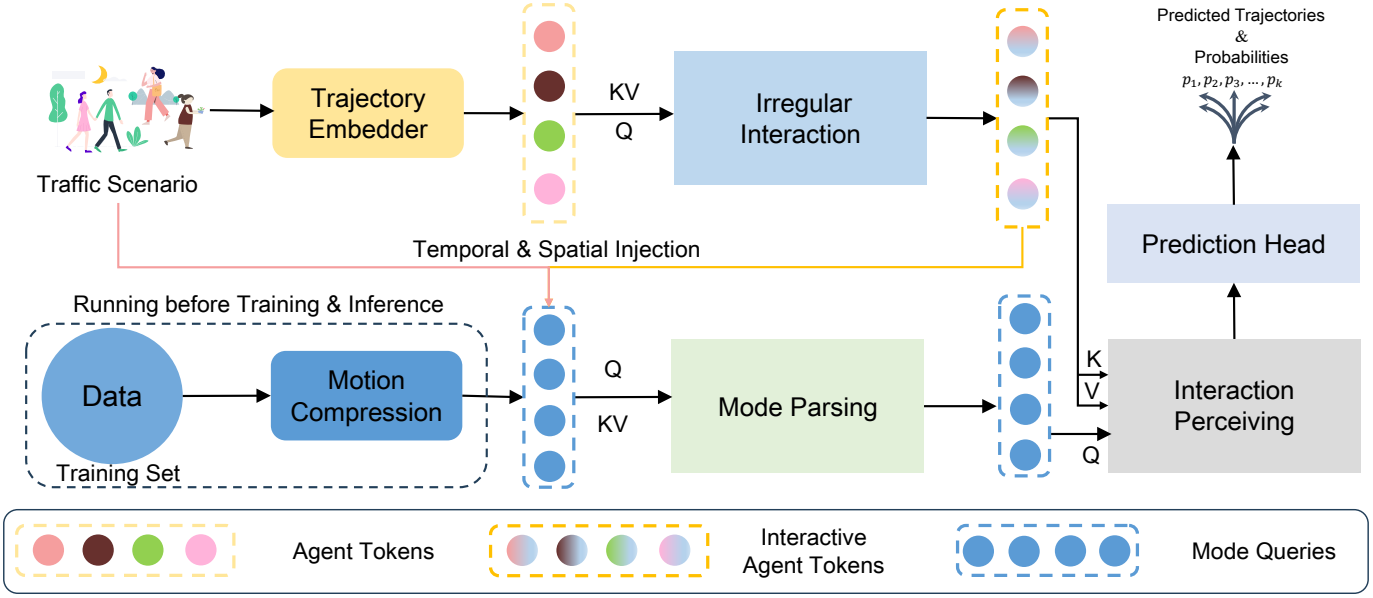


Fig. 3. The framework of STP. Before the model training and inference, STP employs the early-sparsity, where a motion compression module compresses the world knowledge from the entire training data into sparse mode tokens to represent multimodal motion behaviors. In the model training stage, the trajectory embedder generates agent tokens as the trajectory temporal feature for each pedestrian in the traffic scenario. After that, the social-level encoder builds the irregular interaction to generate the interactive agent tokens. Simultaneously, the generated sparse mode tokens are injected into the center pedestrian to obtain the mode queries. Finally, a mode parsing block distinguishes mutual relationships across mode queries, and an interaction perceiving block introduces the interactive information from the interactive agent tokens to predict the multimodal results.

3 Proposed Method

3.1 Problem Definition

Given the observed trajectories of multiple interactive pedestrians, pedestrian trajectory prediction aims to forecast the corresponding future trajectory. Assume that a traffic scenario with length T contains N pedestrians. We extract N trajectory coordinate sequences $\{x_t^n, y_t^n\}_{t=1,n=1}^{T,N}$ for each pedestrian n at time step t. The trajectory model observes the past sub-trajectories $\{x_t^n, y_t^n\}_{t=1,n=1}^{T_{obs},N}$ and predicts the future sub-trajectories $\{x_t^n, y_t^n\}_{t=1,n=1}^{T_{fut},N}$, where $T = T_{obs} + T_{fut}$. Due to the multimodality of pedestrian motion behaviors, the pedestrian could take multiple possible future trajectories. Therefore, the trajectory predictor is required to forecast diverse future trajectories, but only a single true future trajectory (ground truth) is provided for model training.

3.2 Method Overview

Our proposed Spare Trajectory prediction Model (STP) aims to achieve a dual focus both on efficiency and accuracy following an efficient principle. STP is packed into a transformer-style encoder-decoder architecture to forecast diverse future behaviors with three modules: motion compression, social interaction, and multimodal prediction. Thus, STP tokenizes the extracted various features to cater the transformer terminology. For the motion compression, the early-sparsity strategy is employed to perform the global motion behaviors with spare motion modes. To avoid the frequent and time-expensive late-sparsity, the world knowledge from the entire training data is compressed into spare motion modes, termed as sparse mode tokens, before the model training and inference, which are shared for all predictions. For the social interaction, a trajectory embedder captures the trajectory temporal features, named agent tokens, for each pedestrian in the traffic scenario. To perform global interaction with lower computational complexity, a social-level encoder receives the agent tokens to build irregular interaction with sparse pedestrians in the non-Euclidean trajectory space, producing interactive agent tokens. For the multimodal prediction, a mode embedder injects the sparse motion modes into a specific scenario to obtain mode queries. Leveraging the strong expressiveness of transformers, a trajectory-level decoder maps the mode queries into multimodal future trajectories by parsing their relationships and perceiving the interaction from interactive agent tokens, effectively improving the prediction speed and accuracy.

3.3 Motion Compression

Instead of relying on the frequent and time-expensive late-sparsity, STP employs an early-sparsity strategy to improve prediction speed. Following the efficient principle, this strategy involves a motion compressor, which condenses the world knowledge from the training data into sparse motion modes to represent the global motion behaviors. To achieve this, the motion compressor first employs two rigid transformations to align the training trajectories and then uses a one-time distance-based measurement to compress them into spare motion modes.

Trajectory Transformation. Given a fixed view, the trajectory is invariant to the rigid transformation. For example, a pedestrian going straight and then turning left shows the same behavior after applying translation or rotation to the trajectory of this pedestrian. For the training trajectories with length T, the front sub-trajectories with length T_{obs} are the observed trajectories, while the next sub-trajectories with length T_{fut} are the future trajectories. We first translate the T_{obs} trajectory points of the trajectories into the origin of the coordinate system. Then, the initial trajectory points of the translated trajectories are rotated to the positive X-axis. In this case, the direction of future trajectories is aligned to a relatively fixed region. That is, the distance between trajectories with similar motion behaviors is small. Thus, we can explicitly obtain the

diverse motion representations by a distance measurement strategy to cover global motion behaviors.

Distance Measurement. Based on the aligned training trajectories, we use the L_2 distance measurement to compress the trajectories into sparse motion modes. To perform the global motion behaviors with a sparse form, a one-time clustering operation is used on the aligned training trajectories to obtain L centers $\mathbf{C} \in \mathbb{R}^{L \times T_c \times 2}$ as the sparse motion modes, where $T_c \in [1, 2, ..., T]$ is the length of the spares motion modes. The value of T_c depends on the size of the clustered training trajectories. For example, $T_c = T_{fut}$ when we cluster the training future trajectories and $T_c = 1$ when we cluster the final point of training trajectories.

Thanks to our motion compression, the frequent and time-expensive late-sparsity for each prediction is reduced to a one-time compression for all predictions before the model training and inference step, effectively improving the prediction speed. The generated sparse motion modes are fed into the later decoder part to generate multimodal future trajectories.

3.4 Social Interaction

The social interaction module models the interactions among pedestrians in a traffic scenario. To avoid the quadratic global interaction, STP builds an irregular interaction to perform global interaction using the efficient sparse interaction guided by the proposed efficient principle. Specifically, STP first uses a trajectory embedder to generate the agent tokens and then employs a social-level encoder to produce the interactive agent tokens.

3.4.1 Trajectory Embedder

Before modeling social interaction, STP employs a trajectory embedder to capture the trajectory temporal dependence as the agent tokens. Given a specific traffic scenario with N pedestrians, we obtain N trajectory sequences $\mathbf{X} \in \mathbb{R}^{N \times T_{obs} \times c}$, where T_{obs} is the trajectory length and c is the dimension of a trajectory point. Since there exist invalid trajectory points caused by the missing detection or tracking, we apply the max-pooling operation on the temporal channel to obtain the individual temporal feature for each pedestrian. Due to the sequence-independent pooling, the one-hot sequential encoding is concatenated with the trajectory points to provide the temporal information before the trajectory embedding. A PointNet [59] based network is employed to generate agent tokens $\mathbf{E}_a \in \mathbb{R}^{N \times D}$, as follows:

$$\mathbf{E}^a = \phi(\mathbf{X}, \mathbf{W}^a) + \mathbf{b}^a, \quad (1)$$

where D represents the feature dimension, ϕ represents the stacked multi-layer perceptrons with the batch normalization and ReLU activation function. $\mathbf{W}^a$ and $\mathbf{b}^a$ are the learnable parameter matrices and bias, respectively.

3.4.2 Social-level Encoder

Based on the efficient principle, the social-level encoder performs global interaction in a sparse manner to improve computational efficiency. Concretely, an irregular attention is proposed to learn the multiple arbitrary interactive positions, where the sparse interaction is built around each position individually, and the global interaction is performed by fusing multiple sparse interactions. To this end, STP explores two strategies to achieve this irregular interaction. The first strategy is irregular grid interaction, which transforms the non-Euclidean trajectory space into an irregular grid space. Therefore, the deformable attention [16] can be used to build irregular interaction without the Euclidean constraint. The second strategy is irregular sampling interaction, which further breaks the grid constraint to directly model the sparse interaction between the interactive positions and their surrounding neighbors.

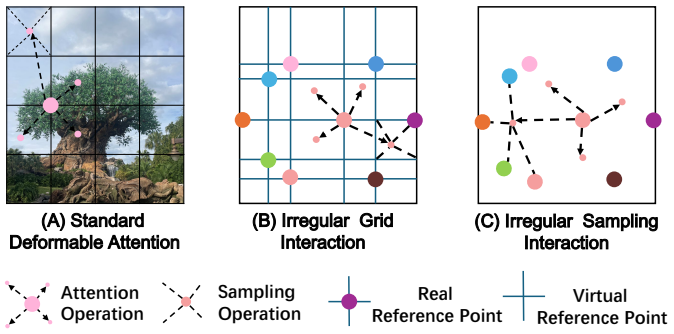


Fig. 4. Illustration of different irregular interactions. (A) is the standard deformable attention in an Euclidean space, while (B) and (C) are the proposed irregular grid interaction and irregular sampling interaction in a non-Euclidean space, respectively.

Preliminaries. The deformable attention (DA) is developed from the deformable convolution [60] to build attention with a small set of sampled keys. Compared to the self-attention mechanism [51], it generates learnable key positions and attention scores to achieve adaptive attention. Given an input feature map $\mathbf{I} \in \mathbb{R}^{C \times H \times W}$, the deformable attention is calculated by

$$\mathrm{DA}(\mathbf{z}, \mathbf{p}, \mathbf{I}) = \sum_{m=1}^{M} \mathbf{W}_m [\sum_{k=1}^{K} A_{mk} \cdot \mathbf{W}'_m \mathbf{I}(\mathbf{p} + \Delta \mathbf{p}_{mk})], \quad (2)$$

where $\mathbf{z}$ is the content feature of a query element with the 2D positions (reference points) $\mathbf{p}$, m and k index the attention head and sampled keys, respectively. K ($K \ll HW$) is the number of sampled keys. $\Delta \mathbf{p}_{mk} \in \mathbb{R}^2$ and $\{A_{mk} | \sum_{k=1}^{K} A_{mk} = 1\}$ represent the learnable offset and learnable scalar normalized attention score of the k^{th} sampled key in the m^{th} attention head, respectively. Both $\Delta \mathbf{p}_{mk}$ and A_{mk} are obtained by a learnable linear projection. $\mathbf{I}(\mathbf{p} + \Delta \mathbf{p}_{mk})$ is the key generation function implemented by bilinear interpolation due to the standard Euclidean space of the feature map. Unfortunately, this Euclidean requirement impedes the deformable attention to work in a non-Euclidean space.

Irregular Grid Interaction. The first strategy transforms the traffic scenario into an irregular grid as shown in Figure 4.(B). Given the traffic scenario $\mathbf{X} \in \mathbb{R}^{N \times T_{obs} \times c}$, we extract the current positions $\mathbf{p} \in \mathbb{R}^{N \times 2}$, *i.e.*, the position of the last trajectory point. The grid is generated via pairwise vertical connection between different abscissas and ordinates of these trajectory points. Since the density of the grid is not homogeneous compared to the images, we call this grid as irregular grid. That is, the cells in the grid have different sizes. For N trajectory points, the irregular grid has N^2 intersections, which are considered as the reference points in our irregular grid interaction. We name these intersections equal to the current positions real reference points, while the others are named virtual reference points. The content features of real reference points are initialized by the agent tokens, while the counterparts of the virtual reference points are initialized to zero.

Thanks to our real and virtual reference points, we can directly operate the standard deformable attention assisted by the generated

irregular grid to model the expected irregular interaction. The arbitrary interactive positions are obtained by the learnable offsets. The sparse neighbors are generated by the bilinear interpolation from the grid intersections around the interactive positions. Different from the encoder of the standard deformable attention, we only apply the attention on the real reference points to further reduce the computational complexity.

Irregular Sampling Interaction. Since the irregular grid exists many virtual reference points, leading to invalid interactions with zero content features. The second strategy, *i.e.*, irregular sampling interaction, is employed to break grid constraint as illustrated in Figure 4. (C). The irregular sampling interaction considers the discrete traffic scenario as an implicit continuous social interactive field based on a natural fact: an arbitrary position in the traffic scenario is affected by the nearby pedestrians.

Given the agent tokens $\mathbf{E}^a \in \mathbb{R}^{N\times D}$ with the positions $\mathbf{p} \in \mathbb{R}^2$, we first generates K learnable offsets $\{\Delta\mathbf{p}_k \in \mathbb{R}^{N\times 2}\}_{k=1}^{K}$ via a learnable linear projection on $\mathbf{E}^a$. Starting at the k^{th} learnable offset $\Delta\mathbf{p}_k \in \mathbb{R}^{N\times 2}$, the interactive position $\mathbf{p}_k^{\text{int}} \in \mathbb{R}^{N\times 2}$ is obtained by the vector addition between the the positions $\mathbf{p}$ and the corresponding offset $\Delta\mathbf{p}_k$.

To build the sparse interaction at the interactive position $\mathbf{p}_k^{\text{int}}$, two types of positional embeddings are used to encode the positions $\mathbf{p}$ and interactive points $\mathbf{p}_k^{\text{int}}$, formulated by:

$$\begin{aligned}\mathbf{P}_k^{\text{int}} &= \varphi(\mathbf{p}_k^{\text{int}}, \mathbf{W}^{\text{int}}) + \mathbf{b}^{\text{int}},\\ \mathbf{P}^{\text{cur}} &= \varphi(\mathbf{p}, \mathbf{W}^{\text{cur}}) + \mathbf{b}^{\text{cur}},\end{aligned} \tag{3}$$

where $\mathbf{P}_k^{\text{int}} \in \mathbb{R}^{N\times D}$ is the positional embedding of $\mathbf{p}_k^{\text{int}}$ to represent the interaction where the pedestrian does. $\mathbf{P}^{\text{cur}} \in \mathbb{R}^{N\times D}$ represents the current positions of neighbors. $\mathbf{P}^{\text{cur}}$ is obtained before the interaction and shared at each interaction block.

Subsequently, we generate the dynamic query and static key to prepare the sparse interaction around the interactive position $\mathbf{p}_k^{\text{int}}$, as follows:

$$\begin{aligned}\mathbf{Q} &= \varphi(\mathbf{E}_a + \mathbf{P}_k^{\text{int}}, \mathbf{W}^q) + \mathbf{b}^q\\ \mathbf{K} &= \varphi(\mathbf{E}_a + \mathbf{P}^{\text{cur}}, \mathbf{W}^k) + \mathbf{b}^k,\\ \mathbf{V} &= \varphi(\mathbf{E}_a, \mathbf{W}^v) + \mathbf{b}^v,\end{aligned} \tag{4}$$

where φ represents a learnable linear projection. $\mathbf{Q} \in \mathbb{R}^{N\times D}$ is the dynamic query with the dynamic positional information. $\mathbf{K} \in \mathbb{R}^{N\times D}$ is the static key with static positional information and $\mathbf{V} \in \mathbb{R}^{N\times D}$ is the value vector.

Therefore, the sparse interaction can be modeled with the nearest S neighbors around the interactive position $\mathbf{p}_k^{\text{int}}$, as follows:

$$\begin{aligned}\tilde{\mathbf{K}}_k, \tilde{\mathbf{V}}_k &= \mathcal{D}(\mathbf{K}, \mathbf{V}, \mathbf{p}, \mathbf{p}_k^{\text{int}}),\\ \mathbf{Z}_{mk} &= \text{softmax}(\frac{\mathbf{Q}_m \tilde{\mathbf{K}}_{mk}^T}{\sqrt{d}})\tilde{\mathbf{V}}_{mk},\end{aligned} \tag{5}$$

where $\mathcal{D}$ represents the distance function to find the nearest S neighbors. $\tilde{\mathbf{K}}_k \in \mathbb{R}^{N\times S\times D}$ and $\tilde{\mathbf{V}}_k \in \mathbb{R}^{N\times S\times D}$ are corresponding key vectors and value vectors of the nearest S neighbors, respectively. m indexes the m^{th} attention head. d represents the feature dimension of each head's query, key, and value vector. $\mathbf{Z}_{mk} \in \mathbb{R}^{N\times d}$ is the sparse interactive features of m^{th} attention head at the interactive position $\mathbf{p}_k^{\text{int}}$.

For the K interactive positions, we can obtain K sparse interactive features $\mathbf{Z} \in \mathbb{R}^{N\times S\times D}$ by concatenating K head-flattened sparse interactive features.

To perform the global interaction with the obtained sparse interactive features, we use an adaptive fusion, as follows:

$$\mathbf{S} = \varphi(\mathbf{Z}, \mathbf{W}^s) + \mathbf{b}^s, \tag{6}$$

where $\mathbf{S} \in \mathbb{R}^{N\times K\times 1}$ is the fusion score. Thus, the global interactive features $\mathbf{F} \in \mathbb{R}^{N\times D}$ are obtained by a weighted fusion operation as follows:

$$\begin{aligned}\hat{\mathbf{Z}} &= \varphi(\mathbf{Z}, \mathbf{W}^z) + \mathbf{b}^z,\\ \mathbf{F} &= \hat{\mathbf{Z}}^T \text{softmax}(\mathbf{S}),\end{aligned} \tag{7}$$

where $\hat{\mathbf{Z}} \in \mathbb{R}^{N\times K\times D}$ is the transformed features. The softmax function normalizes $\mathbf{S}$ on the second dimension.

Following the standard Transformer [51], $\mathbf{F}$ is fed into a layer normalization and a feed-forward network with residual connection to obtain the interactive agent tokens $\mathbf{E}^{ia} \in \mathbb{R}^{N\times D}$, which are fed into the next decoder part to introduce social interactions, thus predicting social-acceptable multimodal future trajectories.

Computational Complexity. We analyze the computational complexity of our proposed two types of irregular interaction against the previously commonly used global interaction based on the self-attention mechanism. Given N agent tokens with D dimension, the complexity of computing query, key, and value is $\mathcal{O}(3ND^2)$. The complexity of computing attention score and the softmax is $\mathcal{O}(N^2D + N^2)$. The complexity of the matrix multiplication between attention score and value vector is $\mathcal{O}(N^2D)$. Thus, the total complexity of global interaction is $\mathcal{O}(2N^2D+N^2+3ND^2)$, which grows quadratically with N.

For the irregular grid interaction, the generation of the irregular grid is implemented by an unstable sorting on the x and y components, respectively. Thus, the complexity of irregular grid generation is $\mathcal{O}(2NlogN)$, while it is only generated once to share with all interaction blocks. The complexity of computing offsets and normalized attention weights is $\mathcal{O}(3NDMK)$, where M and K are the number of attention heads and sampled keys, respectively. The complexity of computing values is $\mathcal{O}(ND^2)$. The complexity of bilinear interpolation and the weighted sum in attention is $\mathcal{O}(5NKD)$. The total complexity of irregular gird complexity is $\mathcal{O}(\min(2NlogN, 3NDMK + 5NKD + ND^2))$, growing linearly-logarithmic with N.

For the irregular sampling interaction, the complexity of computing offsets is $\mathcal{O}(2NDK)$. The complexity of computing dynamic positional embedding is $\mathcal{O}(2NDK)$. The complexity of computing key and value is $\mathcal{O}(2ND^2)$. The complexity of computing query is $\mathcal{O}(NKD^2)$. The nearest S neighbors are found in two operations, *i.e.*, distance computing, and distance sort. The complexity of distance computing is $\mathcal{O}(KN)$. Since we only require the top nearest S neighbors, the complexity of distance sort can be reduced into $\mathcal{O}(KN)$. Due to the sparse interaction, the complexity of computing attention score and the softmax is $\mathcal{O}(NKSD + NKS)$. The complexity of the matrix multiplication between attention score and value vector is $\mathcal{O}(NKSD)$. In general, the total complexity is about $\mathcal{O}(4NDK + 2ND^2 + NKD^2 + 2KN + 2NKSD + NKS)$, which grows linearly with N.

3.5 Multimodal Prediction

The multimodal prediction module receives the interactive agent tokens and sparse motion modes to predict multimodal future trajectories. To this end, STP first employs a mode embedder,

introducing scenario-specific information into the sparse motion modes to generate the mode queries. Then, STP builds a trajectory-level decoder to map the generated mode queries into multimodal future trajectories by leveraging the strong expressiveness of transformers.

3.5.1 Mode Embedder

Since the sparse mode tokens generated from our motion compression represent the global motion behaviors, the mode embedder injects temporal and spatial interaction information into the sparse motion modes, respectively.

Temporal Injection. The temporal interaction is represented by the temporal dependence of a trajectory. Given a specific trajectory $\mathbf{X}_i \in \mathbb{R}^{T_{obs} \times c}$ $(i \in [1, 2, ..., N])$ and the sparse mode modes $\mathbf{C} \in \mathbb{R}^{L \times T_c \times 2}$, we concatenate $\mathbf{X}_i$ and $\mathbf{C}$ to produce the time-specific sparse motion modes $\hat{\mathbf{C}} \in \mathbb{R}^{L \times \hat{T} \times 2}$ by extracting their positional information and broadcasting their shape, where $\hat{T} = T_{obs} + T_c$. Afterward, Multiple stacked multi-layer perceptrons with the batch normalization and ReLU activation function are used on $\hat{\mathbf{C}}$ to extract corresponding features $\mathbf{E}^{m,t} \in \mathbb{R}^{L \times D}$.

Spatial Injection. We further inject the spatial interactions from neighbors into $\mathbf{E}^{m,t}$. Specifically, the corresponding interactive agent token $\mathbf{E}_i^{ia} \in \mathbb{R}^{1 \times D}$ $(i \in [1, 2, ..., N])$ is concatenated with $\mathbf{E}^{m,t}$ to obtain the final scenario-specific sparse mode modes $\mathbf{E}^m \in \mathbb{R}^{L \times D}$, as follows:

$$\mathbf{E}^m = \phi(\mathbf{E}_i^{ia} \cup \mathbf{E}^{m,t}, \mathbf{W}^s) + \mathbf{b}^s, \tag{8}$$

where $\cup$ represents the shape-related function, which first broadcasts $\mathbf{E}_i^{ia}$ to align with $\mathbf{E}^{m,t}$ and then concatenates with them. ϕ is a multi-layer perceptron with a ReLU activation function. $\mathbf{E}^m$ are considered as the mode queries to generate multimodal future trajectories through the next decoder.

3.5.2 Trajectory-level Decoder

The trajectory-level decoder maps the motion queries into multimodal future trajectories with two blocks, *i.e.*, the mode parsing block, and the interaction perceiving block. The former parses the relationship across mode queries, and the latter perceives the encoded interactive features.

Mode Parsing. Unlike the above social-level encoder to obtain interactive features across pedestrians, this block parses the relationships across various mode queries to distinguish each other. Given the mode queries $\mathbf{E}^m$, the mode parsing block employs the multi-head self-attention mechanism on $\mathbf{E}^m$ to generate interactive mode queries. A layer-normalization layer with the residual connection [61] is stacked behind the attention operation. Note that we do not add the positional embedding in this self-attention block because the mode queries have included the positional information.

Interaction Perceiving. This block helps the interactive mode queries to perceive the social interactive information, thus predicting socially acceptable future trajectories. Specifically, a multi-head cross-attention mechanism is employed to achieve it, where the interactive mode queries are considered as the queries, and the interactive agent tokens generated from the social-level encoder are considered as the keys and values. Following the standard Transformer, a layer-normalization layer and a feed-forward network with residual connection are stacked behind the multi-head cross-attention mechanism. Note that positional embedding is unnecessary because the trajectory coordinates show the positional information of pedestrians.

3.5.3 Trajectory Prediction

This module contains a dual prediction and a guided prediction, where the former predicts the final multimodal future trajectories and the latter guides the feature learning of the social-level encoder.

dual Prediction. Most previous methods [7], [8], [14] predict diverse future trajectories but neglect the probabilities of predicted trajectories in pedestrian trajectory prediction. It is disadvantageous to make safe decisions. Here, we use dual prediction heads to simultaneously achieve regression and classification tasks. Specifically, a regression head and a classification head are employed on the final features obtained from the above prediction decoder to forecast diverse future trajectories and corresponding probabilities, respectively. The regression head and classification head are implemented by a multi-layer perceptron with the ReLU activation function, respectively. Different from the naive transformer, the regression and classification head are decoded into full diverse future trajectories and corresponding probabilities in parallel, not the autoregressive style.

Guided Prediction. To fully use the information of neighbor future trajectories, the guided prediction uses a regression head to predict neighbor future trajectory on the interactive agent tokens $\mathbf{E}^{ia}$ to guide the interaction learning of the social-level encoder. This regression head is implemented by a learnable linear projection. Different from the existing neighbor prediction [58], we do not further encode the predicted neighbor future trajectories to enhance the interactive agent tokens.

Model Training & Inference. Due to a single provided true future trajectory (ground truth) $\hat{\mathbf{Y}}$ for multimodal trajectory prediction, we use the variety loss [13] to optimize the prediction model. Given the predicted multimodal future trajectories $\{\mathbf{Y}_i\}_{i=1}^L$ and corresponding probabilities $\{p_i\}_{i=1}^L$, we first calculate the distance between each prediction and the ground truth, and then update the prediction $\mathbf{Y}_j$ by a smooth L_1 loss $\mathcal{L}^{\text{reg}}$, where j index the prediction with the minimum distance. For the classification, the index of p_j is considered as the target $\hat{p}$ to learn the prediction probabilities by a cross-entropy loss $\mathcal{L}^{\text{cls}}$. For the guided prediction, the predicted neighbor trajectory $\{\mathbf{Y}_n^{\text{nei}}\}_{n=1}^N$ are optimized by a smooth L_1 loss $\mathcal{L}^{\text{nei}}$ by the neighbor ground truth $\{\hat{\mathbf{Y}}_n^{\text{nei}}\}_{n=1}^N$.

Finally, STP can be trained in an end-to-end way as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}^{\text{reg}} + \lambda_2 \mathcal{L}^{\text{cls}} + \lambda_3 \mathcal{L}^{\text{nei}} \tag{9}$$

where λ_1, λ_2 and λ_3 are used to balance the loss function.

In the inference step, STP outputs multiple predicted trajectories and selects the expected number of predicted trajectories with the top predicted probabilities to cover diverse motion behaviors.

4 Experiments and Discussions

In this section, we show that STP, implemented on our efficient principle, achieves remarkable accuracy performance and faster inference speed compared to existing state-of-the-art methods with late-sparsity or not. In addition, we carry out detailed ablation studies to evaluate the performance contribution of each component of the proposed method. Finally, we further evaluate the effectiveness of STP by qualitative visualization evaluation.

4.1 Experimental Setting

Evaluation Datasets. We conduct experiments on four benchmark datasets, *i.e.*, ETH-UCY-V1 [17], [18], ETH-UCY-V2 [17], [18],

[52] Stanford Drone Dataset (SDD) [19] and SportVU NBA [20] to evaluate our proposed method. ETH [17] and UCY [18] are the most widely used benchmarks for pedestrian trajectory prediction. There are $1,536$ individual pedestrians in complex interactive scenarios, such as pedestrian crossing, group walking, and collision avoidance. Both versions, V1 and V2, contain 5 subsets, where ETH includes ETH and HOTEL subsets, and UCY includes UNIV, ZARA01, and ZARA02 subsets. The primary difference between V1 and V2 is the sampling interval of the ETH subset, where V1 has a longer interval than V2 to assess the model's generalization on unbalanced data distributions. Following prior studies [7], [8], we employ a leave-one-out cross-validation method, training on four subsets and testing on the remaining subset. The trajectories are recorded using meters as the unit.

SDD [19] is another benchmark for pedestrian trajectory prediction in a bird's-eye view. Different from the ETH-UCY-V1 and ETH-UCY-V2, collected from a single scenario for each subset. SDD is collected from various scenarios. It captures the trajectories of multiple types of agents (*e.g.*, pedestrians, bicyclists, skateboarders, cars, buses, and golf carts) on a university campus. The dataset includes over 11,000 individual pedestrians, resulting in more than 185,000 pedestrian interactions and 40,000 interactions between pedestrians and other scene elements. We adopt the standard training and testing splits used in previous studies [41], [62]. The trajectories in SDD are recorded in a pixel coordinate system, using pixel as the unit.

The SportVU NBA movement dataset [20] focuses exclusively on NBA games from the 2015-2016 regular season and provides rich interactions among players in a cooperative game setting. Due to the frequent adversarial and cooperative agent interactions and non-linear motions, the interactions in this dataset differ significantly from those in the ETH, UCY, and SDD datasets. Following prior work [8], we use two subsets as benchmarks: Rebounding and Scoring, which consist of 257,230 and 2,958,480 20-frame trajectories, respectively. The average trajectory length is approximately 4 meters, with a time interval of 0.12 seconds between frames.

Sampling Interval. We observe a trajectory of 8 time steps (3.2 seconds) and predict the next trajectory of 12 time steps (4.8 seconds) on ETH-UCY-V1 and SDD datasets. For the ETH-UCY-V2 dataset, we observe a trajectory of 8 time steps (1.92 seconds) and predict the subsequent trajectory of 12 time steps (2.88 seconds) on the ETH subset, while we observe a trajectory of 8 time steps (3.2 seconds) and predict the subsequent trajectory of 12 time steps (4.8 seconds) on the remaining four subsets. For the SportVU NBA movement dataset, we observe a trajectory of 8 time steps (0.96 seconds) and predict the subsequent trajectory of 12 time steps (1.44 seconds). Similar to existing methods, we construct a margin trajectory scenario to achieve trajectory prediction, where the scenario is normalized to originate the last trajectory point and orient the movement direction of the center pedestrian.

Evaluation Metrics. We evaluate our proposed and compared methods by four metrics, *i.g.*, Average Displacement Error (ADE), and Final Displacement Error (FDE), brier-ADE, and brier-FDE. Given the true future trajectory (ground truth) $\{x_t, y_t\}_{t=1}^{T_{fut}}$ and the corresponding predicted K trajectories, ADE and FDE are used to measure the $\ell 2$ distance between ground truth and the corresponding closest predicted trajectory $\{\hat{x}_t, \hat{y}_t\}_{t=1}^{T_{fut}}$,as shown in Eq. (10).

$$\begin{aligned} \text{ADE} &= \frac{1}{T_{fut}} \sum_{t=1}^{T_{fut}} \sqrt{(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2}, \\ \text{FDE} &= \sqrt{(x_{-1} - \hat{x}_{-1})^2 + (y_{-1} - \hat{y}_{-1})^2}, \end{aligned} \tag{10}$$

where subscript -1 index of the last point of the trajectory.

brier-ADE and brier-FDE [63] are similar to ADE and FDE but add the probability $\hat{p}$ of the closest predicted trajectory, as shown in Eq. (11):

$$\begin{aligned} \text{brier-ADE} &= \text{ADE} + (1 - \hat{p})^2, \\ \text{brier-FDE} &= \text{FDE} + (1 - \hat{p})^2. \end{aligned} \tag{11}$$

To evaluate the multimodal prediction performance, all methods generate 20 predicted future trajectories to measure the above metrics and report the minimum one to make comparisons fairly.

Implementation. We cluster the future trajectories to generate $L = 20$ sparse motion tokens with the length $T_c = 12$. The feature dimension $D = 128$ is shared on the social interaction and multimodal prediction modules. The number of attention heads $M = 8$ on all attention-related operations. We stack 2 multi-layer perceptrons (MLP) on the trajectory embedder. The number K of learnable offsets in irregular sampling interaction is set to 4. We find the nearest $S = 4$ neighbors to make the sparse interaction. We stack 2 MLPs on temporal injection. We stack 2 social-level encoders and 2 trajectory-level decoders on ETH-UCY-V1 and ETH-UCY-V2. We stack 2 social-level encoders and 3 trajectory-level decoders on SDD and NBA. The learning rate is set to 0.001 on four datasets with the AdamW optimizer. The cosine annealing is used to adjust the learning rate. The batch size is set to $128, 128$, and 256 on the ETH-UCY-V1, ETH-UCY-V2, and SDD, respectively. Due to the large number of data on the NBA dataset, the large batch size 4×256 accelerates the training process with distributed training. The epoch is set to 100 on ETH-UCY-V1, ETH-UCY-V2, and NBA, respectively. The epoch of the NBA is set at 500. All experiments are conducted on RTX 2080 Ti GPU, where the experiments on NBA use 4 GPUs and the rest of experiments use 1 GPU. We will release the related code to provide more detailed implementation.

4.2 Comparison with State-of-the-Art Methods

In this section, we compare STP with state-of-the-art multimodal pedestrian trajectory prediction methods on ETH [17] (two versions), UCY [18], SDD [19], and SportVU NBA [20].

ETH-UCY-V1. Table 1 presents the comparison results of our method with state-of-the-art methods on ADE and FDE metrics. STP reduces the average displacement error (ADE) and final displacement error (FDE) from 0.21/0.36 to 0.20/0.32 on average compared to the TUTR [21] (our conference version), demonstrating the effectiveness of the extended irregular interaction to capture global interaction with an efficient sparse structure. Furthermore, STP eliminates the accuracy gap compared to the methods with late-sparsity (marked by †). Specifically, STP achieves state-of-the-art performance in average ADE and is on par with the previous best method, *i.e.*, SocialVAE+FPC [8], in FDE. Compared to another late-sparsity method, MemoNet [7], STP shows significant superiority both on ADE and FDE. What's more, STP outperforms the recent diffusion-based method [57] and large language model (LLM) augmented method [55]. All of them

TABLE 1
Multimodal trajectory prediction comparison with state-of-the-art methods on ETH-UCY-V1 using ADE/FDE metrics. * is an augmented work by a large language model. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is underlined. The lower, the better.

Model	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
SGAN [13]	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 [24]	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 [23]	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 [44]	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 [44]	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 [64]	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 [10]	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 [11]	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† [41]	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
CAGN [12]	AAAI2022	0.41/0.65	0.13/0.23	0.32/0.54	0.21/0.38	0.16/0.33	0.25/0.43
SIT [9]	AAAI2022	0.39/0.62	0.14/0.22	0.27/0.47	0.29/0.33	0.16/0.29	0.23/0.38
SocialVAE [8]	ECCV2022	0.47/0.76	0.14/0.22	0.25/0.47	0.20/0.37	0.14/0.28	0.24/0.36
SocialVAE+FPC† [8]	ECCV2022	0.41/0.58	0.13/0.19	**0.21**/0.36	**0.17**/0.29	**0.13/0.22**	0.21/**0.32**
MemoNet† [7]	ICCV2022	0.40/0.61	**0.11**/0.17	0.24/0.43	0.18/0.32	0.14/0.24	0.21/0.35
LED [57]	CVPR2023	0.39/0.58	**0.11**/0.17	0.26/0.43	0.18/**0.26**	**0.13/0.22**	0.21/0.33
TUTR [21]	ICCV2023	0.40/0.61	**0.11**/0.18	0.23/0.42	0.18/0.34	**0.13**/0.25	0.21/0.36
LMTraj-SUP* [55]	CVPR2024	0.41/0.62	0.12/**0.16**	0.22/**0.35**	0.20/0.32	0.18/0.28	0.23/0.35
STP (Ours)	-	**0.36/0.51**	**0.11/0.16**	0.24/0.41	**0.17**/0.31	**0.13**/0.23	**0.20/0.32**

TABLE 2
Multimodal trajectory prediction comparison with state-of-the-art methods on ETH-UCY-V2 using ADE/FDE metrics. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower the better.

Model	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
STAR [52]	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 [62]	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 [43]	TPAMI2023	0.29/0.47	0.12/0.18	0.29/0.51	0.20/0.35	0.15/0.27	0.21/0.35
SICNet† [14]	ICCV2023	0.27/0.45	**0.11/0.16**	0.26/0.46	0.19/0.33	0.14/0.24	0.19/0.33
SingularTrajectory [65]	CVPR2024	0.35/0.42	0.13/0.19	**0.25/0.44**	0.19/**0.32**	0.15/0.25	0.21/0.32
STP (Ours)	-	**0.24/0.38**	**0.11/0.16**	0.26/0.45	**0.18/0.32**	**0.13/0.23**	**0.18/0.30**

demonstrate the effectiveness of our STP in generating accurate multimodal trajectory predictions.

ETH-UCY-V2. Table 2 shows the performance comparisons on ETH-UCY-V2, which has more balanced data among different subsets than ETH-UCY-V1. STP achieves the best or second-best performance on each subset and state-of-the-art performance both on average ADE and FDE. Specifically, STP improves the performance of the previous post-processing method SICNet [14] from 0.19/0.33 to 0.18/0.30 on ADE/FDE. Compared to a recent diffusion-based method [65], our method still outperforms it, reducing the ADE/FDE from 0.21/0.32 to 0.18/0.30. The remarkable experimental results on ETH-UCY-V1 and ETH-UCY-V2 demonstrate that STP is effective both in balanced and unbalanced trajectory scenarios.

Stanford Drone Dataset. We further evaluate our method on the commonly used rich-scenario dataset SDD. As shown in Table 3, STP significantly improves the accuracy performance of the conference version, TUTR [21], from 7.76/12.69 to 7.43/11.81 on ADE/FDE. Furthermore, STP achieves state-of-the-art ADE performance and the second-best FDE performance. Compared with the previous best method, SocialVAE+FPC [8] with late-sparsity, STP has a minor accuracy gap of 0.09 (11.81-11.72) on FDE, while SocialVAE+FPC has a larger performance gap of 0.67 (8.10-7.43) on ADE. In general, STP is superior to the compared methods and enables the removal of expensive late-sparsity to make efficient and accurate predictions.

SportVU NBA Movement Dataset. We further evaluate STP on a special trajectory dataset, *i.e.*, the SportVU NBA Movement Dataset, with player interaction compared to the foregoing pedestrian interaction. The experimental results are shown in Table 4, where STP significantly outperforms all previous methods without the late-sparsity. Compared to the prior best method, STP achieves the best ADE (0.05 improvement) and second-best FDE (0.03 gap) on the Rebounding and achieves state-of-the-art performance on the Scoring. In general, STP is superior to all previous methods, showcasing the effectiveness of multimodal prediction on special scenarios (ball game).

Comparison in brier-ADE/FDE. Since prior works neglect probability, we select multiple methods with different multimodal prediction strategies to make comparisons on brier-ADE/FDE. SIT [9] builds a manual tree to model multimodal future trajectories and provides probability information without late-sparsity.

TABLE 3
Multimodal trajectory prediction comparison with state-of-the-art methods on SDD using ADE/FDE metrics. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Model	Venue/Year	ADE	FDE
Sophie [24]	CVPR2019	16.27	29.38
SGAN [13]	CVPR2018	27.23	41.44
Desire [22]	CVRP2018	19.25	34.05
CF-VAE [24]	CVPR2019	12.60	22.30
SimAug [66]	ECCV2020	10.27	19.71
PECNet† [41]	ECCV2020	9.96	15.88
PCCSNet [62]	ICCV2021	8.62	16.16
SIT [9]	AAAI2022	9.13	15.42
IMP [43]	TPAMI2023	8.98	15.54
SocialVAE [8]	ECCV2022	8.88	14.81
SocialVAE+FPC† [8]	ECCV2022	8.10	**11.72**
MemoNet† [7]	CVPR2022	8.56	12.66
SICNet† [14]	ICCV2023	8.44	13.65
TUTR [21]	ICCV2023	<u>7.76</u>	12.69
STP (Ours)	-	**7.43**	<u>11.81</u>

CAGN [12] uses the Gaussian Mixture Model (GMM) to model diverse future trajectories without probability information. SocialVAE+FPC [8] first models a latent space to generate dense future trajectories and then cluster them into multimodal future trajectories but without probability information. TUTR [21] is our conference version, which predicts trajectories and corresponding probabilities. To generate the probabilities, we conduct two variants of CAGN and SocialVAE+FPC to make a comparison with our proposed method. CAGN predicts 20 Gaussian components and the learnable weights of each component are considered as the corresponding probabilities. SocialVAE+FPC predicts abundant trajectories and clusters them into a GMM, where the weights of each component are probabilities. As shown in Table 7 and Table 5, STP achieves state-of-the-art performance both on brier-ADE and brier-FDE. Specifically, STP improves the brier-ADE/brier-FDE of our conference version [21] from 0.95/1.10 to 0.86/1.01 on ETH-UCY-V1 and 8.44/13.53 to 8.33/12.90 on SDD. Furthermore, STP significantly outperforms the conducted variants on both datasets. All of that showcases the effectiveness of STP in predicting multimodal future trajectories and corresponding probabilities. In addition, we find that learning probability brings pressure to the prediction models. In our opinion, the worst performance of our STP on brier-FDE should be that the best FDE is selected with the worst probability (0), *i.e.*, $11.81 + 1 = 12.81$, referring to Table 3. However, the performance of brier-FDE (12.90) is worse than 12.81 as shown in Table 5. The similar phenomena also occurs in the compared methods.

Comparison in Prediction Speed. We compare the prediction speed with previous state-of-the-art methods in sparse and dense traffic scenarios, respectively. We set the number of pedestrians N as equal to $5, 10, 20, 40$, and 80, respectively. The larger N represents a more dense scenario. Note that the data-processing (*e.g.*, rotation and translation) is not considered to calculate the inference time. As shown in Table 6, both STP and TUTR (conference version [21]) significantly outperform the methods (MemoNet [7], SocialVAE+FPC [8]) with late-sparsity. Considering the prediction length with 4.8 seconds and 12 time steps, the predictor requires a prediction within 0.4 seconds to achieve real-time prediction. However, MemoNet and SocialVAE+FPC suffer from higher prediction delays that cost $1.2989s$ and $2.3401s$ to predict a $4.8s$ trajectory in a dense scene, respectively. In contrast, both STP and our conference version are capable of making real-time predictions easily. Specifically, STP achieves about $60\times$ speed improvement in sparse scenes and $20\times$ speed improvement in dense scenes compared to SocialVAE+FPC. Compared to our conference version, more time consumption comes from the interaction between each pair of pedestrians, while the TUTR only models the interaction between the central pedestrian and neighbor pedestrians. Nevertheless, our method eliminates the performance gap with the late-sparisty methods to dual focus both on efficient and accurate prediction. The comparisons between accuracy and speed showcase the feasibility of our proposed efficient principle, *i.e.*, leveraging the sparse structures to perform the global effects, on social interaction and multimodal prediction.

TABLE 4
Multimodal trajectory prediction comparisons with state-of-the-art methods on NBA using ADE/FDE metrics. The best performance is in bold formatting, while the second-best performance is represented with an underline. † represents the methods with late-sparsity. The lower, the better.

Method	Venue/Year	Rebounding	Scoring
Linear	-	2.14/5.09	2.07/4.81
Trajectron++ [56]	ECCV2020	0.98/1.93	0.73/1.46
BiTraP [67]	RAL2021	0.83/1.72	0.74/1.49
SGNet-ED [68]	RAL2022	0.78/1.55	0.68/1.30
SocialVAE [8]	ECCV2022	0.72/1.37	0.64/1.17
SocialVAE+FPC† [8]	ECCV2022	<u>0.66</u>/**1.10**	<u>0.58</u>/<u>0.95</u>
STP (Ours)		**0.61**/<u>1.13</u>	**0.51/0.91**

TABLE 5
Multimodal trajectory prediction comparisons on SDD using brier-ADE/brier-FDE. * represents the methods without probability prediction. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Method	Venue/Year	brier-ADE/brier-FDE
CAGN* [12]	AAAI2022	15.36/26.42
SIT [9]	AAAI2022	10.06/16.33
SocialVAE+FPC* [8]	ECCV2022	9.57/14.75
TUTR [21]	ICCV2023	<u>8.44</u>/<u>13.53</u>
STP (Ours)	-	**8.33/12.90**

TABLE 6
Comparisons on prediction time recorded by seconds. Our method significantly outperforms the compared methods.

N	MemoNet [7]	SocialVAE+FPC [8]	TUTR [21]	STP (Ours)
5	0.3221	0.6067	0.0050	0.0101
10	0.4058	0.7385	0.0046	0.0094
20	0.5358	0.9198	0.0050	0.0098
40	0.7784	1.3038	0.0052	0.0149
80	1.2989	2.3401	0.0076	0.0406

TABLE 7
Multimodal trajectory prediction comparisons on ETH-UCY-V1 using brier-ADE/brier-FDE. * represents the conducted model variant. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Method	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
CAGN* [12]	AAAI2022	1.43/1.78	1.18/1.44	1.47/2.04	1.29/1.78	1.23/1.65	1.32/1.73
SIT [9]	AAAI2022	1.29/1.49	1.03/1.14	1.38/1.82	1.08/1.23	0.99/1.13	1.15/1.36
SocialVAE+FPC* [8]	ECCV2022	1.37 / 1.61	1.02/1.09	1.12/1.31	1.07/1.20	1.04/1.17	1.12/1.27
TUTR [21]	ICCV2023	1.21/1.41	0.80/0.86	0.99/1.19	1.03/1.19	0.73/0.85	0.95/1.10
STP (Ours)	-	**1.20/1.38**	**0.52/0.61**	**0.98/1.18**	**0.98/1.15**	**0.64/0.75**	**0.86/1.01**

TABLE 8
Ablation study about sparse motion modes on ETH-UCY-V1 and Stanford Drone Dataset using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Generative Mode	0.39/0.58	0.12/0.18	0.26/0.48	0.19/0.34	0.15/0.27	0.22/0.37	8.05/12.92
Prior Generative Mode	0.50/0.82	0.15/0.24	0.41/0.84	0.23/0.41	0.18/0.32	0.29/0.52	10.88/18.68
Sparse Motion Mode (Ours)	**0.36/0.51**	**0.11/0.16**	0.24/**0.41**	**0.17/0.31**	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**
Learnable Embedding	0.36/0.53	0.12/0.18	**0.23/0.41**	**0.17/0.31**	**0.13/0.23**	0.20/0.33	7.54/12.22
Goal Point	**0.34/0.51**	**0.11**/0.18	**0.23/0.41**	**0.17/0.30**	**0.13/0.23**	**0.19/0.32**	7.71/12.17
Mean Point	0.35/**0.51**	**0.11**/0.17	0.24/0.42	**0.17**/0.31	**0.13/0.23**	0.20/**0.32**	7.54/11.98
Full Trajectory (Ours)	0.36/**0.51**	**0.11/0.16**	0.24/**0.41**	**0.17**/0.31	**0.13/0.23**	0.20/**0.32**	**7.43/11.81**

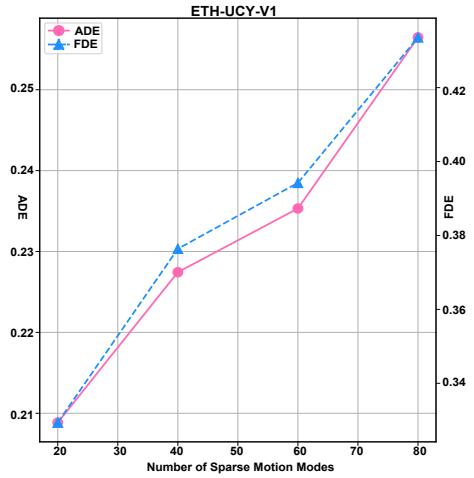


Fig. 5. Comparisons across different number of sparse motion modes on ETH-UCY-V1 using ADE/FDE metrics. The lower, the better.

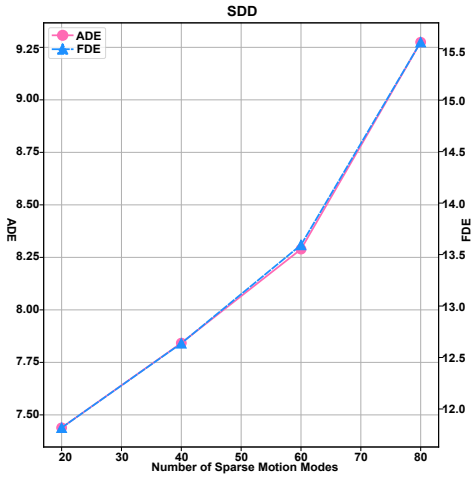


Fig. 6. Comparisons across different number of sparse motion modes on SDD using ADE/FDE metrics. The lower, the better.

4.3 Ablation Study

We conduct a series of ablative experiments to evaluate the performance contribution of the proposed early-sparsity and irregular interaction, where each component is replaced by the corresponding counterpart or removed while keeping the others unchanged.

4.3.1 *Early-Sparsity*

The early-sparsity is proposed to achieve efficient prediction. It involves a motion compressor to generate the sparse motion modes, which are the core to predict the final multimodal future behaviors.

Here, we conduct related experiments to validate the effectiveness of our sparse motion modes.

Comparison with prior Dense Motion Modes. We investigate the effectiveness of our generated sparse motion modes by making comparisons with previously used dense motion modes [8], [41], [52], which are sampled repeatedly from a latent space to represent the multimodal motion behaviors. We sample 20 latent variables from a normal Gaussian distribution (16 dimensions) to replace our sparse motion modes. Specifically, two variants are used to optimize these latent variables, *i.e.*, the Generative Mode and the Prior Generative Mode. The former uses the variety loss [13] to train the prediction model, while the latter [8], [41] uses variational strategy to generate multimodal results with different training and inference stages. In the training stage, the future trajectory is encoded into a latent variable as the center motion mode, combined

TABLE 9
Comparisons between the irregular interaction and global/local interaction on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Global Int	0.39/0.58	0.12/0.18	0.24/0.41	0.19/0.35	0.14/0.25	0.21/0.35	7.58/11.96
Global Int w Memory	0.37/0.52	0.12/0.19	**0.23**/0.41	**0.17/0.30**	0.15/0.26	**0.20**/0.33	7.62/12.06
Marginal Local Int	0.41/0.60	0.13/0.20	**0.23/0.40**	0.18/0.34	0.14/0.25	0.21/0.35	7.51/11.86
Joint Local Int w Memory	0.37/0.52	0.14/0.20	**0.23**/0.41	**0.17/0.30**	0.14/0.24	0.21/0.33	7.71/12.20
Irregular Int (Ours)	**0.36/0.51**	**0.11/0.16**	0.24/0.41	**0.17**/0.31	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**

TABLE 10
Ablation study about different irregular interactions on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Irregular Grid Int	0.37/0.53	**0.11**/0.18	**0.24/0.40**	0.18/0.32	0.14/**0.23**	**0.20**/0.33	7.72/12.31
Irregular Sampling Int	**0.36/0.51**	**0.11/0.16**	**0.24**/0.41	**0.17/0.31**	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**

with the sampled 19 latent variables to predict multimodal future trajectories. The center motion mode is enforced to follow a normal Gaussian distribution by the KL divergence, and the predicted trajectory from the center motion mode is used to optimize the prediction model. In the inference stage, we sample 20 latent variables to generate the prediction results. As shown in the upper block of Table 8, we observe that: 1) our method significantly outperforms the conducted two variants; 2) the prediction loss is more effective than the prior loss to improve accuracy. The reason for the first observation could be that the prediction from latent variables to future behaviors is a dense to sparse process, where similar samples lead to repetitive predictions and thus suffer from higher error. Thus, prior works [8], [14] use the late-sparsity to retrench the repetitive predictions and therefore improve accuracy but suffer from expensive prediction time. For the second observation, the Prior Generative Mode constrains the predicted trajectory generated from a prior latent variable by the prior loss, thus deviating from diversity. while the Generative Mode magnifies the differences of the sampled latent variables by the prediction loss to improve the diversity of predictions. Nonetheless, the random sampled latent variables within a certain number of sampling times can not represent the global distribution, thus impeding further accurate prediction. In contrast, our sparse motion modes compressed from world knowledge are capable of covering the global motion behaviors to pursue a more accurate prediction.

Comparison with different Sparse Motion Modes. The sparse motion modes represent the global motion behaviors. Here, we conduct multiple strategies to obtain different sparse motion modes. As shown in the lower block of Table 8, the Latent Embedding, referring to the object query in DETR [69], uses 20 learnable embeddings to build the sparse motion modes. It can be considered as a latent compression by the neural network. Referring to [41] and [43], the Goal Point clusters the last point of aligned training trajectories to generate sparse motion modes. In contrast, the Mean Point clusters the mean value of aligned training future trajectories into sparse motion modes. Note that neither [41] nor [43] cluster the goal points or mean value, just predicting them. The experimental results show that both the Latent Embedding, Goal Point, and Mean Value show apparent performance discrepancy on ETH-UCY-V1 and SDD datasets. Specifically, the Goal Point shows the best ADE on ETH-UCY-V1 while undergoing the worst ADE on SDD. Our method (Full Trajectory) reaches a balanced performance, achieving the best FDE on ETH-UCY-V1 and significant state-of-the-art ADE/FDE on SDD. It shows the well generalization of the full trajectory on various scenarios.

Impact of the Number of the Sparse Motion Modes. Multi-modal prediction in pedestrian trajectory prediction measures the prediction performance with a fixed number of motion modes. Here. we analyze the impact of the number of sparse motion modes. We cluster $20, 40, 60$ and 80 sparse motion modes to represent the diverse motion behaviors. To select the best 20 trajectories, we predict corresponding confidence when the number of sparse motion modes exceeds the required number of 20. As shown in Figure 5 and Figure 6, 20 motion modes achieve the best performance both on ETH-UCY-V1 and SDD datasets. The possible reason could be that the 20 motion modes are enough to cover the various motion behaviors of pedestrians. In addition, the classification task, *i.e.*, selecting the predicted trajectories with top-20 confidences is still challenging.

4.3.2 *Irregular Interaction*

Our STP proposes the irregular interaction to perform global interaction with efficient sparse interaction. We conduct ablated experiments against the commonly used global interaction and local interaction to evaluate its effectiveness. Additionally, we conduct detailed quantitative experiments to measure the impact of the hyper-parameters. Finally, we make detailed comparisons to assess the effectiveness of different irregular interactions. We default to using irregular sampling interaction for comparisons due to its superior performance.

Comparison with Global Interaction. Two strategies are used to make comparisons with global interaction. As shown in the first block of Table 9, the Global Int uses the multi-head self-attention mechanism [51] among all neighbors to model the social interactions, which are combined with the sparse motion modes to generate multimodal future trajectories, referring to the [52]. Global interaction models social interaction using the multi-head self-attention mechanism [51] across all neighbors. The Global Int w Memory adds extra memory attention behind the global interaction module to align with our method. The hidden dimension, number of

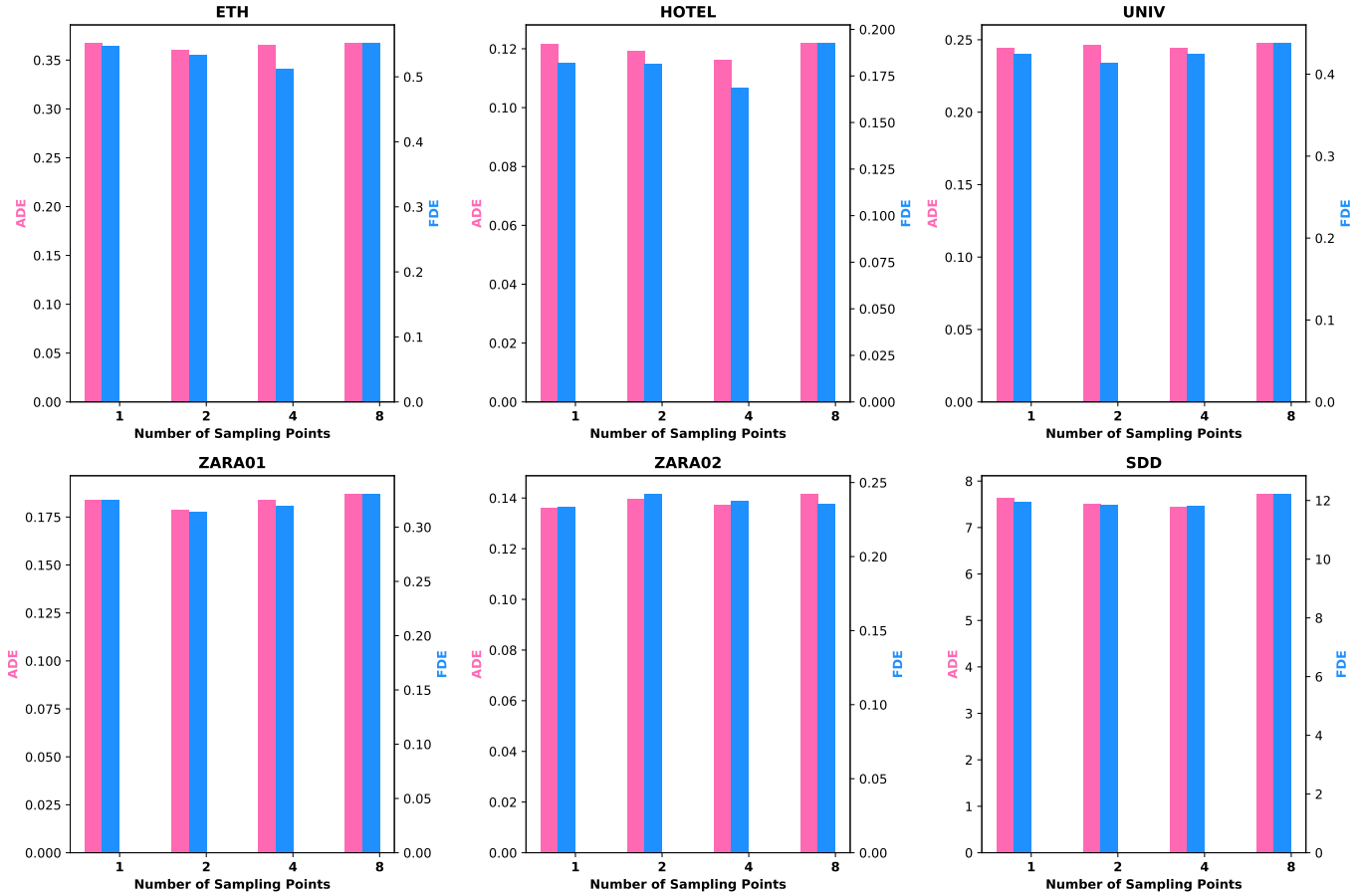


Fig. 7. Comparisons across different number of interactive positions on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

heads, and number of layers are identical to our irregular interaction. Our method achieves the best performance on both ETH-UCY-V1 and SDD datasets. We observe that the global interaction with memory attention introduces a performance conflict, reducing the error on ETH-UCY-V1 but increasing the error on SDD. The reason could be that the memory module further models the interaction, bringing a double-edged sword effect on various interactive scenarios. In contrast, our irregular interaction models global interaction with a sparse style, which is generalizable and performs better on both datasets.

Comparison with Local Interaction. We further compare the prior local interaction, which models the interaction within the radius r. Referring to the SocialVAE [8], the r is set to 2 and 5 on the ETH-UCY-V1 and SDD datasets, respectively. As shown in the second block of Table 9, the Marginal Local Int only models the local interaction between the center object and its neighbor objects within the radius r, while the Joint Local Int w Memory first builds joint local interaction, where each object interacts with its neighbors within the same radius r. Then, the memory attention between the sparse motion modes and encoded interactive features is added behind the joint local interaction module. As shown in the second block of Table 9, our method achieves the lowest prediction error (ADE/FDE) both on ETH-UCY-V1 and SDD datasets. Similar to the global interaction, the memory modules also bring the performance conflict between two datasets. We further notice that the ETH-UCY-V1 is suitable for global interaction, while the SDD is suitable for local interaction. In contrast, our method can generalize into various interactive scenarios, significantly showcasing the superiority of our proposed irregular interaction.

Comparisons with Different Irregular Interactions. Here, we conduct ablated studies to compare two proposed irregular interactions. Specifically, the Irregular Gird Interaction (IGI) is one of our proposed methods to build irregular interaction using standard deformable attention mechanism [60] on an irregular grid. The Irregular Sampling Interaction (ISI) is the other proposed method to achieve efficient irregular interaction by adaptive sampling. As shown in Table 10, the ISI performs best on ETH-UCY-V1 and SDD datasets. The reason could be that IGI suffers from many empty interactions because many sampling locations in the built irregular grid are empty, as shown in Figure 4. In contrast, all sampling points in the ISI are valid to make a specific sparse interaction, leading to superior performance.

Impact of the number of Interactive Positions. The learnable interactive positions are the core to modeling global interaction with a sparse style for our irregular interaction. Here, we conduct experiments to analyze the impact of the number (K) of interactive positions. Specifically, we set K to $1, 2, 4, 8$, respectively. As illustrated in Figure 7, our irregular interaction shows lower sensitivity for different numbers of interactive positions on each dataset. The reason could be that the learnable interactive positions in our irregular interaction are adaptive. It can capture various sparse interactions at different layers despite the less interactive positions, such as $K = 1$. Observing the performance change,

we find that the $K = 1, 8$ are slightly worse than the $K = 2, 4$. We speculate that the more interactive positions lead to over-interaction for the scenarios with a small number of objects, while the less interactive positions result in under-interaction for the scenarios with many objects. $K = 4$ is a balanced value in different scenarios.

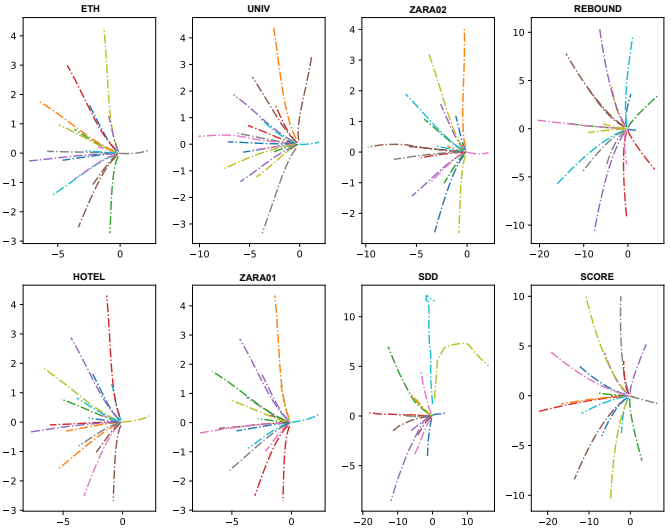


Fig. 8. Visualizations of the sparse motion modes generated from our proposed world compression. The motion direction is from right to left.

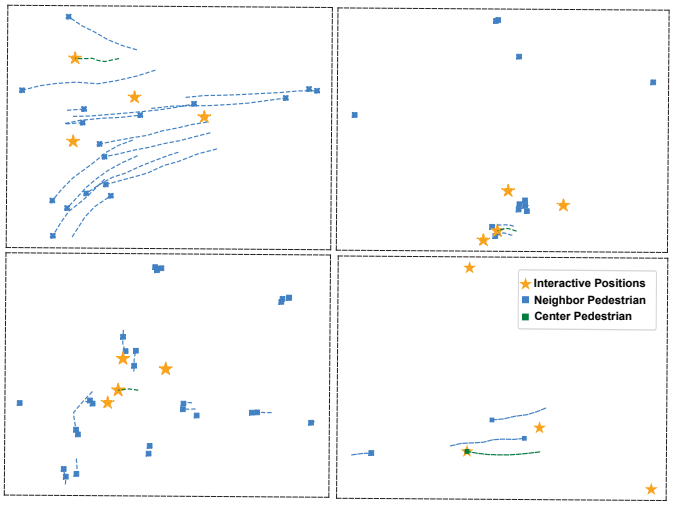


Fig. 9. Visualizations of the learnable interactive positions generated from our proposed irregular interaction. Best view in color.

4.4 Visualization Results

We provide the visualized results of STP on sparse motion modes, irregular interaction, and predicted multimodal future trajectories.

Sparse Motion Modes. Our efficient and accurate prediction comes from the generated sparse motion modes compressed from the world motion behaviors. We provide an intuitive visualization of the generated sparse motion modes to evaluate their ability to represent diverse motion behaviors. As shown in Figure 8, the generated sparse motion modes are capable of covering various motion behaviors, such as going straight, turning left/right with different angles, or turning back.

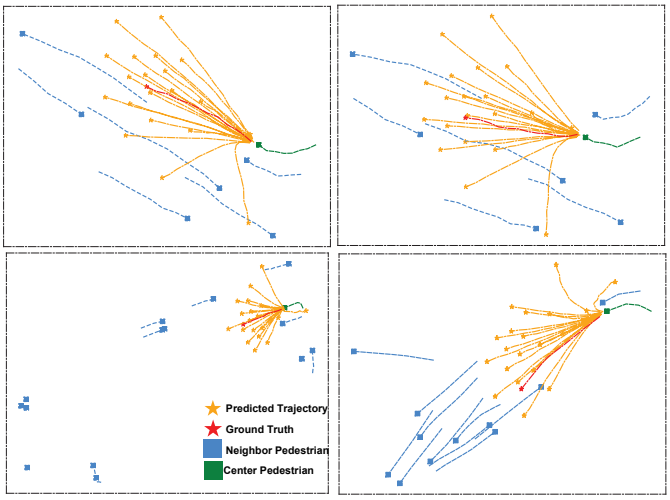


Fig. 10. Visualizations of the predicted multimodal future trajectories. Best view in color.

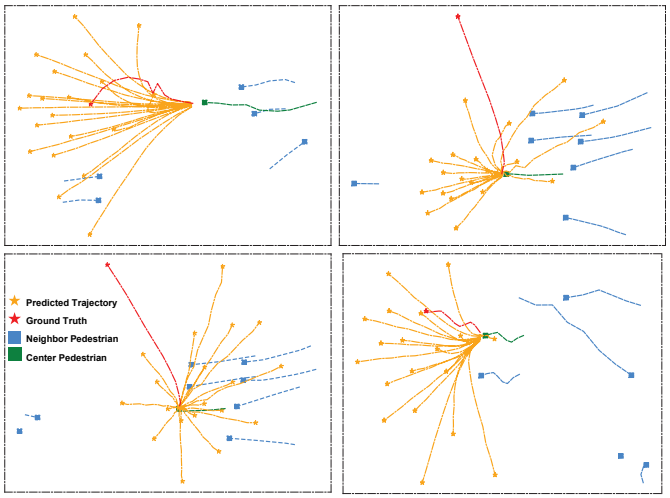


Fig. 11. Visualizations of the failure cases of the predicted multimodal future trajectories. Best view in color.

Irregular Interaction. Our irregular interaction builds various sparse interactions at learnable interactive positions to perform global interaction. We select some representative interactive scenarios from each of the subsets on ETH-UCY-V1 and SDD to visualize the distribution of the learned interactive positions. We visualize the learned interactive positions of the center pedestrian at the second interaction layer to make a consistency between ETH-UCY-V1 and SDD. As illustrated in Figure 9, our irregular interactions enable the model to capture various sparse interactions at different interactive positions. For the left two dense scenarios, the irregular interaction in the upper scenario captures the sparse interaction at the far positions compared to the irregular interaction in the lower scenario. For the right more sparse scenarios, the irregular interaction also captures the sparse interaction with various distances. It shows the adaptability of irregular interaction in various scenarios.

Predicted Trajectory. We visualize the predicted multimodal future trajectories to show the superiority in multimodal trajectory prediction. As shown in Figure 10, the visualized results show the STP is capable of covering the true motion intentions. Specifically, the upper two figures show the intention of turning right, while the

lower two figures show the intention of turning right. Furthermore, the predicted multimodal future trajectories show well diversity in covering various motion behaviors, such a going straight, turning left/right, avoiding collision, and walking with the dense crowd.

Failure Cases. The failure cases of our method are visualized in Figure 11, which is mainly reflected in the sharp turning and zigzag motion trajectory. The reason could be the trajectory is single to capture the complex motion intention. Ego-information, such as pose and face, is important for understanding detailed motion behaviors.

5 Conclusion

This paper devotes to build an efficient and accurate pedestrian trajectory prediction model. To achieve that, we present an efficient principle, *i.e.*, leveraging the sparse structure to perform global effects, to achieve both high accuracy and real-time speed. To this end, a sparse trajectory prediction model, termed STP, is developed to instantiate this efficient principle in the foundational social interaction module and multimodal prediction module of pedestrian trajectory prediction. For the social interaction module, an irregular interaction is proposed to perform global interaction with an efficient sparse interaction. Compared to the conflict between global and local interaction in various scenarios, the irregular interaction showcases the well generalization in various scenarios. For the multimodal prediction module, an early-sparsity strategy is proposed to generate the sparse motion modes before the model training and inference, which enables covering the global motion behaviors and avoiding the frequent late-sparsity to improve the prediction speed. Furthermore, the sparse motion modes show the effective ability to predict multimodal future trajectories. We evaluate STP on four commonly used datasets, and the experimental results demonstrate that STP maximizes both accuracy and prediction speed, achieving state-of-the-art performance and significantly improving inference speed by about $20\times$ - $60\times$ to satisfy the real-time demand.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2021YFB1714700, in part by NSFC under Grants 62088102, 62106192, and 12326608, in part by the Natural Science Foundation of Shaanxi Province under Grant 2022JC-41, and in part by Fundamental Research Funds for the Central Universities under Grant XTR042021005.

References

[1] F. Leon and M. Gavrilescu, "A review of tracking, prediction and decision making methods for autonomous driving," *arXiv preprint arXiv:1909.07707*, 2019.

[2] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Trans. Intell. Trans. Syst.*, vol. 23, no. 1, pp. 33–47, 2020.

[3] 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.

[4] 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.

[5] 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.

[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] C. Xu, W. Mao, W. Zhang, and S. Chen, "Remember intentions: Retrospective-memory-based trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2022, pp. 6488–6497.

[8] P. Xu, J.-B. Hayet, and I. Karamouzas, "Socialvae: Human trajectory prediction using timewise latents," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 511–528.

[9] L. Shi, L. Wang, C. Long, S. Zhou, F. Zheng, N. Zheng, and G. Hua, "Social interpretable tree for pedestrian trajectory prediction," in *Proc. AAAI. Conf. Artif. Intell.*, vol. 36, no. 2, 2022, pp. 2235–2243.

[10] 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.

[11] 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.

[12] J. Duan, L. Wang, C. Long, S. Zhou, F. Zheng, L. Shi, and G. Hua, "Complementary attention gated network for pedestrian trajectory prediction," in *Proc. AAAI. Conf. Artif. Intell.*, vol. 36, no. 1, 2022, pp. 542–550.

[13] 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.

[14] Y. Dong, L. Wang, S. Zhou, and G. Hua, "Sparse instance conditioned multimodal trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 9763–9772.

[15] 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.

[16] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent.*, 2020.

[17] 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.

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

[19] 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.

[20] Y. Yue, P. Lucey, P. Carr, A. Bialkowski, and I. Matthews, "Learning fine-grained spatial models for dynamic sports play prediction," in *Proc. IEEE Int. Conf. Data Mining*. IEEE, 2014, pp. 670–679.

[21] L. Shi, L. Wang, S. Zhou, and G. Hua, "Trajectory unified transformer for pedestrian trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 9675–9684.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

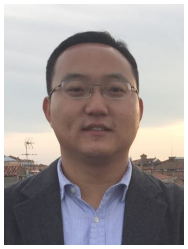
[26] Z. Cao, H. Gao, K. Mangalam, Q.-Z. Cai, M. Vo, and J. Malik, "Long-term human motion prediction with scene context," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 387–404.

[27] 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. 15 233–15 242.

[28] N. Shafiee, T. Padir, and E. Elhamifar, "Introvert: Human trajectory prediction via conditional 3d attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16 815–16 825.

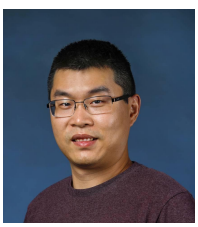
[29] A. Rasouli, M. Rohani, and J. Luo, "Bifold and semantic reasoning for pedestrian behavior prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15 600–15 610.
[30] J. Yue, D. Manocha, and H. Wang, "Human trajectory prediction via neural social physics," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 376–394.
[31] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 7, no. 2, pp. 2716–2723, 2022.
[32] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, p. 4282, 1995.
[33] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research: The 14th International Symposium ISRR*. Springer, 2011, pp. 3–19.
[34] I. Karamouzas, B. Skinner, and S. J. Guy, "Universal power law governing pedestrian interactions," *Physical Review Letters*, vol. 113, no. 23, p. 238701, 2014.
[35] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, 2007.
[36] K. Kim, D. Lee, and I. Essa, "Gaussian process regression flow for analysis of motion trajectories," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2011, pp. 1164–1171.
[37] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 961–971.
[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comp.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[39] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2342–2350.
[40] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2067–2075.
[41] 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.
[42] 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.
[43] L. Shi, L. Wang, C. Long, S. Zhou, W. Tang, N. Zheng, and G. Hua, "Representing multimodal behaviors with mean location for pedestrian trajectory prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 11 184–11 202, 2023.
[44] 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.
[45] 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.
[46] 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.
[47] 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.
[48] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. on neural networks*, vol. 20, no. 1, pp. 61–80, 2008.
[49] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2016.
[50] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018.
[51] 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.
[52] 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.
[53] 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.
[54] 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.
[55] I. Bae, J. Lee, and H.-G. Jeon, "Can language beat numerical regression? language-based multimodal trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.
[56] 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.
[57] W. Mao, C. Xu, Q. Zhu, S. Chen, and Y. Wang, "Leapfrog diffusion model for stochastic trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 5517–5526.
[58] S. Shi, L. Jiang, D. Dai, and B. Schiele, "Motion transformer with global intention localization and local movement refinement," *Proc. Adv. Neural Inf. Process. Syst.*, 2022.
[59] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.
[60] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 764–773.
[61] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[62] 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.
[63] M. Wang, X. Zhu, C. Yu, W. Li, Y. Ma, R. Jin, X. Ren, D. Ren, M. Wang, and W. Yang, "GANet: Goal area network for motion forecasting," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2023, pp. 1609–1615.
[64] 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.
[65] I. Bae, Y.-J. Park, and H.-G. Jeon, "Singulartrajectory: Universal trajectory predictor using diffusion model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.
[66] 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.
[67] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, "BiTraP: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 6, no. 2, pp. 1463–1470, 2021.
[68] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 7, no. 2, pp. 2716–2723, 2022.
[69] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

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. From 2023 to 2024, he is a visiting scholar with University of Illinois at Chicago, USA. He is currently a Ph.D. student with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University. His research interests include automatic driving and multimodal video understanding.



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 is 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 PR, MVA, and PRL.

Sanping Zhou (Member, IEEE) received his Ph.D. degree from Xi'an Jiaotong University, Xi'an, China, in 2020. From 2018 to 2019, he was a Visiting Ph.D. Student with Robotics Institute, Carnegie Mellon University. He is currently an Associate Professor with the Institute of Artificial Intelligence and Robotics at Xi'an Jiaotong University. His research interests include machine learning, computer vision and embodied intelligence, with a focus on meta learning, multi-task learning, object detection, multi-target tracking, trajectory prediction, medical image segmentation, visual navigation and visual grasping.



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.

Gang Hua (Fellow, IEEE) received the B.S. and M.S. degrees in Automatic Control Engineering from Xi'an Jiaotong University (XJTU), Xi'an, China, in 1999 and 2002, respectively. He received the Ph.D. degree in Electrical Engineering and Computer Science at Northwestern University, Evanston, Illinois, USA, in 2006. He is currently the Vice President of the Multimodal Experiences Research Lab at Dolby Laboratories. His research focuses on computer vision, pattern recognition, machine learning, robotics, towards general Artificial Intelligence, with primary applications in cloud and edge intelligence. Before that, he was the CTO of Convenience Bee, and the Managing Director and Chief Scientist of its research branch in US, Wormpex AI Research (2018-2024). He also 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 Senior 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 a 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 Senior Scientist (2006-09) at Microsoft Live labs Research. He is an associate editor of TPAMI and MVA. He is a general chair of ICCV'2027 and a program chair of CVPR'2019&2022. He is the author of more than 200 peer reviewed publications in prestigious international journals and conferences. He holds 35 US patents and has 15 more US patents pending. He is the recipient of the 2015 IAPR Young Biometrics Investigator Award. He is an IEEE Fellow, an IAPR Fellow, and an ACM Distinguished Scientist.

Summary of Changes from the Conference Version

Le Wang
Institute of Artificial Intelligence and Robotics,
Xi'an Jiaotong University
lewang@mail.xjtu.edu.cn

October 19, 2024

This manuscript extends our ICCV'2023 conference paper "TUTR: Trajectory Unified Transformer for Pedestrian Trajectory Prediction" to achieve efficient and accurate prediction in two fundamental modules of pedestrian trajectory prediction, i.e., social interaction and multimodal prediction. The conference version employs the early-sparsity strategy to improve the prediction speed at the multimodal prediction modules. In complementary, this version proposes an irregular interaction to reduce the computational complexity of social interaction modules. The detailed new major contributions are presented in four major aspects.

1. We introduce a novel Sparse Trajectory Prediction model (STP) to achieve accurate, real-time performance, which unifies the proposed early-sparsity in the conference version and the irregular interaction in this version into a transformer-style encoder-decoder framework by following a new efficient principle that leverages the sparse structures to achieve global effects.

2. We propose a novel irregular interaction mechanism designed to perform global interaction in a sparse yet effective manner, thus reducing the complexity of social interaction.

3. We present more clearer motivation and technical details.

4. We conduct more experiments on two extra datasets to evaluate the effectiveness of the proposed method.

Specifically, the detailed differences between this manuscript and our conference version are as follows:

1. **In Section 1 - Introduction**: First, the goal of the conference version and this version are unified to pursue the dual focus on efficiency and accuracy. Second, we

have added an efficient principle to guide the efficient and accurate prediction in two fundamental modules, *i.e.*, social interaction module, and multimodal prediction module. Third, we have added the motivation of the proposed irregular interaction to reduce the computational complexity in the social interaction module. Finally, we have revised the motivation of the conference version into modeling the early-sparsity in multimodal prediction module to improve prediction speed caused by previous late-sparsity. We have highlighted the differences between this manuscript and the conference version.

2. **In Section 2 - Related Work**: We have added a brief review of the latest related works in pedestrian trajectory prediction. We have highlighted the differences between this manuscript and the conference version.

3. **In Section 3 - Proposed Method**

   (a) In Section 3.4 - Social Interaction, we have introduced an effective irregular interaction, which proposes two strategies to perform global interaction in a sparse manner, thus reducing computational complexity.

   (b) In Section 3.4.2 - Irregular Grid Interaction, we have added the first proposed strategy, *i.e.*, irregular grid interaction, which builds an irregular grid and thus enables the deformable attention to extract irregular interaction without the Euclidean constraint.

   (c) In Section 3.4.2 - Irregular Sampling Interaction, we have introduced the second proposed irregular sampling interaction, which further breaks the grid constraint to directly model the sparse interaction between the interaction points and their surrounding neighbors.

   (d) In Section 3.4.2 - Computational Complexity, we have added the comparisons of computational complexity between our proposed two types of irregular interaction and the previously commonly used global interaction based on the self-attention mechanism.

4. **In Section 4 - Experiments and Discussions**:

   (a) In Section 4.1 - Experimental Setting, we have added two new commonly used datasets, *i.e.*, ETH-UCY-V2 dataset and the SportVU NBA movement dataset.

   (b) In Section 4.2 - Comparison with the State-of-the-Art Methods, we have included more recent methods (Table 1 and Table 3) for comparison and conducted additional experiments on the newly added datasets, *i.e.*, SDD (Table 2) and NBA (Table 4).

(c) In Section 4.3.1 - Early Sparsity, we have added the ablated study *i.e.*, "Comparison with prior Dense Motion modes", to discuss the superiority of our sparse motion modes and the inferior of previous dense motion modes (Table 8).

(d) In Section 4.3.2 - Irregular Interaction, we have added the ablation studies to evaluate the effectiveness of our proposed irregular interaction against the commonly used global interaction and local interaction (Table 9).

(e) In Section 4.4 - Visualization Results, we have added the visualization of the proposed irregular interaction (Figure 9) and the failure cases of our predicted trajectories (Figure 11).

Apart from these, many other minor changes in texts are scattered around the paper to support the method and make the paper more understandable and complete.

Last but not least, we highlight every part of the text that is significantly different from the conference version in blue, which is included as a summary of changes.

Sparse Trajectory Prediction

Liushuai Shi, *Student Member, IEEE,* Le Wang, *Senior Member, IEEE,* Sanping Zhou, *Member, IEEE,* Wei Tang, *Member, IEEE,* and Gang Hua, *Fellow, IEEE*

Abstract—Pedestrian trajectory prediction is crucial for ensuring safe decision-making in intelligent robotic systems. While this task demands real-time performance, previous works have primarily focused on improving prediction accuracy, often neglecting efficiency. Dense predictions with time-consuming post-clustering steps and global interactions with quadratic computational complexity result in a trade-off between accuracy and speed. In this paper, we propose a novel Sparse Trajectory Prediction (STP) model that aims to achieve both high accuracy and real-time speed by following an efficient principle: leveraging sparse structures to achieve global effects. STP instantiates this principle within a transformer-style encoder-decoder framework. In the encoder, STP introduces irregular interaction, which builds sparse interactions with dynamic interactive positions, reducing computational complexity from quadratic to linear while maintaining global interaction. In the decoder, STP applies an early-sparsity strategy to generate sparse motion modes that represent global motion behaviors. These modes are shared across all predictions, eliminating redundant computations. By harnessing the expressive power of transformers, STP maps these sparse motion modes into multimodal future trajectories, significantly improving prediction speed while ensuring accuracy. Experimental results on four commonly used datasets demonstrate that STP maximizes both accuracy and prediction speed, achieving state-of-the-art performance and significantly improving prediction speed by about $20\times$ - $60\times$ to satisfy the real-time demand.

Index Terms—Pedestrian Trajectory Prediction, Sparse Interaction, Transformer

1 Introduction

Pedestrian trajectory prediction is critical for ensuring safe and accurate decision-making in intelligent robotics. It serves as a bridge between the perception module upstream and the planning module downstream [1], [2] in various intelligent applications, such as autonomous vehicles [3], surveillance systems [4], [5], and other motion prediction tasks [6]. However, trajectory prediction is extremely challenging due to the intricate motion multimodality present in complex, interactive environments. It requires the predictor to extract social interaction features to forecast diverse motion behaviors represented by multiple socially acceptable future trajectories.

As a real-time demanding task, both accuracy and efficiency in trajectory prediction are crucial for making safe decisions in dynamic and unpredictable traffic scenarios. However, current research prioritizes accuracy at the expense of efficiency, creating a significant bottleneck for real-world deployment. The challenge arises from the multimodal nature of future behaviors, which has led to various methods that explore different mode spaces to capture diverse motion patterns, such as explicit Gaussian spaces [10], [11], [12], latent spaces [8], [13], and memory spaces [7], [14]. The multiple motion modes are sampled repeatedly from the generated space to account for the multimodal motion behaviors, but they suffer from inaccurate prediction due to similar sampling motion modes. To pursue higher accuracy, recent works [7], [8], [14] adopt dense prediction with late-sparsity strategies, where a large number of potential trajectories are generated first. A clustering algorithm (e.g., K-means) is then applied to eliminate redundant predictions, refining the output to the desired set. While this approach improves accuracy, it introduces significant inefficiencies, as generating and clustering a large volume of predictions becomes computationally expensive. As shown in Figure 1, these methods (marked by '+') create a trade-off between accuracy and computational speed, making them ill-suited for real-time applications. Furthermore, as the number of pedestrians increases, efficiency deteriorates even more, moving further away from real-time feasibility, as

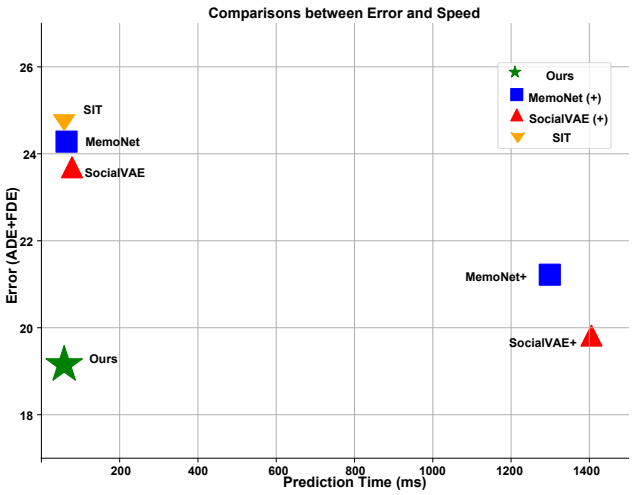


Fig. 1. The comparisons against the methods (MemoNet [7],SocialVAE [8]) with the late-sparsity (marked by '+') and the SIT [9] without late-sparsity. The late-sparsity leads to an obvious trade-off between accuracy and speed.

- *Liushuai Shi, Le Wang and Sanping Zhou are with the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and 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.)*
- *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 the Multimodal Experiences Lab, Dolby Laboratory, Bellevue, WA 98004, USA. E-mail: ganghua@gmail.com.*
- *Part of this work was done while Liushuai Shi was a visiting scholar at University of Illinois Chicago.*

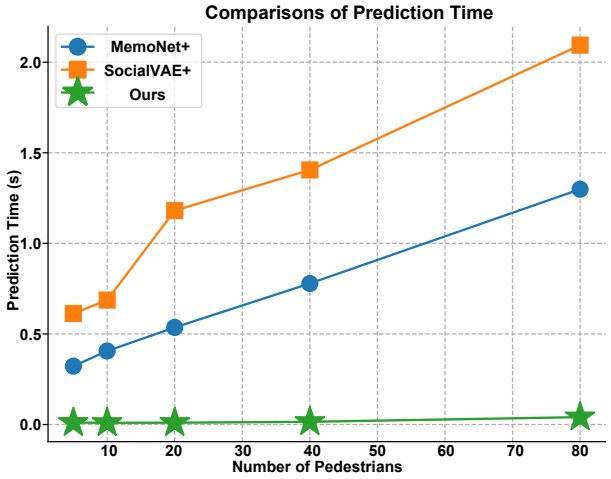


Fig. 2. The prediction time (seconds) changing with different number of pedestrians. All methods predict the trajectory with 12 time steps (4.2 seconds).

illustrated in Figure 2. More recently, interaction modeling has evolved from local interactions [15] to more comprehensive global interactions [8], [9], [11], [13], which introduce richer social behavior modeling but at the cost of quadratic computational complexity with respect to the number of neighbors. This further exacerbates the efficiency problem.

This paper introduces a novel framework, termed the Sparse Trajectory Prediction (STP), to strike a dual focus on accuracy and efficiency, delivering precise, real-time predictions, as demonstrated in Figure 1. At the heart of STP is a new efficiency principle: leveraging sparse structures to achieve global effects. STP applies this principle across two key modules of pedestrian trajectory prediction: social interaction and multimodal prediction, which are integrated within a unified transformer-based architecture.

Efficient Social Interaction. STP reduces the computational complexity of global interactions by exploiting the inherent sparsity in social interactions: a pedestrian mostly interacts with only a small subset of neighbors. A simple approach to building this sparse interaction is to fix the interaction distance [8]. However, this approach fails to model more challenging interactions (e.g., long-range interactions) and is difficult to generalize across various scenarios due to its reliance on the distance hyperparameter. This conflict between global and local interactions drives us to ask: Can we model global interaction in a sparse manner?

To address this, STP introduces an irregular interaction mechanism that learns multiple arbitrary interaction points, where the pedestrian engages with a sparse set of neighbors at each point individually, thereby improving computational efficiency. These sparse interactions are then fused around each point to achieve global interaction. Standard deformable attention [16] struggles in non-Euclidean trajectory spaces due to the irregularity of social fields. Therefore, STP employs an irregular sampling strategy that removes the constraints of Euclidean space. This approach enables STP to perform global interaction efficiently, benefiting from the reduced computational cost of sparse interactions.

Multimodal Prediction with Early-Sparsity. STP introduces an early-sparsity strategy for multimodal prediction. It treats late-sparsity as a dense-to-sparse process that extracts diverse results from dense candidates. However, this dense-sparse operation must be performed frequently for each prediction, leading to inefficiency. In contrast, our early-sparsity strategy is designed to represent global motion behaviors with a one-time dense-to-sparse process.

Specifically, early-sparsity involves a motion compressor that eliminates the need for repeated late-sparsity operations. The motion compressor condenses world knowledge from the training dataset into diverse, sparse motion modes independent of model training and inference. These motion modes are shared across all predictions, capturing global motion behaviors efficiently. Leveraging the expressive power of Transformers, these sparse motion modes are mapped into multimodal future trajectories for specific scenarios, significantly improving prediction speed while maintaining accuracy.

Unified Transformer Architecture. STP seamlessly integrates the irregular interaction and early-sparsity strategies in a unified transformer-style encoder-decoder architecture. The framework consists of three two components: *(1) a Motion Compressor*, which employs the early-sparsity to obtain the sparse motion modes by compressing the world knowledge, *(2) a Social-level Encoder*, which incorporates irregular interactions to reduce computational complexity by focusing on sparse yet effective social interactions, and *(3) a Trajectory-level Decoder*, which analyzes the relationships between the sparse motion modes to enhance diversity, and perceives encoded social interactions with the sparse motion modes to produce accurate, efficient multimodal predictions.

We evaluate STP on four commonly used datasets, *i.e.*, ETH-UCY-V1 [17], [18], ETH-UCY-V2 [17], [18], Stanford Drones Dataset (SDD) [19] and the SportVU NBA movement dataset [20]. The experimental results demonstrate the effectiveness of our proposed efficiency principle in achieving the dual focus on both accuracy and speed. Specifically, STP outperforms the state-of-the-art methods in terms of accuracy, including the late-sparsity methods. Additionally, STP offers superior efficiency, significantly improving prediction speed by about $20\times$ - $60\times$ against previous well-tuned state-of-the-art methods that rely on late-sparsity.

The contributions of this paper are summarized below.

- Both efficiency and accuracy in pedestrian trajectory prediction are crucial to downstream applications, but current research prioritizes accuracy at the expense of efficiency, creating a significant bottleneck for real-world deployment. We introduce a novel Sparse Trajectory Prediction model (STP) to achieve accurate, real-time performance by following a new efficient principle that leverages the sparse structures to achieve global effects.
- To reduce the complexity of social interaction, we propose a novel irregular interaction mechanism designed to perform global interaction in a sparse yet effective manner.
- Toward efficient multi-modal prediction, we propose a novel early-sparsity strategy to generate sparse motion modes, which are shared in all predictions to represent global motion behaviors following the proposed efficient principle.
- We integrate the irregular interaction and early-sparsity strategies in a unified Transformer architecture. It maps the sparse motion modes into specific multimodal future trajectories, effectively improving the prediction speed.
- Extensive experiments on four benchmarks demonstrate the efficiency and accuracy of our proposed method against existing state-of-the-art methods.

This paper extends our previous conference paper [21] to

achieve efficient and accurate prediction in two fundamental modules of pedestrian trajectory prediction, *i.e*, social interaction and multimodal prediction. The conference version employs the early-sparsity strategy to improve the prediction speed at the multimodal prediction modules. In complementary, this version proposes an irregular interaction to reduce the computational complexity of social interaction modules. Furthermore, the proposed early-sparsity and irregular interaction are unified as the proposed efficient principle. Finally, we present clearer motivation and more technical details about the proposed method and conduct more experiments on two extra datasets to evaluate the effectiveness of the 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

Research on pedestrian trajectory prediction is briefly categorized into two classes: prediction based on environment information (*e.g.*, semantic map) [22], [23], [24], [25], [26], [27], [28], [29], [30], [31] and prediction based on social interaction from neighbors. In this paper, we focus on the latter to achieve efficient prediction.

Considering the influence factors of human motion, the pedestrian trajectory predictor extracts social interactive features in both temporal and spatial dimensions and predicts diverse future trajectories to cover multimodal motion behaviors. This section briefly reviews related work in social interaction and multimodal trajectory prediction to present the current research status.

2.1 Social Interaction Extraction

Physical Models. Before deep learning, many works design specific physical models to forecast a deterministic future trajectory. Social force [32], motion velocity [33], and energy [34] are commonly used to model the motion behavior of pedestrians. Also, some works employ the statistical model, such as Gaussian processes [35], [36], to deal with the uncertainty of future trajectories. However, they are difficult to generalize into the complex motion patterns and spatial interactions.

Deep Learning Models. As deep learning develops in the community, most deep models in pedestrian trajectory prediction extract social interaction via fashion data-driven strategy. In temporal dimension, the recurrence-based methods [8], [13], [37] use the recurrent neural networks (RNNs) [38], [39], [40] to model the sequential motion dependence. Due to the inefficient recurrent structure, researchers refit many deep models, such as Multilayer perceptrons (MLPs) [9], [21], [41], temporal convolutional networks (TCNs) [10], [42] and temporal-attention models [11], [43] to capture the temporal features from observed trajectory.

In the spatial dimension, the pooling mechanism is first used to integrate spatial interaction in a local radius [15] or global scene [13]. Since the graph structure can better describe the trajectory scene, the graph-based methods [10], [11], [43], [44], [45], [46], [47] model the spatial interaction by graph neural networks (GCNs) [48], [49] and its variants [50]. Motivated by the success of Transformer [51], recent many works [9], [11], [24], [52], [53], [54] employ the self-attention mechanism to extract spatial features. Even [55] finetunes a large language model (LLM) to generate the social interactive features. However, the self-attention module in Transformer suffers from quadratic computation, and irrelevant interactions could influence the modeling of social interaction, increasing the risk of overfitting. Existing method [11] models the sparse attention to drop out the redundant interactions while it learns adaptive attention mask and thus still undergoes the quadratic computation. In contrast, our STP proposes an irregular interaction to engage with interested neighbors adaptively with a sparse style, thus reducing the computational complexity into a fixed window. What's more, prior transformer-based methods [52], [53] only focus on the social interaction extraction in spatial and temporal dimensions. In contrast, our STP unifies the pedestrian trajectory prediction modules, *i.e.*, social interaction, and multimodal trajectory prediction, into an encoder-decoder transformer architecture, achieving efficient and accurate prediction.

2.2 Multimodal Trajectory Prediction

Due to the motion multimodality [13], [22], pedestrians could take various motion behaviors represented by diverse future trajectories. There are two major generative strategies to deal with such multimodal prediction tasks. The former encoders the future trajectories into a latent space by a generative model, such as generative adversarial networks (GANs) [13], [24], conditional variational autoencoder (CVAE) [8], [41], [53], [56] and diffusion model [57]. They sample multiple latent variables in this space to decode multimodal future trajectories. Specifically, STAR [52] directly samples multiple latent variables and fuses them into social features to enforce the model outputting multimodal results. The latter assumes the trajectory points follow a Gaussian distribution [10], [11] or Gaussian Mixture Model (GMM) [12], [43] and estimate this distribution to obtain an explicit space, where the multimodal future trajectories are also obtained via multiple random samplings. In addition, some works [7], [14] employ the explicit memory bank to store multiple trajectory instances. SIT [9] attempts to build a hand-designed independent trajectory anchor to improve diversity. Due to the repeated random sampling, the sampled trajectories deviate from diversity, leading to inaccurate prediction. In addition, the memory banks suffer from the unbalance of the dataset, and the models suffer from strong bias when selecting the dominant trajectories, reducing diversity.

The late-sparsity strategy enhances prediction diversity to pursue accurate prediction further. PECNet [41] changes the variance of latent space in the sampling process. AgentFormer [53] penalizes the pairwise distance of predicted trajectories. However, a more effective method is using dense to sparse post-processing [7], [8], [14]. They first sample many trajectories and then cluster them into the desired number of trajectories. Unfortunately, this post-processing step suffers from the expensive prediction time and loses the probability of predicted trajectories. In contrast, STP employs an early-sparsity strategy to compress the world knowledge from the training data into sparse motion modes. By harnessing the expressive power of transformers, STP maps these sparse motion modes into multimodal future trajectories, significantly improving prediction speed while ensuring accuracy. MTR [58] is a concurrent work with our conference version [21] compressing the goal point to represent multimodal future trajectories. MTR focuses on vehicle trajectory prediction with rich HD maps and traffic elements (*e.g.*, lane and traffic sign) to restrict the movement of traffic agents. In contrast, pedestrian trajectory prediction focuses on the social interaction between pedestrians without strong rule constraints.

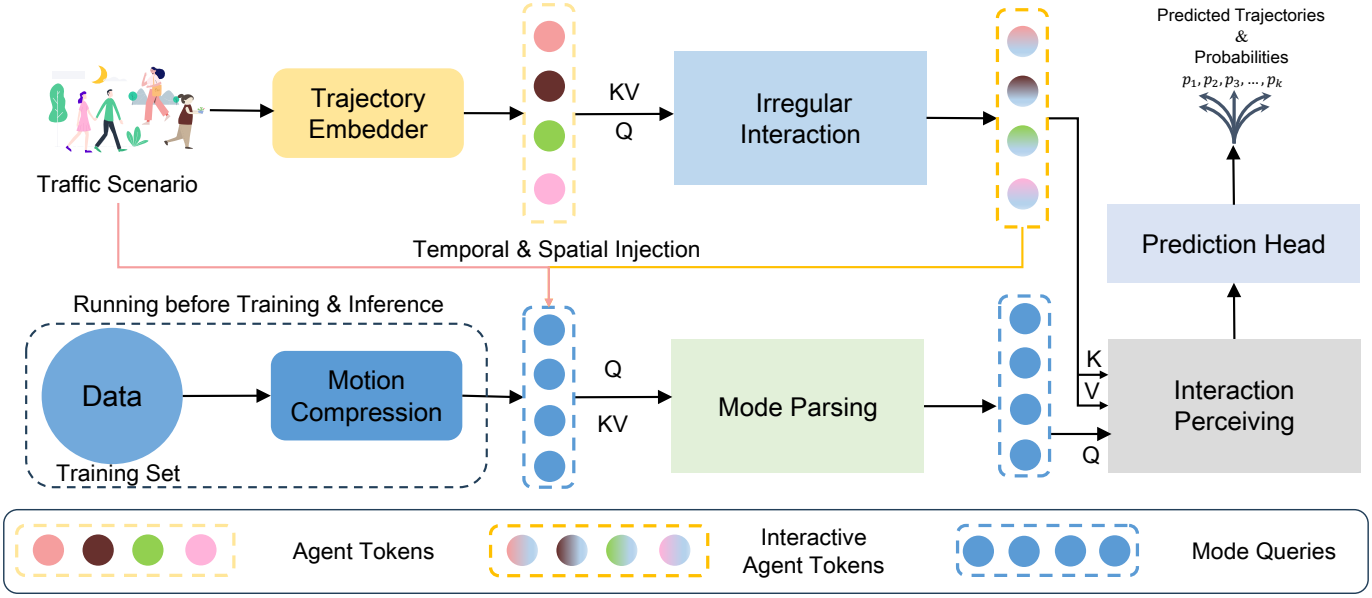


Fig. 3. The framework of STP. Before the model training and inference, STP employs the early-sparsity, where a motion compression module compresses the world knowledge from the entire training data into sparse mode tokens to represent multimodal motion behaviors. In the model training stage, the trajectory embedder generates agent tokens as the trajectory temporal feature for each pedestrian in the traffic scenario. After that, the social-level encoder builds the irregular interaction to generate the interactive agent tokens. Simultaneously, the generated sparse mode tokens are injected into the center pedestrian to obtain the mode queries. Finally, a mode parsing block distinguishes mutual relationships across mode queries, and an interaction perceiving block introduces the interactive information from the interactive agent tokens to predict the multimodal results.

3 Proposed Method

3.1 Problem Definition

Given the observed trajectories of multiple interactive pedestrians, pedestrian trajectory prediction aims to forecast the corresponding future trajectory. Assume that a traffic scenario with length T contains N pedestrians. We extract N trajectory coordinate sequences $\{x_t^n, y_t^n\}_{t=1,n=1}^{T,N}$ for each pedestrian n at time step t. The trajectory model observes the past sub-trajectories $\{x_t^n, y_t^n\}_{t=1,n=1}^{T_{obs},N}$ and predicts the future sub-trajectories $\{x_t^n, y_t^n\}_{t=1,n=1}^{T_{fut},N}$, where $T = T_{obs} + T_{fut}$. Due to the multimodality of pedestrian motion behaviors, the pedestrian could take multiple possible future trajectories. Therefore, the trajectory predictor is required to forecast diverse future trajectories, but only a single true future trajectory (ground truth) is provided for model training.

3.2 Method Overview

Our proposed Spare Trajectory prediction Model (STP) aims to achieve a dual focus both on efficiency and accuracy following an efficient principle. STP is packed into a transformer-style encoder-decoder architecture to forecast diverse future behaviors with three modules: motion compression, social interaction, and multimodal prediction. Thus, STP tokenizes the extracted various features to cater the transformer terminology. For the motion compression, the early-sparsity strategy is employed to perform the global motion behaviors with spare motion modes. To avoid the frequent and time-expensive late-sparsity, the world knowledge from the entire training data is compressed into spare motion modes, termed as sparse mode tokens, before the model training and inference, which are shared for all predictions. For the social interaction, a trajectory embedder captures the trajectory temporal features, named agent tokens, for each pedestrian in the traffic scenario. To perform global interaction with lower computational complexity, a social-level encoder receives the agent tokens to build irregular interaction with sparse pedestrians in the non-Euclidean trajectory space, producing interactive agent tokens. For the multimodal prediction, a mode embedder injects the sparse motion modes into a specific scenario to obtain mode queries. Leveraging the strong expressiveness of transformers, a trajectory-level decoder maps the mode queries into multimodal future trajectories by parsing their relationships and perceiving the interaction from interactive agent tokens, effectively improving the prediction speed and accuracy.

3.3 Motion Compression

Instead of relying on the frequent and time-expensive late-sparsity, STP employs an early-sparsity strategy to improve prediction speed. Following the efficient principle, this strategy involves a motion compressor, which condenses the world knowledge from the training data into sparse motion modes to represent the global motion behaviors. To achieve this, the motion compressor first employs two rigid transformations to align the training trajectories and then uses a one-time distance-based measurement to compress them into spare motion modes.

Trajectory Transformation. Given a fixed view, the trajectory is invariant to the rigid transformation. For example, a pedestrian going straight and then turning left shows the same behavior after applying translation or rotation to the trajectory of this pedestrian. For the training trajectories with length T, the front sub-trajectories with length T_{obs} are the observed trajectories, while the next sub-trajectories with length T_{fut} are the future trajectories. We first translate the T_{obs} trajectory points of the trajectories into the origin of the coordinate system. Then, the initial trajectory points of the translated trajectories are rotated to the positive X-axis. In this case, the direction of future trajectories is aligned to a relatively fixed region. That is, the distance between trajectories with similar motion behaviors is small. Thus, we can explicitly obtain the

diverse motion representations by a distance measurement strategy to cover global motion behaviors.

Distance Measurement. Based on the aligned training trajectories, we use the L_2 distance measurement to compress the trajectories into sparse motion modes. To perform the global motion behaviors with a sparse form, a one-time clustering operation is used on the aligned training trajectories to obtain L centers $\mathbf{C} \in \mathbb{R}^{L \times T_c \times 2}$ as the sparse motion modes, where $T_c \in [1, 2, ..., T]$ is the length of the spares motion modes. The value of T_c depends on the size of the clustered training trajectories. For example, $T_c = T_{fut}$ when we cluster the training future trajectories and $T_c = 1$ when we cluster the final point of training trajectories.

Thanks to our motion compression, the frequent and time-expensive late-sparsity for each prediction is reduced to a one-time compression for all predictions before the model training and inference step, effectively improving the prediction speed. The generated sparse motion modes are fed into the later decoder part to generate multimodal future trajectories.

3.4 Social Interaction

The social interaction module models the interactions among pedestrians in a traffic scenario. To avoid the quadratic global interaction, STP builds an irregular interaction to perform global interaction using the efficient sparse interaction guided by the proposed efficient principle. Specifically, STP first uses a trajectory embedder to generate the agent tokens and then employs a social-level encoder to produce the interactive agent tokens.

3.4.1 *Trajectory Embedder*

Before modeling social interaction, STP employs a trajectory embedder to capture the trajectory temporal dependence as the agent tokens. Given a specific traffic scenario with N pedestrians, we obtain N trajectory sequences $\mathbf{X} \in \mathbb{R}^{N \times T_{obs} \times c}$, where T_{obs} is the trajectory length and c is the dimension of a trajectory point. Since there exist invalid trajectory points caused by the missing detection or tracking, we apply the max-pooling operation on the temporal channel to obtain the individual temporal feature for each pedestrian. Due to the sequence-independent pooling, the one-hot sequential encoding is concatenated with the trajectory points to provide the temporal information before the trajectory embedding. A PointNet [59] based network is employed to generate agent tokens $\mathbf{E}_a \in \mathbb{R}^{N \times D}$, as follows:

$$\mathbf{E}^a = \phi(\mathbf{X}, \mathbf{W}^a) + \mathbf{b}^a, \quad (1)$$

where D represents the feature dimension, ϕ represents the stacked multi-layer perceptrons with the batch normalization and ReLU activation function. $\mathbf{W}^a$ and $\mathbf{b}^a$ are the learnable parameter matrices and bias, respectively.

3.4.2 *Social-level Encoder*

Based on the efficient principle, the social-level encoder performs global interaction in a sparse manner to improve computational efficiency. Concretely, an irregular attention is proposed to learn the multiple arbitrary interactive positions, where the sparse interaction is built around each position individually, and the global interaction is performed by fusing multiple sparse interactions. To this end, STP explores two strategies to achieve this irregular interaction. The first strategy is irregular grid interaction, which transforms the non-Euclidean trajectory space into an irregular grid space. Therefore, the deformable attention [16] can be used to build irregular interaction without the Euclidean constraint. The second strategy is irregular sampling interaction, which further breaks the grid constraint to directly model the sparse interaction between the interactive positions and their surrounding neighbors.

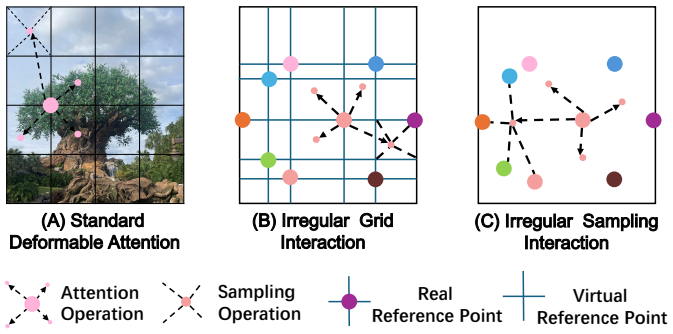


Fig. 4. Illustration of different irregular interactions. (A) is the standard deformable attention in an Euclidean space, while (B) and (C) are the proposed irregular grid interaction and irregular sampling interaction in a non-Euclidean space, respectively.

Preliminaries. The deformable attention (DA) is developed from the deformable convolution [60] to build attention with a small set of sampled keys. Compared to the self-attention mechanism [51], it generates learnable key positions and attention scores to achieve adaptive attention. Given an input feature map $\mathbf{I} \in \mathbb{R}^{C \times H \times W}$, the deformable attention is calculated by

$$\mathrm{DA}(\mathbf{z}, \mathbf{p}, \mathbf{I}) = \sum_{m=1}^{M} \mathbf{W}_m [\sum_{k=1}^{K} A_{mk} \cdot \mathbf{W}'_m \mathbf{I}(\mathbf{p} + \Delta \mathbf{p}_{mk})], \quad (2)$$

where $\mathbf{z}$ is the content feature of a query element with the 2D positions (reference points) $\mathbf{p}$, m and k index the attention head and sampled keys, respectively. K ($K \ll HW$) is the number of sampled keys. $\Delta \mathbf{p}_{mk} \in \mathbb{R}^2$ and $\{A_{mk} | \sum_{k=1}^{K} A_{mk} = 1\}$ represent the learnable offset and learnable scalar normalized attention score of the k^{th} sampled key in the m^{th} attention head, respectively. Both $\Delta \mathbf{p}_{mk}$ and A_{mk} are obtained by a learnable linear projection. $\mathbf{I}(\mathbf{p} + \Delta \mathbf{p}_{mk})$ is the key generation function implemented by bilinear interpolation due to the standard Euclidean space of the feature map. Unfortunately, this Euclidean requirement impedes the deformable attention to work in a non-Euclidean space.

Irregular Grid Interaction. The first strategy transforms the traffic scenario into an irregular grid as shown in Figure 4.(B). Given the traffic scenario $\mathbf{X} \in \mathbb{R}^{N \times T_{obs} \times c}$, we extract the current positions $\mathbf{p} \in \mathbb{R}^{N \times 2}$, *i.e.*, the position of the last trajectory point. The grid is generated via pairwise vertical connection between different abscissas and ordinates of these trajectory points. Since the density of the grid is not homogeneous compared to the images, we call this grid as irregular grid. That is, the cells in the grid have different sizes. For N trajectory points, the irregular grid has N^2 intersections, which are considered as the reference points in our irregular grid interaction. We name these intersections equal to the current positions real reference points, while the others are named virtual reference points. The content features of real reference points are initialized by the agent tokens, while the counterparts of the virtual reference points are initialized to zero.

Thanks to our real and virtual reference points, we can directly operate the standard deformable attention assisted by the generated

irregular grid to model the expected irregular interaction. The arbitrary interactive positions are obtained by the learnable offsets. The sparse neighbors are generated by the bilinear interpolation from the grid intersections around the interactive positions. Different from the encoder of the standard deformable attention, we only apply the attention on the real reference points to further reduce the computational complexity.

Irregular Sampling Interaction. Since the irregular grid exists many virtual reference points, leading to invalid interactions with zero content features. The second strategy, *i.e.*, irregular sampling interaction, is employed to break grid constraint as illustrated in Figure 4. (C). The irregular sampling interaction considers the discrete traffic scenario as an implicit continuous social interactive field based on a natural fact: an arbitrary position in the traffic scenario is affected by the nearby pedestrians.

Given the agent tokens $\mathbf{E}^a \in \mathbb{R}^{N\times D}$ with the positions $\mathbf{p} \in \mathbb{R}^2$, we first generates K learnable offsets $\{\Delta\mathbf{p}_k \in \mathbb{R}^{N\times 2}\}_{k=1}^{K}$ via a learnable linear projection on $\mathbf{E}^a$. Starting at the k^{th} learnable offset $\Delta\mathbf{p}_k \in \mathbb{R}^{N\times 2}$, the interactive position $\mathbf{p}_k^{\text{int}} \in \mathbb{R}^{N\times 2}$ is obtained by the vector addition between the the positions $\mathbf{p}$ and the corresponding offset $\Delta\mathbf{p}_k$.

To build the sparse interaction at the interactive position $\mathbf{p}_k^{\text{int}}$, two types of positional embeddings are used to encode the positions $\mathbf{p}$ and interactive points $\mathbf{p}_k^{\text{int}}$, formulated by:

$$\begin{aligned}\mathbf{P}_k^{\text{int}} &= \varphi(\mathbf{p}_k^{\text{int}}, \mathbf{W}^{\text{int}}) + \mathbf{b}^{\text{int}},\\ \mathbf{P}^{\text{cur}} &= \varphi(\mathbf{p}, \mathbf{W}^{\text{cur}}) + \mathbf{b}^{\text{cur}},\end{aligned} \tag{3}$$

where $\mathbf{P}_k^{\text{int}} \in \mathbb{R}^{N\times D}$ is the positional embedding of $\mathbf{p}_k^{\text{int}}$ to represent the interaction where the pedestrian does. $\mathbf{P}^{\text{cur}} \in \mathbb{R}^{N\times D}$ represents the current positions of neighbors. $\mathbf{P}^{\text{cur}}$ is obtained before the interaction and shared at each interaction block.

Subsequently, we generate the dynamic query and static key to prepare the sparse interaction around the interactive position $\mathbf{p}_k^{\text{int}}$, as follows:

$$\begin{aligned}\mathbf{Q} &= \varphi(\mathbf{E}_a + \mathbf{P}_k^{\text{int}}, \mathbf{W}^q) + \mathbf{b}^q\\ \mathbf{K} &= \varphi(\mathbf{E}_a + \mathbf{P}^{\text{cur}}, \mathbf{W}^k) + \mathbf{b}^k,\\ \mathbf{V} &= \varphi(\mathbf{E}_a, \mathbf{W}^v) + \mathbf{b}^v,\end{aligned} \tag{4}$$

where φ represents a learnable linear projection. $\mathbf{Q} \in \mathbb{R}^{N\times D}$ is the dynamic query with the dynamic positional information. $\mathbf{K} \in \mathbb{R}^{N\times D}$ is the static key with static positional information and $\mathbf{V} \in \mathbb{R}^{N\times D}$ is the value vector.

Therefore, the sparse interaction can be modeled with the nearest S neighbors around the interactive position $\mathbf{p}_k^{\text{int}}$, as follows:

$$\begin{aligned}\tilde{\mathbf{K}}_k, \tilde{\mathbf{V}}_k &= \mathcal{D}(\mathbf{K}, \mathbf{V}, \mathbf{p}, \mathbf{p}_k^{\text{int}}),\\ \mathbf{Z}_{mk} &= \text{softmax}(\frac{\mathbf{Q}_m \tilde{\mathbf{K}}_{mk}^T}{\sqrt{d}})\tilde{\mathbf{V}}_{mk},\end{aligned} \tag{5}$$

where $\mathcal{D}$ represents the distance function to find the nearest S neighbors. $\tilde{\mathbf{K}}_k \in \mathbb{R}^{N\times S\times D}$ and $\tilde{\mathbf{V}}_k \in \mathbb{R}^{N\times S\times D}$ are corresponding key vectors and value vectors of the nearest S neighbors, respectively. m indexes the m^{th} attention head. d represents the feature dimension of each head's query, key, and value vector. $\mathbf{Z}_{mk} \in \mathbb{R}^{N\times d}$ is the sparse interactive features of m^{th} attention head at the interactive position $\mathbf{p}_k^{\text{int}}$.

For the K interactive positions, we can obtain K sparse interactive features $\mathbf{Z} \in \mathbb{R}^{N\times S\times D}$ by concatenating K head-flattened sparse interactive features.

To perform the global interaction with the obtained sparse interactive features, we use an adaptive fusion, as follows:

$$\mathbf{S} = \varphi(\mathbf{Z}, \mathbf{W}^s) + \mathbf{b}^s, \tag{6}$$

where $\mathbf{S} \in \mathbb{R}^{N\times K\times 1}$ is the fusion score. Thus, the global interactive features $\mathbf{F} \in \mathbb{R}^{N\times D}$ are obtained by a weighted fusion operation as follows:

$$\begin{aligned}\hat{\mathbf{Z}} &= \varphi(\mathbf{Z}, \mathbf{W}^z) + \mathbf{b}^z,\\ \mathbf{F} &= \hat{\mathbf{Z}}^T \text{softmax}(\mathbf{S}),\end{aligned} \tag{7}$$

where $\hat{\mathbf{Z}} \in \mathbb{R}^{N\times K\times D}$ is the transformed features. The softmax function normalizes $\mathbf{S}$ on the second dimension.

Following the standard Transformer [51], $\mathbf{F}$ is fed into a layer normalization and a feed-forward network with residual connection to obtain the interactive agent tokens $\mathbf{E}^{ia} \in \mathbb{R}^{N\times D}$, which are fed into the next decoder part to introduce social interactions, thus predicting social-acceptable multimodal future trajectories.

Computational Complexity. We analyze the computational complexity of our proposed two types of irregular interaction against the previously commonly used global interaction based on the self-attention mechanism. Given N agent tokens with D dimension, the complexity of computing query, key, and value is $\mathcal{O}(3ND^2)$. The complexity of computing attention score and the softmax is $\mathcal{O}(N^2D + N^2)$. The complexity of the matrix multiplication between attention score and value vector is $\mathcal{O}(N^2D)$. Thus, the total complexity of global interaction is $\mathcal{O}(2N^2D+N^2+3ND^2)$, which grows quadratically with N.

For the irregular grid interaction, the generation of the irregular grid is implemented by an unstable sorting on the x and y components, respectively. Thus, the complexity of irregular grid generation is $\mathcal{O}(2NlogN)$, while it is only generated once to share with all interaction blocks. The complexity of computing offsets and normalized attention weights is $\mathcal{O}(3NDMK)$, where M and K are the number of attention heads and sampled keys, respectively. The complexity of computing values is $\mathcal{O}(ND^2)$. The complexity of bilinear interpolation and the weighted sum in attention is $\mathcal{O}(5NKD)$. The total complexity of irregular gird complexity is $\mathcal{O}(\min(2NlogN, 3NDMK + 5NKD + ND^2))$, growing linearly-logarithmic with N.

For the irregular sampling interaction, the complexity of computing offsets is $\mathcal{O}(2NDK)$. The complexity of computing dynamic positional embedding is $\mathcal{O}(2NDK)$. The complexity of computing key and value is $\mathcal{O}(2ND^2)$. The complexity of computing query is $\mathcal{O}(NKD^2)$. The nearest S neighbors are found in two operations, *i.e.*, distance computing, and distance sort. The complexity of distance computing is $\mathcal{O}(KN)$. Since we only require the top nearest S neighbors, the complexity of distance sort can be reduced into $\mathcal{O}(KN)$. Due to the sparse interaction, the complexity of computing attention score and the softmax is $\mathcal{O}(NKSD + NKS)$. The complexity of the matrix multiplication between attention score and value vector is $\mathcal{O}(NKSD)$. In general, the total complexity is about $\mathcal{O}(4NDK + 2ND^2 + NKD^2 + 2KN + 2NKSD + NKS)$, which grows linearly with N.

3.5 Multimodal Prediction

The multimodal prediction module receives the interactive agent tokens and sparse motion modes to predict multimodal future trajectories. To this end, STP first employs a mode embedder,

introducing scenario-specific information into the sparse motion modes to generate the mode queries. Then, STP builds a trajectory-level decoder to map the generated mode queries into multimodal future trajectories by leveraging the strong expressiveness of transformers.

3.5.1 Mode Embedder

Since the sparse mode tokens generated from our motion compression represent the global motion behaviors, the mode embedder injects temporal and spatial interaction information into the sparse motion modes, respectively.

Temporal Injection. The temporal interaction is represented by the temporal dependence of a trajectory. Given a specific trajectory $\mathbf{X}_i \in \mathbb{R}^{T_{obs} \times c}$ $(i \in [1, 2, ..., N])$ and the sparse mode modes $\mathbf{C} \in \mathbb{R}^{L \times T_c \times 2}$, we concatenate $\mathbf{X}_i$ and $\mathbf{C}$ to produce the time-specific sparse motion modes $\hat{\mathbf{C}} \in \mathbb{R}^{L \times \hat{T} \times 2}$ by extracting their positional information and broadcasting their shape, where $\hat{T} = T_{obs} + T_c$. Afterward, Multiple stacked multi-layer perceptrons with the batch normalization and ReLU activation function are used on $\hat{\mathbf{C}}$ to extract corresponding features $\mathbf{E}^{m,t} \in \mathbb{R}^{L \times D}$.

Spatial Injection. We further inject the spatial interactions from neighbors into $\mathbf{E}^{m,t}$. Specifically, the corresponding interactive agent token $\mathbf{E}_i^{ia} \in \mathbb{R}^{1 \times D}$ $(i \in [1, 2, ..., N])$ is concatenated with $\mathbf{E}^{m,t}$ to obtain the final scenario-specific sparse mode modes $\mathbf{E}^m \in \mathbb{R}^{L \times D}$, as follows:

$$\mathbf{E}^m = \phi(\mathbf{E}_i^{ia} \cup \mathbf{E}^{m,t}, \mathbf{W}^s) + \mathbf{b}^s, \tag{8}$$

where $\cup$ represents the shape-related function, which first broadcasts $\mathbf{E}_i^{ia}$ to align with $\mathbf{E}^{m,t}$ and then concatenates with them. ϕ is a multi-layer perceptron with a ReLU activation function. $\mathbf{E}^m$ are considered as the mode queries to generate multimodal future trajectories through the next decoder.

3.5.2 Trajectory-level Decoder

The trajectory-level decoder maps the motion queries into multimodal future trajectories with two blocks, *i.e.*, the mode parsing block, and the interaction perceiving block. The former parses the relationship across mode queries, and the latter perceives the encoded interactive features.

Mode Parsing. Unlike the above social-level encoder to obtain interactive features across pedestrians, this block parses the relationships across various mode queries to distinguish each other. Given the mode queries $\mathbf{E}^m$, the mode parsing block employs the multi-head self-attention mechanism on $\mathbf{E}^m$ to generate interactive mode queries. A layer-normalization layer with the residual connection [61] is stacked behind the attention operation. Note that we do not add the positional embedding in this self-attention block because the mode queries have included the positional information.

Interaction Perceiving. This block helps the interactive mode queries to perceive the social interactive information, thus predicting socially acceptable future trajectories. Specifically, a multi-head cross-attention mechanism is employed to achieve it, where the interactive mode queries are considered as the queries, and the interactive agent tokens generated from the social-level encoder are considered as the keys and values. Following the standard Transformer, a layer-normalization layer and a feed-forward network with residual connection are stacked behind the multi-head cross-attention mechanism. Note that positional embedding is unnecessary because the trajectory coordinates show the positional information of pedestrians.

3.5.3 Trajectory Prediction

This module contains a dual prediction and a guided prediction, where the former predicts the final multimodal future trajectories and the latter guides the feature learning of the social-level encoder.

dual Prediction. Most previous methods [7], [8], [14] predict diverse future trajectories but neglect the probabilities of predicted trajectories in pedestrian trajectory prediction. It is disadvantageous to make safe decisions. Here, we use dual prediction heads to simultaneously achieve regression and classification tasks. Specifically, a regression head and a classification head are employed on the final features obtained from the above prediction decoder to forecast diverse future trajectories and corresponding probabilities, respectively. The regression head and classification head are implemented by a multi-layer perceptron with the ReLU activation function, respectively. Different from the naive transformer, the regression and classification head are decoded into full diverse future trajectories and corresponding probabilities in parallel, not the autoregressive style.

Guided Prediction. To fully use the information of neighbor future trajectories, the guided prediction uses a regression head to predict neighbor future trajectory on the interactive agent tokens $\mathbf{E}^{ia}$ to guide the interaction learning of the social-level encoder. This regression head is implemented by a learnable linear projection. Different from the existing neighbor prediction [58], we do not further encode the predicted neighbor future trajectories to enhance the interactive agent tokens.

Model Training & Inference. Due to a single provided true future trajectory (ground truth) $\hat{\mathbf{Y}}$ for multimodal trajectory prediction, we use the variety loss [13] to optimize the prediction model. Given the predicted multimodal future trajectories $\{\mathbf{Y}_i\}_{i=1}^{L}$ and corresponding probabilities $\{p_i\}_{i=1}^{L}$, we first calculate the distance between each prediction and the ground truth, and then update the prediction $\mathbf{Y}_j$ by a smooth L_1 loss $\mathcal{L}^{\text{reg}}$, where j index the prediction with the minimum distance. For the classification, the index of p_j is considered as the target $\hat{p}$ to learn the prediction probabilities by a cross-entropy loss $\mathcal{L}^{\text{cls}}$. For the guided prediction, the predicted neighbor trajectory $\{\mathbf{Y}_n^{\text{nei}}\}_{n=1}^{N}$ are optimized by a smooth L_1 loss $\mathcal{L}^{\text{nei}}$ by the neighbor ground truth $\{\hat{\mathbf{Y}}_n^{\text{nei}}\}_{n=1}^{N}$.

Finally, STP can be trained in an end-to-end way as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}^{\text{reg}} + \lambda_2 \mathcal{L}^{\text{cls}} + \lambda_3 \mathcal{L}^{\text{nei}} \tag{9}$$

where λ_1, λ_2 and λ_3 are used to balance the loss function.

In the inference step, STP outputs multiple predicted trajectories and selects the expected number of predicted trajectories with the top predicted probabilities to cover diverse motion behaviors.

4 Experiments and Discussions

In this section, we show that STP, implemented on our efficient principle, achieves remarkable accuracy performance and faster inference speed compared to existing state-of-the-art methods with late-sparsity or not. In addition, we carry out detailed ablation studies to evaluate the performance contribution of each component of the proposed method. Finally, we further evaluate the effectiveness of STP by qualitative visualization evaluation.

4.1 Experimental Setting

Evaluation Datasets. We conduct experiments on four benchmark datasets, *i.e.*, ETH-UCY-V1 [17], [18], ETH-UCY-V2 [17], [18],

[52] Stanford Drone Dataset (SDD) [19] and SportVU NBA [20] to evaluate our proposed method. ETH [17] and UCY [18] are the most widely used benchmarks for pedestrian trajectory prediction. There are $1,536$ individual pedestrians in complex interactive scenarios, such as pedestrian crossing, group walking, and collision avoidance. Both versions, V1 and V2, contain 5 subsets, where ETH includes ETH and HOTEL subsets, and UCY includes UNIV, ZARA01, and ZARA02 subsets. The primary difference between V1 and V2 is the sampling interval of the ETH subset, where V1 has a longer interval than V2 to assess the model's generalization on unbalanced data distributions. Following prior studies [7], [8], we employ a leave-one-out cross-validation method, training on four subsets and testing on the remaining subset. The trajectories are recorded using meters as the unit.

SDD [19] is another benchmark for pedestrian trajectory prediction in a bird's-eye view. Different from the ETH-UCY-V1 and ETH-UCY-V2, collected from a single scenario for each subset. SDD is collected from various scenarios. It captures the trajectories of multiple types of agents (*e.g.*, pedestrians, bicyclists, skateboarders, cars, buses, and golf carts) on a university campus. The dataset includes over 11,000 individual pedestrians, resulting in more than 185,000 pedestrian interactions and 40,000 interactions between pedestrians and other scene elements. We adopt the standard training and testing splits used in previous studies [41], [62]. The trajectories in SDD are recorded in a pixel coordinate system, using pixel as the unit.

The SportVU NBA movement dataset [20] focuses exclusively on NBA games from the 2015-2016 regular season and provides rich interactions among players in a cooperative game setting. Due to the frequent adversarial and cooperative agent interactions and non-linear motions, the interactions in this dataset differ significantly from those in the ETH, UCY, and SDD datasets. Following prior work [8], we use two subsets as benchmarks: Rebounding and Scoring, which consist of 257,230 and 2,958,480 20-frame trajectories, respectively. The average trajectory length is approximately 4 meters, with a time interval of 0.12 seconds between frames.

Sampling Interval. We observe a trajectory of 8 time steps (3.2 seconds) and predict the next trajectory of 12 time steps (4.8 seconds) on ETH-UCY-V1 and SDD datasets. For the ETH-UCY-V2 dataset, we observe a trajectory of 8 time steps (1.92 seconds) and predict the subsequent trajectory of 12 time steps (2.88 seconds) on the ETH subset, while we observe a trajectory of 8 time steps (3.2 seconds) and predict the subsequent trajectory of 12 time steps (4.8 seconds) on the remaining four subsets. For the SportVU NBA movement dataset, we observe a trajectory of 8 time steps (0.96 seconds) and predict the subsequent trajectory of 12 time steps (1.44 seconds). Similar to existing methods, we construct a margin trajectory scenario to achieve trajectory prediction, where the scenario is normalized to originate the last trajectory point and orient the movement direction of the center pedestrian.

Evaluation Metrics. We evaluate our proposed and compared methods by four metrics, *i.g.*, Average Displacement Error (ADE), and Final Displacement Error (FDE), brier-ADE, and brier-FDE. Given the true future trajectory (ground truth) $\{x_t, y_t\}_{t=1}^{T_{fut}}$ and the corresponding predicted K trajectories, ADE and FDE are used to measure the $\ell 2$ distance between ground truth and the corresponding closest predicted trajectory $\{\hat{x}_t, \hat{y}_t\}_{t=1}^{T_{fut}}$,as shown in Eq. (10).

$$\begin{aligned} \text{ADE} &= \frac{1}{T_{fut}} \sum_{t=1}^{T_{fut}} \sqrt{(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2}, \\ \text{FDE} &= \sqrt{(x_{-1} - \hat{x}_{-1})^2 + (y_{-1} - \hat{y}_{-1})^2}, \end{aligned} \tag{10}$$

where subscript -1 index of the last point of the trajectory.

brier-ADE and brier-FDE [63] are similar to ADE and FDE but add the probability $\hat{p}$ of the closest predicted trajectory, as shown in Eq. (11):

$$\begin{aligned} \text{brier-ADE} &= \text{ADE} + (1 - \hat{p})^2, \\ \text{brier-FDE} &= \text{FDE} + (1 - \hat{p})^2. \end{aligned} \tag{11}$$

To evaluate the multimodal prediction performance, all methods generate 20 predicted future trajectories to measure the above metrics and report the minimum one to make comparisons fairly.

Implementation. We cluster the future trajectories to generate $L = 20$ sparse motion tokens with the length $T_c = 12$. The feature dimension $D = 128$ is shared on the social interaction and multimodal prediction modules. The number of attention heads $M = 8$ on all attention-related operations. We stack 2 multi-layer perceptrons (MLP) on the trajectory embedder. The number K of learnable offsets in irregular sampling interaction is set to 4. We find the nearest $S = 4$ neighbors to make the sparse interaction. We stack 2 MLPs on temporal injection. We stack 2 social-level encoders and 2 trajectory-level decoders on ETH-UCY-V1 and ETH-UCY-V2. We stack 2 social-level encoders and 3 trajectory-level decoders on SDD and NBA. The learning rate is set to 0.001 on four datasets with the AdamW optimizer. The cosine annealing is used to adjust the learning rate. The batch size is set to $128, 128$, and 256 on the ETH-UCY-V1, ETH-UCY-V2, and SDD, respectively. Due to the large number of data on the NBA dataset, the large batch size 4×256 accelerates the training process with distributed training. The epoch is set to 100 on ETH-UCY-V1, ETH-UCY-V2, and NBA, respectively. The epoch of the NBA is set at 500. All experiments are conducted on RTX 2080 Ti GPU, where the experiments on NBA use 4 GPUs and the rest of experiments use 1 GPU. We will release the related code to provide more detailed implementation.

4.2 Comparison with State-of-the-Art Methods

In this section, we compare STP with state-of-the-art multimodal pedestrian trajectory prediction methods on ETH [17] (two versions), UCY [18], SDD [19], and SportVU NBA [20].

ETH-UCY-V1. Table 1 presents the comparison results of our method with state-of-the-art methods on ADE and FDE metrics. STP reduces the average displacement error (ADE) and final displacement error (FDE) from 0.21/0.36 to 0.20/0.32 on average compared to the TUTR [21] (our conference version), demonstrating the effectiveness of the extended irregular interaction to capture global interaction with an efficient sparse structure. Furthermore, STP eliminates the accuracy gap compared to the methods with late-sparsity (marked by †). Specifically, STP achieves state-of-the-art performance in average ADE and is on par with the previous best method, *i.e.*, SocialVAE+FPC [8], in FDE. Compared to another late-sparsity method, MemoNet [7], STP shows significant superiority both on ADE and FDE. What's more, STP outperforms the recent diffusion-based method [57] and large language model (LLM) augmented method [55]. All of them

TABLE 1
Multimodal trajectory prediction comparison with state-of-the-art methods on ETH-UCY-V1 using ADE/FDE metrics. * is an augmented work by a large language model. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is underlined. The lower, the better.

Model	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	Avg
SGAN [13]	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 [24]	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 [23]	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 [44]	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 [44]	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 [64]	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 [10]	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 [11]	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† [41]	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
CAGN [12]	AAAI2022	0.41/0.65	0.13/0.23	0.32/0.54	0.21/0.38	0.16/0.33	0.25/0.43
SIT [9]	AAAI2022	0.39/0.62	0.14/0.22	0.27/0.47	0.29/0.33	0.16/0.29	0.23/0.38
SocialVAE [8]	ECCV2022	0.47/0.76	0.14/0.22	0.25/0.47	0.20/0.37	0.14/0.28	0.24/0.36
SocialVAE+FPC† [8]	ECCV2022	0.41/0.58	0.13/0.19	**0.21**/0.36	**0.17**/0.29	**0.13/0.22**	0.21/**0.32**
MemoNet† [7]	ICCV2022	0.40/0.61	**0.11**/0.17	0.24/0.43	0.18/0.32	0.14/0.24	0.21/0.35
LED [57]	CVPR2023	0.39/0.58	**0.11**/0.17	0.26/0.43	0.18/**0.26**	**0.13/0.22**	0.21/0.33
TUTR [21]	ICCV2023	0.40/0.61	**0.11**/0.18	0.23/0.42	0.18/0.34	**0.13**/0.25	0.21/0.36
LMTraj-SUP* [55]	CVPR2024	0.41/0.62	0.12/**0.16**	0.22/**0.35**	0.20/0.32	0.18/0.28	0.23/0.35
STP (Ours)	-	**0.36/0.51**	**0.11/0.16**	0.24/0.41	**0.17**/0.31	**0.13**/0.23	**0.20/0.32**

TABLE 2
Multimodal trajectory prediction comparison with state-of-the-art methods on ETH-UCY-V2 using ADE/FDE metrics. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower the better.

Model	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
STAR [52]	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 [62]	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 [43]	TPAMI2023	0.29/0.47	0.12/0.18	0.29/0.51	0.20/0.35	0.15/0.27	0.21/0.35
SICNet† [14]	ICCV2023	0.27/0.45	**0.11/0.16**	0.26/0.46	0.19/0.33	0.14/0.24	0.19/0.33
SingularTrajectory [65]	CVPR2024	0.35/0.42	0.13/0.19	**0.25/0.44**	0.19/**0.32**	0.15/0.25	0.21/0.32
STP (Ours)	-	**0.24/0.38**	**0.11/0.16**	0.26/0.45	**0.18/0.32**	**0.13/0.23**	**0.18/0.30**

demonstrate the effectiveness of our STP in generating accurate multimodal trajectory predictions.

ETH-UCY-V2. Table 2 shows the performance comparisons on ETH-UCY-V2, which has more balanced data among different subsets than ETH-UCY-V1. STP achieves the best or second-best performance on each subset and state-of-the-art performance both on average ADE and FDE. Specifically, STP improves the performance of the previous post-processing method SICNet [14] from 0.19/0.33 to 0.18/0.30 on ADE/FDE. Compared to a recent diffusion-based method [65], our method still outperforms it, reducing the ADE/FDE from 0.21/0.32 to 0.18/0.30. The remarkable experimental results on ETH-UCY-V1 and ETH-UCY-V2 demonstrate that STP is effective both in balanced and unbalanced trajectory scenarios.

Stanford Drone Dataset. We further evaluate our method on the commonly used rich-scenario dataset SDD. As shown in Table 3, STP significantly improves the accuracy performance of the conference version, TUTR [21], from 7.76/12.69 to 7.43/11.81 on ADE/FDE. Furthermore, STP achieves state-of-the-art ADE performance and the second-best FDE performance. Compared with the previous best method, SocialVAE+FPC [8] with late-sparsity, STP has a minor accuracy gap of 0.09 (11.81-11.72) on FDE, while SocialVAE+FPC has a larger performance gap of 0.67 (8.10-7.43) on ADE. In general, STP is superior to the compared methods and enables the removal of expensive late-sparsity to make efficient and accurate predictions.

SportVU NBA Movement Dataset. We further evaluate STP on a special trajectory dataset, *i.e.*, the SportVU NBA Movement Dataset, with player interaction compared to the foregoing pedestrian interaction. The experimental results are shown in Table 4, where STP significantly outperforms all previous methods without the late-sparsity. Compared to the prior best method, STP achieves the best ADE (0.05 improvement) and second-best FDE (0.03 gap) on the Rebounding and achieves state-of-the-art performance on the Scoring. In general, STP is superior to all previous methods, showcasing the effectiveness of multimodal prediction on special scenarios (ball game).

Comparison in brier-ADE/FDE. Since prior works neglect probability, we select multiple methods with different multimodal prediction strategies to make comparisons on brier-ADE/FDE. SIT [9] builds a manual tree to model multimodal future trajectories and provides probability information without late-sparsity.

TABLE 3
Multimodal trajectory prediction comparison with state-of-the-art methods for SDD on ADE/FDE metrics. † represents the methods with late-sparsity. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Model	Venue/Year	ADE	FDE
Sophie [24]	CVPR2019	16.27	29.38
SGAN [13]	CVPR2018	27.23	41.44
Desire [22]	CVRP2018	19.25	34.05
CF-VAE [24]	CVPR2019	12.60	22.30
SimAug [66]	ECCV2020	10.27	19.71
PECNet† [41]	ECCV2020	9.96	15.88
PCCSNet [62]	ICCV2021	8.62	16.16
SIT [9]	AAAI2022	9.13	15.42
IMP [43]	TPAMI2023	8.98	15.54
SocialVAE [8]	ECCV2022	8.88	14.81
SocialVAE+FPC† [8]	ECCV2022	8.10	**11.72**
MemoNet† [7]	CVPR2022	8.56	12.66
SICNet† [14]	ICCV2023	8.44	13.65
TUTR [21]	ICCV2023	<u>7.76</u>	12.69
STP (Ours)	-	**7.43**	<u>11.81</u>

CAGN [12] uses the Gaussian Mixture Model (GMM) to model diverse future trajectories without probability information. SocialVAE+FPC [8] first models a latent space to generate dense future trajectories and then cluster them into multimodal future trajectories but without probability information. TUTR [21] is our conference version, which predicts trajectories and corresponding probabilities. To generate the probabilities, we conduct two variants of CAGN and SocialVAE+FPC to make a comparison with our proposed method. CAGN predicts 20 Gaussian components and the learnable weights of each component are considered as the corresponding probabilities. SocialVAE+FPC predicts abundant trajectories and clusters them into a GMM, where the weights of each component are probabilities. As shown in Table 7 and Table 5, STP achieves state-of-the-art performance both on brier-ADE and brier-FDE. Specifically, STP improves the brier-ADE/brier-FDE of our conference version [21] from 0.95/1.10 to 0.86/1.01 on ETH-UCY-V1 and 8.44/13.53 to 8.33/12.90 on SDD. Furthermore, STP significantly outperforms the conducted variants on both datasets. All of that showcases the effectiveness of STP in predicting multimodal future trajectories and corresponding probabilities. In addition, we find that learning probability brings pressure to the prediction models. In our opinion, the worst performance of our STP on brier-FDE should be that the best FDE is selected with the worst probability (0), *i.e.*, $11.81 + 1 = 12.81$, referring to Table 3. However, the performance of brier-FDE (12.90) is worse than 12.81 as shown in Table 5. The similar phenomena also occurs in the compared methods.

Comparison in Prediction Speed. We compare the prediction speed with previous state-of-the-art methods in sparse and dense traffic scenarios, respectively. We set the number of pedestrians N as equal to $5, 10, 20, 40$, and 80, respectively. The larger N represents a more dense scenario. Note that the data-processing (*e.g.*, rotation and translation) is not considered to calculate the inference time. As shown in Table 6, both STP and TUTR (conference version [21]) significantly outperform the methods (MemoNet [7], SocialVAE+FPC [8]) with late-sparsity. Considering the prediction length with 4.8 seconds and 12 time steps, the predictor requires a prediction within 0.4 seconds to achieve real-time prediction. However, MemoNet and SocialVAE+FPC suffer from higher prediction delays that cost $1.2989s$ and $2.3401s$ to predict a $4.8s$ trajectory in a dense scene, respectively. In contrast, both STP and our conference version are capable of making real-time predictions easily. Specifically, STP achieves about $60\times$ speed improvement in sparse scenes and $20\times$ speed improvement in dense scenes compared to SocialVAE+FPC. Compared to our conference version, more time consumption comes from the interaction between each pair of pedestrians, while the TUTR only models the interaction between the central pedestrian and neighbor pedestrians. Nevertheless, our method eliminates the performance gap with the late-sparisty methods to dual focus both on efficient and accurate prediction. The comparisons between accuracy and speed showcase the feasibility of our proposed efficient principle, *i.e.*, leveraging the sparse structures to perform the global effects, on social interaction and multimodal prediction.

TABLE 4
Multimodal trajectory prediction comparisons with state-of-the-art methods on NBA using ADE/FDE metrics. The best performance is in bold formatting, while the second-best performance is represented with an underline. † represents the methods with late-sparsity. The lower, the better.

Method	Venue/Year	Rebounding	Scoring
Linear	-	2.14/5.09	2.07/4.81
Trajectron++ [56]	ECCV2020	0.98/1.93	0.73/1.46
BiTraP [67]	RAL2021	0.83/1.72	0.74/1.49
SGNet-ED [68]	RAL2022	0.78/1.55	0.68/1.30
SocialVAE [8]	ECCV2022	0.72/1.37	0.64/1.17
SocialVAE+FPC† [8]	ECCV2022	<u>0.66</u>/**1.10**	<u>0.58</u>/<u>0.95</u>
STP (Ours)		**0.61**/<u>1.13</u>	**0.51/0.91**

TABLE 5
Multimodal trajectory prediction comparisons on SDD using brier-ADE/brier-FDE. * represents the methods without probability prediction. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Method	Venue/Year	brier-ADE/brier-FDE
CAGN* [12]	AAAI2022	15.36/26.42
SIT [9]	AAAI2022	10.06/16.33
SocialVAE+FPC* [8]	ECCV2022	9.57/14.75
TUTR [21]	ICCV2023	<u>8.44</u>/<u>13.53</u>
STP (Ours)	-	**8.33/12.90**

TABLE 6
Comparisons on prediction time recorded by seconds. Our method significantly outperforms the compared methods.

N	MemoNet [7]	SocialVAE+FPC [8]	TUTR [21]	STP (Ours)
5	0.3221	0.6067	0.0050	0.0101
10	0.4058	0.7385	0.0046	0.0094
20	0.5358	0.9198	0.0050	0.0098
40	0.7784	1.3038	0.0052	0.0149
80	1.2989	2.3401	0.0076	0.0406

TABLE 7
Multimodal trajectory prediction comparisons on ETH-UCY-V1 using brier-ADE/brier-FDE. * represents the conducted model variant. The best performance is in bold formatting, while the second-best performance is represented with an underline. The lower, the better.

Method	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
CAGN* [12]	AAAI2022	1.43/1.78	1.18/1.44	1.47/2.04	1.29/1.78	1.23/1.65	1.32/1.73
SIT [9]	AAAI2022	1.29/1.49	1.03/1.14	1.38/1.82	1.08/1.23	0.99/1.13	1.15/1.36
SocialVAE+FPC* [8]	ECCV2022	1.37 / 1.61	1.02/1.09	1.12/1.31	1.07/1.20	1.04/1.17	1.12/1.27
TUTR [21]	ICCV2023	1.21/1.41	0.80/0.86	0.99/1.19	1.03/1.19	0.73/0.85	0.95/1.10
STP (Ours)	-	**1.20/1.38**	**0.52/0.61**	**0.98/1.18**	**0.98/1.15**	**0.64/0.75**	**0.86/1.01**

TABLE 8
Ablation study about sparse motion modes on ETH-UCY-V1 and Stanford Drone Dataset using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Generative Mode	0.39/0.58	0.12/0.18	0.26/0.48	0.19/0.34	0.15/0.27	0.22/0.37	8.05/12.92
Prior Generative Mode	0.50/0.82	0.15/0.24	0.41/0.84	0.23/0.41	0.18/0.32	0.29/0.52	10.88/18.68
Sparse Motion Mode (Ours)	**0.36/0.51**	**0.11/0.16**	0.24/**0.41**	**0.17/0.31**	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**
Learnable Embedding	0.36/0.53	0.12/0.18	**0.23/0.41**	**0.17/0.31**	**0.13/0.23**	0.20/0.33	7.54/12.22
Goal Point	**0.34/0.51**	**0.11**/0.18	**0.23/0.41**	**0.17/0.30**	**0.13/0.23**	**0.19/0.32**	7.71/12.17
Mean Point	0.35/**0.51**	**0.11**/0.17	0.24/0.42	**0.17**/0.31	**0.13/0.23**	0.20/**0.32**	7.54/11.98
Full Trajectory (Ours)	0.36/**0.51**	**0.11/0.16**	0.24/**0.41**	**0.17**/0.31	**0.13/0.23**	0.20/**0.32**	**7.43/11.81**

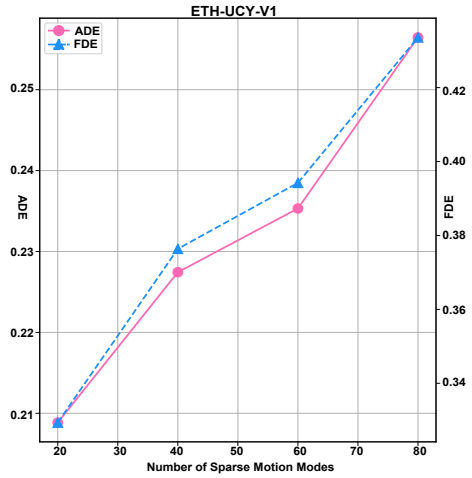


Fig. 5. Comparisons across different number of sparse motion modes on ETH-UCY-V1 using ADE/FDE metrics. The lower, the better.

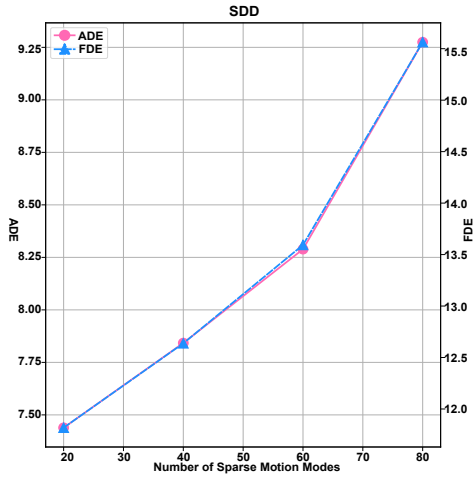


Fig. 6. Comparisons across different number of sparse motion modes on SDD using ADE/FDE metrics. The lower, the better.

4.3 Ablation Study

We conduct a series of ablative experiments to evaluate the performance contribution of the proposed early-sparsity and irregular interaction, where each component is replaced by the corresponding counterpart or removed while keeping the others unchanged.

4.3.1 *Early-Sparsity*

The early-sparsity is proposed to achieve efficient prediction. It involves a motion compressor to generate the sparse motion modes, which are the core to predict the final multimodal future behaviors.

Here, we conduct related experiments to validate the effectiveness of our sparse motion modes.

Comparison with prior Dense Motion Modes. We investigate the effectiveness of our generated sparse motion modes by making comparisons with previously used dense motion modes [8], [41], [52], which are sampled repeatedly from a latent space to represent the multimodal motion behaviors. We sample 20 latent variables from a normal Gaussian distribution (16 dimensions) to replace our sparse motion modes. Specifically, two variants are used to optimize these latent variables, *i.e.*, the Generative Mode and the Prior Generative Mode. The former uses the variety loss [13] to train the prediction model, while the latter [8], [41] uses variational strategy to generate multimodal results with different training and inference stages. In the training stage, the future trajectory is encoded into a latent variable as the center motion mode, combined

TABLE 9
Comparisons between the irregular interaction and global/local interaction on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Global Int	0.39/0.58	0.12/0.18	0.24/0.41	0.19/0.35	0.14/0.25	0.21/0.35	7.58/11.96
Global Int w Memory	0.37/0.52	0.12/0.19	**0.23**/0.41	**0.17/0.30**	0.15/0.26	**0.20**/0.33	7.62/12.06
Marginal Local Int	0.41/0.60	0.13/0.20	**0.23/0.40**	0.18/0.34	0.14/0.25	0.21/0.35	7.51/11.86
Joint Local Int w Memory	0.37/0.52	0.14/0.20	**0.23**/0.41	**0.17/0.30**	0.14/0.24	0.21/0.33	7.71/12.20
Irregular Int (Ours)	**0.36/0.51**	**0.11/0.16**	0.24/0.41	**0.17**/0.31	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**

TABLE 10
Ablation study about different irregular interactions on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

Method	ETH	HOTEL	UNIV	ZARA01	ZARA02	Avg	SDD
Irregular Grid Int	0.37/0.53	**0.11**/0.18	**0.24/0.40**	0.18/0.32	0.14/**0.23**	**0.20**/0.33	7.72/12.31
Irregular Sampling Int	**0.36/0.51**	**0.11/0.16**	**0.24**/0.41	**0.17/0.31**	**0.13/0.23**	**0.20/0.32**	**7.43/11.81**

with the sampled 19 latent variables to predict multimodal future trajectories. The center motion mode is enforced to follow a normal Gaussian distribution by the KL divergence, and the predicted trajectory from the center motion mode is used to optimize the prediction model. In the inference stage, we sample 20 latent variables to generate the prediction results. As shown in the upper block of Table 8, we observe that: 1) our method significantly outperforms the conducted two variants; 2) the prediction loss is more effective than the prior loss to improve accuracy. The reason for the first observation could be that the prediction from latent variables to future behaviors is a dense to sparse process, where similar samples lead to repetitive predictions and thus suffer from higher error. Thus, prior works [8], [14] use the late-sparsity to retrench the repetitive predictions and therefore improve accuracy but suffer from expensive prediction time. For the second observation, the Prior Generative Mode constrains the predicted trajectory generated from a prior latent variable by the prior loss, thus deviating from diversity. while the Generative Mode magnifies the differences of the sampled latent variables by the prediction loss to improve the diversity of predictions. Nonetheless, the random sampled latent variables within a certain number of sampling times can not represent the global distribution, thus impeding further accurate prediction. In contrast, our sparse motion modes compressed from world knowledge are capable of covering the global motion behaviors to pursue a more accurate prediction.

Comparison with different Sparse Motion Modes. The sparse motion modes represent the global motion behaviors. Here, we conduct multiple strategies to obtain different sparse motion modes. As shown in the lower block of Table 8, the Latent Embedding, referring to the object query in DETR [69], uses 20 learnable embeddings to build the sparse motion modes. It can be considered as a latent compression by the neural network. Referring to [41] and [43], the Goal Point clusters the last point of aligned training trajectories to generate sparse motion modes. In contrast, the Mean Point clusters the mean value of aligned training future trajectories into sparse motion modes. Note that neither [41] nor [43] cluster the goal points or mean value, just predicting them. The experimental results show that both the Latent Embedding, Goal Point, and Mean Value show apparent performance discrepancy on ETH-UCY-V1 and SDD datasets. Specifically, the Goal Point shows the best ADE on ETH-UCY-V1 while undergoing the worst ADE on SDD. Our method (Full Trajectory) reaches a balanced performance, achieving the best FDE on ETH-UCY-V1 and significant state-of-the-art ADE/FDE on SDD. It shows the well generalization of the full trajectory on various scenarios.

Impact of the Number of the Sparse Motion Modes. Multi-modal prediction in pedestrian trajectory prediction measures the prediction performance with a fixed number of motion modes. Here. we analyze the impact of the number of sparse motion modes. We cluster $20, 40, 60$ and 80 sparse motion modes to represent the diverse motion behaviors. To select the best 20 trajectories, we predict corresponding confidence when the number of sparse motion modes exceeds the required number of 20. As shown in Figure 5 and Figure 6, 20 motion modes achieve the best performance both on ETH-UCY-V1 and SDD datasets. The possible reason could be that the 20 motion modes are enough to cover the various motion behaviors of pedestrians. In addition, the classification task, *i.e.*, selecting the predicted trajectories with top-20 confidences is still challenging.

4.3.2 *Irregular Interaction*

Our STP proposes the irregular interaction to perform global interaction with efficient sparse interaction. We conduct ablated experiments against the commonly used global interaction and local interaction to evaluate its effectiveness. Additionally, we conduct detailed quantitative experiments to measure the impact of the hyper-parameters. Finally, we make detailed comparisons to assess the effectiveness of different irregular interactions. We default to using irregular sampling interaction for comparisons due to its superior performance.

Comparison with Global Interaction. Two strategies are used to make comparisons with global interaction. As shown in the first block of Table 9, the Global Int uses the multi-head self-attention mechanism [51] among all neighbors to model the social interactions, which are combined with the sparse motion modes to generate multimodal future trajectories, referring to the [52]. Global interaction models social interaction using the multi-head self-attention mechanism [51] across all neighbors. The Global Int w Memory adds extra memory attention behind the global interaction module to align with our method. The hidden dimension, number of

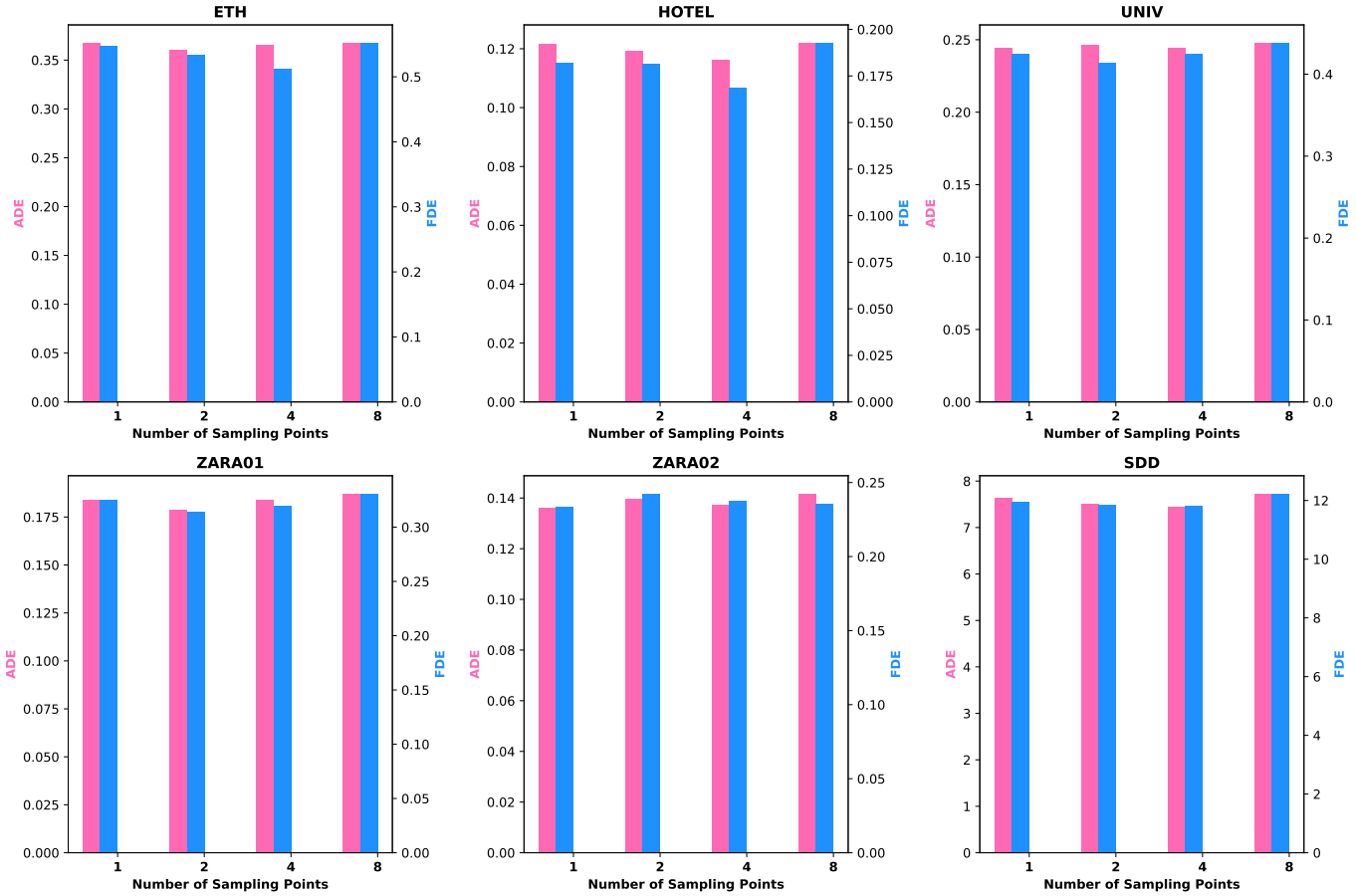


Fig. 7. Comparisons across different number of interactive positions on ETH-UCY-V1 and SDD using ADE/FDE metrics. The lower, the better.

heads, and number of layers are identical to our irregular interaction. Our method achieves the best performance on both ETH-UCY-V1 and SDD datasets. We observe that the global interaction with memory attention introduces a performance conflict, reducing the error on ETH-UCY-V1 but increasing the error on SDD. The reason could be that the memory module further models the interaction, bringing a double-edged sword effect on various interactive scenarios. In contrast, our irregular interaction models global interaction with a sparse style, which is generalizable and performs better on both datasets.

Comparison with Local Interaction. We further compare the prior local interaction, which models the interaction within the radius r. Referring to the SocialVAE [8], the r is set to 2 and 5 on the ETH-UCY-V1 and SDD datasets, respectively. As shown in the second block of Table 9, the Marginal Local Int only models the local interaction between the center object and its neighbor objects within the radius r, while the Joint Local Int w Memory first builds joint local interaction, where each object interacts with its neighbors within the same radius r. Then, the memory attention between the sparse motion modes and encoded interactive features is added behind the joint local interaction module. As shown in the second block of Table 9, our method achieves the lowest prediction error (ADE/FDE) both on ETH-UCY-V1 and SDD datasets. Similar to the global interaction, the memory modules also bring the performance conflict between two datasets. We further notice that the ETH-UCY-V1 is suitable for global interaction, while the SDD is suitable for local interaction. In contrast, our method can generalize into various interactive scenarios, significantly showcasing the superiority of our proposed irregular interaction.

Comparisons with Different Irregular Interactions. Here, we conduct ablated studies to compare two proposed irregular interactions. Specifically, the Irregular Gird Interaction (IGI) is one of our proposed methods to build irregular interaction using standard deformable attention mechanism [60] on an irregular grid. The Irregular Sampling Interaction (ISI) is the other proposed method to achieve efficient irregular interaction by adaptive sampling. As shown in Table 10, the ISI performs best on ETH-UCY-V1 and SDD datasets. The reason could be that IGI suffers from many empty interactions because many sampling locations in the built irregular grid are empty, as shown in Figure 4. In contrast, all sampling points in the ISI are valid to make a specific sparse interaction, leading to superior performance.

Impact of the number of Interactive Positions. The learnable interactive positions are the core to modeling global interaction with a sparse style for our irregular interaction. Here, we conduct experiments to analyze the impact of the number (K) of interactive positions. Specifically, we set K to $1, 2, 4, 8$, respectively. As illustrated in Figure 7, our irregular interaction shows lower sensitivity for different numbers of interactive positions on each dataset. The reason could be that the learnable interactive positions in our irregular interaction are adaptive. It can capture various sparse interactions at different layers despite the less interactive positions, such as $K = 1$. Observing the performance change, we

find that the $K = 1, 8$ are slightly worse than the $K = 2, 4$. We speculate that the more interactive positions lead to over-interaction for the scenarios with a small number of objects, while the less interactive positions result in under-interaction for the scenarios with many objects. $K = 4$ is a balanced value in different scenarios.

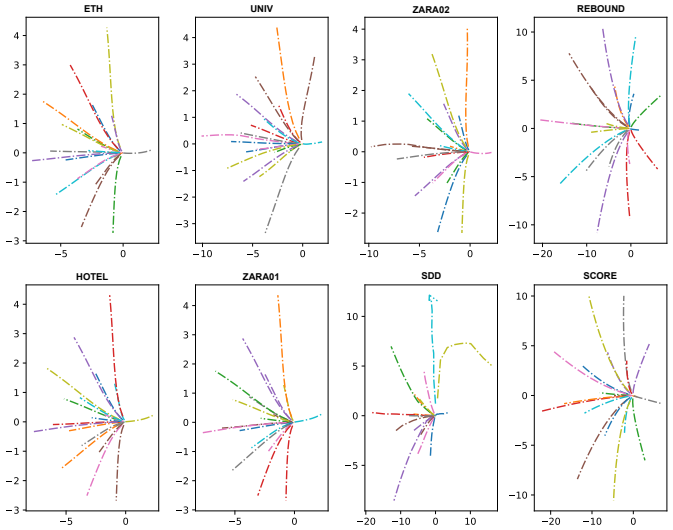


Fig. 8. Visualizations of the sparse motion modes generated from our proposed world compression. The motion direction is from right to left.

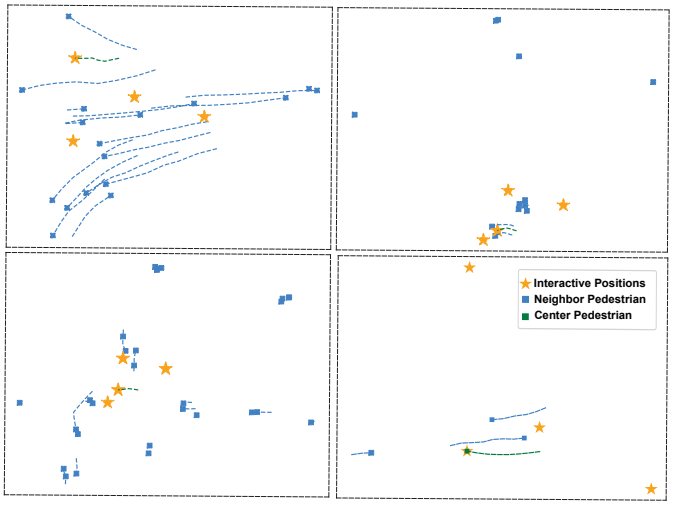


Fig. 9. Visualizations of the learnable interactive positions generated from our proposed irregular interaction. Best view in color.

4.4 Visualization Results

We provide the visualized results of STP on sparse motion modes, irregular interaction, and predicted multimodal future trajectories.

Sparse Motion Modes. Our efficient and accurate prediction comes from the generated sparse motion modes compressed from the world motion behaviors. We provide an intuitive visualization of the generated sparse motion modes to evaluate their ability to represent diverse motion behaviors. As shown in Figure 8, the generated sparse motion modes are capable of covering various motion behaviors, such as going straight, turning left/right with different angles, or turning back.

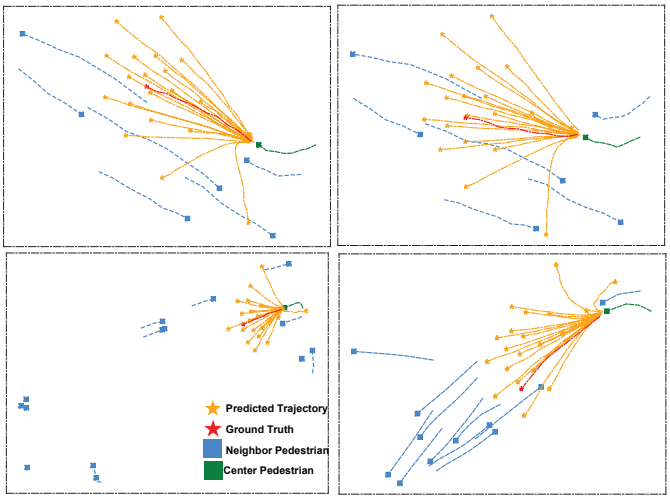


Fig. 10. Visualizations of the predicted multimodal future trajectories. Best view in color.

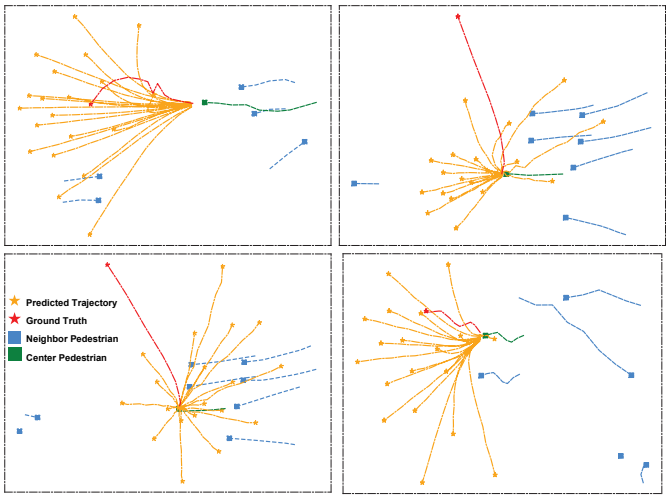


Fig. 11. Visualizations of the failure cases of the predicted multimodal future trajectories. Best view in color.

Irregular Interaction. Our irregular interaction builds various sparse interactions at learnable interactive positions to perform global interaction. We select some representative interactive scenarios from each of the subsets on ETH-UCY-V1 and SDD to visualize the distribution of the learned interactive positions. We visualize the learned interactive positions of the center pedestrian at the second interaction layer to make a consistency between ETH-UCY-V1 and SDD. As illustrated in Figure 9, our irregular interactions enable the model to capture various sparse interactions at different interactive positions. For the left two dense scenarios, the irregular interaction in the upper scenario captures the sparse interaction at the far positions compared to the irregular interaction in the lower scenario. For the right more sparse scenarios, the irregular interaction also captures the sparse interaction with various distances. It shows the adaptability of irregular interaction in various scenarios.

Predicted Trajectory. We visualize the predicted multimodal future trajectories to show the superiority in multimodal trajectory prediction. As shown in Figure 10, the visualized results show the STP is capable of covering the true motion intentions. Specifically, the upper two figures show the intention of turning right, while the

lower two figures show the intention of turning right. Furthermore, the predicted multimodal future trajectories show well diversity in covering various motion behaviors, such a going straight, turning left/right, avoiding collision, and walking with the dense crowd.

Failure Cases. The failure cases of our method are visualized in Figure 11, which is mainly reflected in the sharp turning and zigzag motion trajectory. The reason could be the trajectory is single to capture the complex motion intention. Ego-information, such as pose and face, is important for understanding detailed motion behaviors.

5 Conclusion

This paper devotes to build an efficient and accurate pedestrian trajectory prediction model. To achieve that, we present an efficient principle, *i.e.*, leveraging the sparse structure to perform global effects, to achieve both high accuracy and real-time speed. To this end, a sparse trajectory prediction model, termed STP, is developed to instantiate this efficient principle in the foundational social interaction module and multimodal prediction module of pedestrian trajectory prediction. For the social interaction module, an irregular interaction is proposed to perform global interaction with an efficient sparse interaction. Compared to the conflict between global and local interaction in various scenarios, the irregular interaction showcases the well generalization in various scenarios. For the multimodal prediction module, an early-sparsity strategy is proposed to generate the sparse motion modes before the model training and inference, which enables covering the global motion behaviors and avoiding the frequent late-sparsity to improve the prediction speed. Furthermore, the sparse motion modes show the effective ability to predict multimodal future trajectories. We evaluate STP on four commonly used datasets, and the experimental results demonstrate that STP maximizes both accuracy and prediction speed, achieving state-of-the-art performance and significantly improving inference speed by about $20\times$ - $60\times$ to satisfy the real-time demand.

Acknowledgments

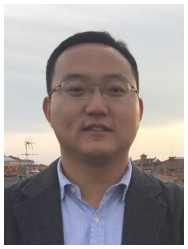
This work was supported in part by the National Key R&D Program of China under Grant 2021YFB1714700, in part by NSFC under Grants 62088102, 62106192, and 12326608, in part by the Natural Science Foundation of Shaanxi Province under Grant 2022JC-41, and in part by Fundamental Research Funds for the Central Universities under Grant XTR042021005.

References

[1] F. Leon and M. Gavrilescu, "A review of tracking, prediction and decision making methods for autonomous driving," *arXiv preprint arXiv:1909.07707*, 2019.
[2] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Trans. Intell. Trans. Syst.*, vol. 23, no. 1, pp. 33–47, 2020.
[3] 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.
[4] 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.
[5] 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.
[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] C. Xu, W. Mao, W. Zhang, and S. Chen, "Remember intentions: Retrospective-memory-based trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2022, pp. 6488–6497.
[8] P. Xu, J.-B. Hayet, and I. Karamouzas, "Socialvae: Human trajectory prediction using timewise latents," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 511–528.
[9] L. Shi, L. Wang, C. Long, S. Zhou, F. Zheng, N. Zheng, and G. Hua, "Social interpretable tree for pedestrian trajectory prediction," in *Proc. AAAI. Conf. Artif. Intell.*, vol. 36, no. 2, 2022, pp. 2235–2243.
[10] 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.
[11] 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.
[12] J. Duan, L. Wang, C. Long, S. Zhou, F. Zheng, L. Shi, and G. Hua, "Complementary attention gated network for pedestrian trajectory prediction," in *Proc. AAAI. Conf. Artif. Intell.*, vol. 36, no. 1, 2022, pp. 542–550.
[13] 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.
[14] Y. Dong, L. Wang, S. Zhou, and G. Hua, "Sparse instance conditioned multimodal trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 9763–9772.
[15] 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.
[16] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent.*, 2020.
[17] 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.
[18] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," in *Comput. Graphics Forum*, 2007, pp. 655–664.
[19] 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.
[20] Y. Yue, P. Lucey, P. Carr, A. Bialkowski, and I. Matthews, "Learning fine-grained spatial models for dynamic sports play prediction," in *Proc. IEEE Int. Conf. Data Mining*. IEEE, 2014, pp. 670–679.
[21] L. Shi, L. Wang, S. Zhou, and G. Hua, "Trajectory unified transformer for pedestrian trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 9675–9684.
[22] 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.
[23] 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.
[24] 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.
[25] 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.
[26] Z. Cao, H. Gao, K. Mangalam, Q.-Z. Cai, M. Vo, and J. Malik, "Long-term human motion prediction with scene context," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 387–404.
[27] 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. 15 233–15 242.
[28] N. Shafiee, T. Padir, and E. Elhamifar, "Introvert: Human trajectory prediction via conditional 3d attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16 815–16 825.

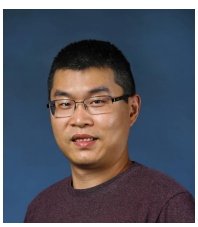
[29] A. Rasouli, M. Rohani, and J. Luo, "Bifold and semantic reasoning for pedestrian behavior prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15 600–15 610.
[30] J. Yue, D. Manocha, and H. Wang, "Human trajectory prediction via neural social physics," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 376–394.
[31] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 7, no. 2, pp. 2716–2723, 2022.
[32] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, p. 4282, 1995.
[33] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research: The 14th International Symposium ISRR*. Springer, 2011, pp. 3–19.
[34] I. Karamouzas, B. Skinner, and S. J. Guy, "Universal power law governing pedestrian interactions," *Physical Review Letters*, vol. 113, no. 23, p. 238701, 2014.
[35] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, 2007.
[36] K. Kim, D. Lee, and I. Essa, "Gaussian process regression flow for analysis of motion trajectories," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2011, pp. 1164–1171.
[37] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 961–971.
[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comp.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[39] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2342–2350.
[40] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2067–2075.
[41] 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.
[42] 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.
[43] L. Shi, L. Wang, C. Long, S. Zhou, W. Tang, N. Zheng, and G. Hua, "Representing multimodal behaviors with mean location for pedestrian trajectory prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 11 184–11 202, 2023.
[44] 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.
[45] 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.
[46] 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.
[47] 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.
[48] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. on neural networks*, vol. 20, no. 1, pp. 61–80, 2008.
[49] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2016.
[50] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018.
[51] 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.
[52] 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.
[53] 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.
[54] 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.
[55] I. Bae, J. Lee, and H.-G. Jeon, "Can language beat numerical regression? language-based multimodal trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.
[56] 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.
[57] W. Mao, C. Xu, Q. Zhu, S. Chen, and Y. Wang, "Leapfrog diffusion model for stochastic trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 5517–5526.
[58] S. Shi, L. Jiang, D. Dai, and B. Schiele, "Motion transformer with global intention localization and local movement refinement," *Proc. Adv. Neural Inf. Process. Syst.*, 2022.
[59] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.
[60] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 764–773.
[61] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[62] 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.
[63] M. Wang, X. Zhu, C. Yu, W. Li, Y. Ma, R. Jin, X. Ren, D. Ren, M. Wang, and W. Yang, "GANet: Goal area network for motion forecasting," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2023, pp. 1609–1615.
[64] 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.
[65] I. Bae, Y.-J. Park, and H.-G. Jeon, "Singulartrajectory: Universal trajectory predictor using diffusion model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.
[66] 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.
[67] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, "BiTraP: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 6, no. 2, pp. 1463–1470, 2021.
[68] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *Proc. IEEE Conf. Rob. Autom. Lett.*, vol. 7, no. 2, pp. 2716–2723, 2022.
[69] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

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. From 2023 to 2024, he is a visiting scholar with University of Illinois at Chicago, USA. He is currently a Ph.D. student with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University. His research interests include automatic driving and multimodal video understanding.



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 is 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 PR, MVA, and PRL.

Sanping Zhou (Member, IEEE) received his Ph.D. degree from Xi'an Jiaotong University, Xi'an, China, in 2020. From 2018 to 2019, he was a Visiting Ph.D. Student with Robotics Institute, Carnegie Mellon University. He is currently an Associate Professor with the Institute of Artificial Intelligence and Robotics at Xi'an Jiaotong University. His research interests include machine learning, computer vision and embodied intelligence, with a focus on meta learning, multi-task learning, object detection, multi-target tracking, trajectory prediction, medical image segmentation, visual navigation and visual grasping.



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.

Gang Hua (Fellow, IEEE) received the B.S. and M.S. degrees in Automatic Control Engineering from Xi'an Jiaotong University (XJTU), Xi'an, China, in 1999 and 2002, respectively. He received the Ph.D. degree in Electrical Engineering and Computer Science at Northwestern University, Evanston, Illinois, USA, in 2006. He is currently the Vice President of the Multimodal Experiences Research Lab at Dolby Laboratories. His research focuses on computer vision, pattern recognition, machine learning, robotics, towards general Artificial Intelligence, with primary applications in cloud and edge intelligence. Before that, he was the CTO of Convenience Bee, and the Managing Director and Chief Scientist of its research branch in US, Wormpex AI Research (2018-2024). He also 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 Senior 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 a 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 Senior Scientist (2006-09) at Microsoft Live labs Research. He is an associate editor of TPAMI and MVA. He is a general chair of ICCV'2027 and a program chair of CVPR'2019&2022. He is the author of more than 200 peer reviewed publications in prestigious international journals and conferences. He holds 35 US patents and has 15 more US patents pending. He is the recipient of the 2015 IAPR Young Biometrics Investigator Award. He is an IEEE Fellow, an IAPR Fellow, and an ACM Distinguished Scientist.

Trajectory Unified Transformer for Pedestrian Trajectory Prediction

Liushuai Shi[1] Le Wang[1*] Sanping Zhou[1] Gang Hua[2]
[1]National Key Laboratory of Human-Machine Hybrid Augmented Intelligence,
National Engineering Research Center for Visual Information and Applications,
Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
[2]Wormpex AI Research

Abstract

Pedestrian trajectory prediction is an essential link to understanding human behavior. Recent work achieves state-of-the-art performance gained from hand-designed post-processing, e.g., clustering. However, this post-processing suffers from expensive inference time and neglects the probability that the predicted trajectory disturbs downstream safety decisions. In this paper, we present Trajectory Unified TRansformer, called TUTR, which unifies the trajectory prediction components, social interaction, and multimodal trajectory prediction, into a transformer encoder-decoder architecture to effectively remove the need for post-processing. Specifically, TUTR parses the relationships across various motion modes using an explicit global prediction and an implicit mode-level transformer encoder. Then, TUTR attends to the social interactions with neighbors by a social-level transformer decoder. Finally, a dual prediction forecasts diverse trajectories and corresponding probabilities in parallel without post-processing. TUTR achieves state-of-the-art accuracy performance and improvements in inference speed of about $10\times$ - $40\times$ compared to previous well-tuned state-of-the-art methods using post-processing.

1. Introduction

Pedestrian trajectory prediction aims to predict the future trajectory based on an observed trajectory. It is an essential link that connects the perception system upward and the planning system downward [13, 18]. Due to the randomness of human motion, there are diverse plausible future trajectories that pedestrians could take [6]. The popular predictor addresses this multimodal prediction task in a generative style. They model the multimodality of the future trajectory in a specific space, such as an explicit Gaussian space [17, 27, 5], a latent space [34, 16, 37, 6], a hand-planning space [26], or a memory bank [33].

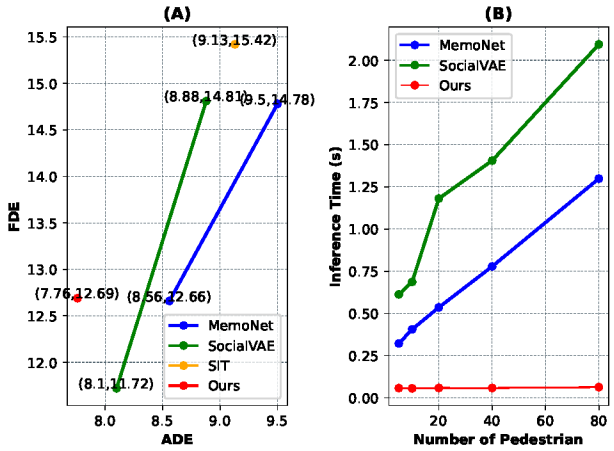


Figure 1. (A) shows the comparison against the methods (MemoNet[33], SocialVAE[34]) with post-processing and the SIT[26] without post-processing, Simultaneously, it presents the accuracy performance variances of MemoNet and SocialVAE that use post-processing or not. (B) shows the inference speed variances as the number of pedestrians increases. Our method achieves a balance of accuracy performance and inference speed.

Recently, some works [33, 34] have achieved significant advances benefiting from hand-designed post-processing, as illustrated in Figure 1 (A). Most of the time, they first sample more plausible future trajectories than the desired number of predictions K. Then, a clustering algorithm (e.g., K-means) is operated on sampled trajectories to generate the desired K predictions, similar to NMS [9] in object detection. However, this post-processing suffers from expensive inference time due to the non-parallel loop iteration in clustering, especially for a dense scene.

As shown in Figure 1 (B), the methods with post-processing lead to being more and more time-consuming as the number of pedestrians increases. Furthermore, the post-processing neglects probability information, disturbing safety decisions. Actually, most works forecast diverse

*Corresponding author.

trajectories equally (*i.e.*, without probability information) in pedestrian trajectory prediction. Similarly, the clustering operation also obtains K centers with equal weights. Although these predictors have significant performance in best-of-K prediction, there is no information on which is the best. It is a disadvantage for the safety decision of an intelligent system such as autonomous driving. Our goal aims to bridge the gap between accuracy and inference speed, keeping the corresponding probabilities of predicted trajectories simultaneously.

To address the above problems, we propose Trajectory Unified TRansformer (named TUTR) to effectively eliminate the need for post-processing in pedestrian trajectory prediction. TUTR unifies the components of pedestrian trajectory prediction, such as social interaction and multimodal trajectory prediction, into a transformer encoder-decoder architecture as illustrated in Figure 2.

TUTR first parses relationships across various motion modes using an explicit global prediction and an implicit mode-level transformer decoder. Specifically, global prediction employs two rigid transformations on training trajectories to obtain general motion modes, which are considered as the input token of the mode-level transformer encoder. Afterward, TUTR attends to the social interactions with neighbors using a social-level transformer decoder to prepare a social-acceptable prediction. Finally, a dual prediction is used to forecast diverse trajectories and corresponding probabilities in parallel to cover the multimodality of future trajectories without any post-processing.

We evaluate TUTR on the most popular datasets for pedestrian trajectory prediction, ETH [19], UCY [14], and SDD [22]. The experimental results show that our proposed method achieves a comparable accuracy performance and faster inference speed without any post-processing step compared with existing state-of-the-art methods. Moreover, TUTR performs the best performance in brier-ADE and brier-FDE, which are two metrics that consider the probabilities of predicted trajectories.

In summary, the contributions of this paper are summarized as follows.

- We propose a new pedestrian trajectory prediction framework (TUTR) based on encoder-decoder transformer architecture entirely to unify the pedestrian trajectory prediction.
- TUTR parses the relationship across various motion modes by the explicit global prediction and implicit mode-level transformer encoder to effectively remove the need for post-processing.
- TUTR achieves state-of-the-art ADE/brier-ADE/brier-FDE, and comparable performance in FDE. Moreover, TUTR performs a faster inference speed to balance accuracy performance and inference speed.

2. Related Works

Research on pedestrian trajectory prediction is briefly categorized into two classes: prediction based on environment information (*e.g.*, semantic map) [2, 15, 25, 21, 38, 30] and prediction based on social interaction from neighbors. In this paper, we focus on the latter to effectively remove the need for post-processing by unifying the pedestrian trajectory prediction into an encoder-decoder architecture.

2.1. Pedestrian Trajectory Prediction

Physical Models. Before deep learning, many works design specific physical models to forecast a deterministic future trajectory. Social force [8], motion velocity [28], and energy [11] are commonly used to model the motion behavior of pedestrians. Also, some works employ the statistical model, such as Gaussian processes [31, 12], to deal with the uncertainty of future trajectories. However, they suffer from bad generalizations when facing various motion patterns and social interactions.

Deep Learning Models. As deep learning develops in the community, most deep models in pedestrian trajectory prediction forecast future trajectories via feature extraction and multimodal trajectory prediction. In feature extraction, many works use deep models, such as recurrent neural networks (RNNs) [1, 6, 34], attention mechanisms [23, 27, 26, 36, 37], and graph neural networks (GCNs) [17], to model the temporal sequential features from the observed trajectory and spatial interaction features from neighbors.

Multimodal Trajectory Prediction. Pedestrians could take various future trajectories due to their motion randomness [6]. To deal with such multimodal prediction tasks, many works employ generative models, such as generative adversarial networks (GANs) [6] and conditional variational autoencoder (CVAE) [16, 34, 37, 24], to generate diverse future trajectories. Besides, some works [26, 17, 5] model the possible future trajectories into a Gaussian distribution or a Gaussian Mixture Model (GMM). A tree-based model [26] covers the possible future trajectories by an interpretable tree. In addition, the memory-based methods [33] store the diverse trajectories in a memory bank. Recently, the post-processing step is commonly used to improve the diversity of predicted trajectories. PECNet [16] changes the variance of latent space. AgentFormer [37] penalizes the pairwise distance of predicted trajectories. However, a more effective post-processing step [33, 34] is sampling a large number of predicted trajectories and then clustering them into the desired number of centers. Unfortunately, the post-processing step, especially for the clustering, suffers from the expensive inference time and loses the probability of predicted trajectories. In contrast, TUTR can directly forecast diverse trajectories without any post-processing step and achieves a balance between inference time and accuracy performance.

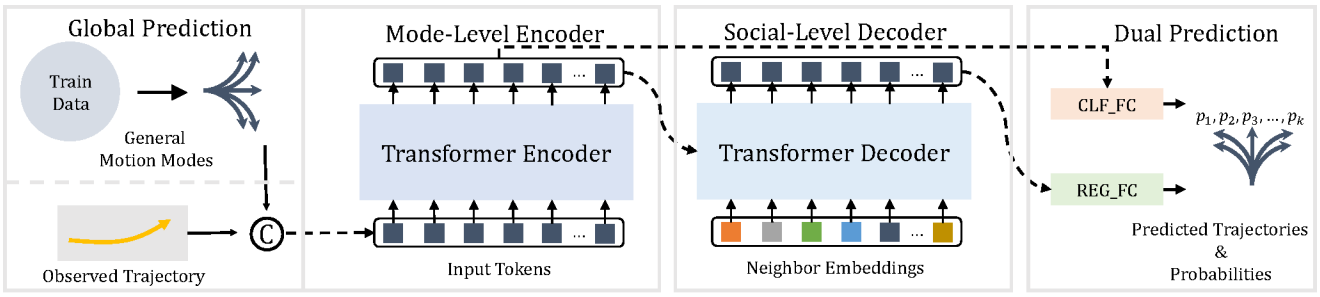


Figure 2. An overview of TUTR. TUTR employs an encoder-decoder transformer architecture to forecast future motion behaviors. Firstly, the global prediction generates general motion modes. Then, the general motion modes concatenated with the observed embedding are considered as the input tokens of a mode-level transformer encoder. Subsequently, the encoder output attends to the social interactions by a social-level decoder. Finally, two shared prediction heads in dual prediction are used to obtain the dual results, *i.e.*, predicted trajectories, and corresponding probabilities.

2.2. Transformers

The Transformer model [29] is first proposed in the machine translation task to replace the recurrent neural networks (RNNs) [10]. Transformers are now popular in many tasks, such as in natural language processing [35, 20] and computer vision [3, 4]. Transformers encode global features through the self-attention mechanism in parallel. Then, the encoder-decoder attention (cross-attention) in the Transformer decoder generates the desired output. In the naive Transformer, the decoder is an auto-regressive model to output tokens one by one.

Unlike previous applications of the transformer to extract global features, TUTR is mainly used to address the question of output, *i.e.*, multimodal trajectory prediction similar to [3] in object detection. Concretely, TUTR first design a global prediction on the whole training trajectories to obtain general motion behaviors, which are considered as the input tokens of the encoder of the transformer. Then, a decoder attends to social interactions with neighbors and the results of the decoder to forecast diverse trajectories in parallel, not the autoregressive style.

Some methods [37, 36] have explored the Transformer [29] architecture in the prediction of pedestrian trajectory. However, the transformer is only used to extract temporal and spatial features. Besides, they employ an auto-regressive model to output trajectory points one by one. Compared to them, TUTR unifies the pedestrian trajectory prediction modules, such as feature extraction and multimodal trajectory prediction, into an encoder-decoder transformer architecture, which includes a mode-level encoder, a social-level decoder, and two dual prediction heads. It achieves better performance and contributes to compatibility with other modules, such as upward motion perception and downstream motion planning. What's more, TUTR employs parallel decoding to generate diverse trajectories, further improving the inference speed compared with auto-regressive decoding.

3. Our Method

3.1. Problem Definition

Pedestrian trajectory prediction aims to forecast the future trajectory of a pedestrian based on the observed trajectories of the pedestrian and its neighbors. Assume that a sequence of traffic scenes with the length T contains N pedestrians. We extract N trajectory coordinate sequences $\{x_t^n, y_t^n\}_{t=1,n=1}^{T,N}$ for each pedestrian at each time step. The trajectory model observes the front of sub-trajectories $\{x_t^n, y_t^n\}_{t=1,n=1}^{T_{obs},N}$ and predicts the next sub-trajectories $\{x_t^n, y_t^n\}_{t=T_{obs}+1,n=1}^{T,N}$. Due to the multimodality of pedestrian motion behavior, there are multiple future trajectories that the pedestrian could take. Therefore, the trajectory model is required to forecast diverse future trajectories, while only a single true future trajectory (ground truth) is provided for model training.

3.2. TUTR Architecture

Here, we introduce our proposed Trajectory Unified TRansformer (TUTR), which contains four components packed into a transformer encoder-decoder architecture to forecast diverse future behaviors as illustrated in Figure 2. The explicit *global prediction* and the implicit *mode-level transformer encoder* are used to parse the relationships across various motion modes. Subsequently, the encoder output attends to the social interactions with neighbors using a *social-level transformer decoder*. Finally, a *dual prediction* is used to obtain dual results (diverse future trajectories and corresponding probabilities) in parallel by two shared prediction heads.

Recall that previous transformer-based methods [36, 37] employ the transformer architecture only on the observed trajectory and its neighbors. Namely, the trajectory points of an observed trajectory are the input tokens of a temporal transformer encoder to obtain temporal features. The trajectory points of the neighbors are the input tokens of a spatial

transformer encoder to obtain spatial features. However, the multimodality of the future trajectory is the main challenge that affects prediction accuracy. Unlike them, TUTR parses the relationship across various modes of future behavior by a *mode-level* transformer encoder. Then, TUTR attends to the social interactions using a *social-level* transformer decoder to directly output the diverse future trajectories without any post-processing step.

Global Prediction. TUTR parses the relationships across various motion modes using an explicit global prediction and an implicit mode-level transformer encoder. Global prediction obtains general motion modes to cover common motion behaviors of a pedestrian, and the results are considered as the input token of the next mode-level transformer encoder. Here, we first employ two rigid transformations to generate normalized trajectories and then use a distance measurement to obtain the general motion modes.

Given a fixed view, the trajectory is invariant for the rigid transformation. For example, a pedestrian shows going straight and then turning left. It also shows the same behaviors after translation or rotation for the trajectory of the pedestrian. For the training trajectories with length T, the front sub-trajectories with length T_{obs} are the observed trajectories, while the next sub-trajectories with length T_{pred} are the future trajectories. We first translate the T_{obs} trajectory points of the trajectories into the origin of the coordinate system. Then, the initial trajectory points of the translated trajectories are rotated to the positive X-axis. In this case, the direction of most future trajectories is normalized to a relatively fixed region. Namely, the trajectories with similar motion behaviors could have a small distance. Thus, we can obtain diverse trajectories explicitly in a distance measurement strategy to cover the common motion behaviors.

Therefore, a clustering operation is used on the normalized future trajectories to obtain L centers $C \in \mathbb{R}^{L\times T_{pred}\times 2}$, where $C = \{c_1, ..., c_L\}$ and each $\{c_l | l \in 1, ..., L\}$ is a trajectory with length T_{pred}. Thus, the centers C represent the general motion modes, which are the input tokens of the next mode-level transformer encoder. Note that C is invariant in the inference step. Namely, global prediction does not lead to additional inference time.

Observed embedding. The general motion modes are considered as the input token of the next described mode-level transformer encoder. They are first reshaped into a $L \times 2T_{pred}$ features and then embedded by a learnable linear transformation to obtain input embeddings E_c as follows:

$$E_c = \phi(C, \boldsymbol{W}_c), \tag{1}$$

where $\phi(\cdot, \cdot)$ is a linear transformation with a learnable parameter matrix $\boldsymbol{W}_c \in \mathbb{R}^{2T_{pred}\times D_e}$, $E_c \in \mathbb{R}^{L\times D_e}$ is the input embeddings.

The previous input embeddings of the transformers need an extra positional embedding [29] to deal with the permutation-invariant of the self-attention mechanism. Unlike them, the elements in C are not limited by their sequences. Thus, the positional embedding is not necessary for the input embedding E_c. However, our goal is to predict diverse trajectories of a pedestrian based on its observed motion states. The input embeddings E_c are required to fit the given input observed trajectory $X \in \mathbb{R}^{B\times T_{obs}\times 2}$, where B is the batch size. Hence, the observed trajectory X is embedded and added to the input embeddings E_c as follows:

$$\begin{aligned} E_o &= \phi(X, \boldsymbol{W}_o), \\ E_e &= E_c + E_o, \end{aligned} \tag{2}$$

where X is reshaped into $B \times 2T_{obs}$ before the linear transformation, $\boldsymbol{W}_o \in \mathbb{R}^{2T_{obs}\times D_e}$ is the learnable parameter matrix. We broadcast the dimensions of E_c and E_o to $B \times L \times D_e$ and perform an add operation between them to obtain the final embedding $E_e \in \mathbb{R}^{B\times L\times D_e}$.

Mode-Level Transformer Encoder. Unlike feature-level transformer encoders to build the global dependence of trajectory points, the mode-level transformer encoder parses the relationships across various modes. Given the input embedding E_e that represents general motion modes based on the observed trajectory, the mode-level transformer encoder employs the standard encoder architecture of a naive transformer on E_e to parse the relationships across various motion modes. Each encoder block includes a multi-head self-attention layer and a Feed-Forward Network (FFN) with the residual connection [7]. Unlike the naive transformer encoder, which adds positional embedding at each encoder block, the observed embedding occurs only once.

Social-Level Transformer Decoder. This decoder is used to extract social interactions with neighbors. It follows the standard decoder architecture of a naive transformer, including an attention layer and a Feed-Forward Network (FFN). The differences with naive transformers lie in four aspects: First, the decoder receives neighboring embeddings, not masked output embeddings. Second, TUTR keeps the encoder-decoder attention and empirically removes the self-attention. Third, the positional embedding is not necessary for the input embedding because the trajectory coordinates have shown the position relationships between pedestrians and their neighbors. Finally, the output embeddings are decoded into diverse future trajectories and corresponding probabilities by the next described dual-prediction in parallel, not the autoregressive style.

Assume that a pedestrian has N neighbors, represented by the neighbor observed trajectories $X_s \in \mathbb{R}^{N\times T_{obs}\times 2}$. Each trajectory in X_s is flattened into a feature vector, leading to a feature matrix $\hat{X}_s \in \mathbb{R}^{N\times 2T_{obs}}$. Then, we embed the feature matrix by a learnable linear transformation to

obtain the input embeddings of the social-level transformer decoder as follows:

$$E_s = \phi(\hat{X}_s, \boldsymbol{W}_s), \tag{3}$$

where $E_s \in \mathbb{R}^{N \times D_e}$ is the input embeddings of the decoder, $\boldsymbol{W}_s \in \mathbb{R}^{2T_{obs} \times D_e}$ is the learnable parameter matrix. After that, the input embeddings E_s are transformed into output embeddings with the subsequent encoder-decoder attention layer and an FFN layer with the residual connection. In this case, these output embeddings attend to the social interactions to forecast social-acceptable trajectories and corresponding probabilities by the next dual prediction.

Dual Prediction. Most previous methods [16, 37, 33, 34] predict diverse future trajectories but neglect the probabilities of predicted trajectories. It is a disadvantage to the safety decision. Here, we use a dual prediction to achieve regression and classification tasks simultaneously. As illustrated in Figure 2, a shared regression prediction head (REG_FC) and a shared classification prediction head (CLF_FC) are used to forecast diverse future trajectories and corresponding probabilities, respectively. In the implementation, we empirically find that placing the classification prediction head to the back of the mode-level encoder could bring better accuracy performance.

Model Training. Due to a single provided true future trajectory (ground truth) $\hat{Y}$ for multimodal trajectory prediction, we employ a greedy training strategy. Specifically, we first obtain the closest clustering centers $c_i, i \in \{1, ..., L\}$ by distance measurement between the ground truth $\hat{Y}$ and L clustering centers $C = \{c_1, ..., c_L\}$ as follows:

$$i = \underset{i \in \{1,...,L\}}{\operatorname{argmin}} (||\hat{Y} - c_i||_2^2). \tag{4}$$

Next, we employ a *nearest neighbor hypothesis*, which means the ground truth $\hat{Y}$ can be obtained by a (deep) transformation of the closest centers c_i, and the predicted trajectory obtained from c_i is the most likely one, *i.e.*, owning the maximum probability. The soft probability $\hat{p}$ of c_i can be represented by the normalized negative distance as follows:

$$p = \operatorname{softmax}(\{-||\hat{Y} - c_i||_2^2 \mid i \in \{1, ..., L\}\}). \tag{5}$$

Consequently, TUTR is used to transform c_i into the desired $\hat{Y}$. In this case, we could predict the future trajectory and its probability with the current motion mode by the ith output embedding of the decoder in the training step, resulting in a predicted trajectory Y and the corresponding soft probabilities p. Finally, TUTR can be trained in an end-to-end way as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{reg}(Y, \hat{Y}) + \lambda_2 \mathcal{L}_{clf}(p, \hat{p}), \tag{6}$$

where λ_1 and λ_2 are used to balance the loss function, $\mathcal{L}_{reg}$ is the Huber loss, and $\mathcal{L}_{clf}$ is the cross entropy loss.

In the inference step, TUTR outputs multiple predicted trajectories and selects K predicted trajectories with Top-K probabilities to cover diverse motion behaviors.

4. Experiments and Discussions

In this section, we show that TUTR achieves a comparable accuracy performance and faster inference speed compared to existing state-of-the-art methods that benefit from the well-designed post-processing step. In addition, we provide detailed ablation studies on components of the proposed method. Finally, we further evaluate the effectiveness of TUTR by qualitative visualization evaluation.

4.1. Experiments Setting

Datasets. We conduct experiments on two benchmark datasets, *i.e.*, ETH-UCY [19, 14], and Stanford Drone Dataset (SDD) [22], to evaluate our proposed method. ETH-UCY is the most widely used benchmark for pedestrian trajectory prediction. It contains trajectories of 1,536 pedestrians collected in four different scenarios with a bird's eye view and divided into five subsets, ETH, HOTEL, UNIV, ZARA1, and ZARA2. On ETH-UCY, we follow prior works [33, 34] that use a leave-one-out method for model training. Namely, we train the proposed model on four subsets and test it on the rest of the subsets. SDD is a larger benchmark dataset in pedestrian trajectory prediction, also captured by bird's eye view. It contains the trajectories of 5,232 pedestrians recorded in eight different scenarios. On SDD, we use the previous train-test split [16] to train and test our proposed model. The model observes the trajectory with length $T_{obs} = 8$ (3.2 seconds) and predicts the next $T_{pred} = 12$ (4.8 seconds) trajectory.

Evaluation Metrics. We evaluate our proposed and compared methods by four metrics, *i.g.*, Average Displacement Error (ADE) and Final Displacement Error (FDE), brier-ADE and brier-FDE. Given a true future trajectory (ground truth) $\{x_t, y_t\}_{t=T_{obs}+1}^{T}$ and the corresponding predicted K trajectories, ADE and FDE are used to measure the $\ell 2$ distance between ground truth and the corresponding closest predicted trajectory $\{\hat{x}_t, \hat{y}_t\}_{t=T_{obs}+1}^{T}$,as shown in Eq. (7).

$$\begin{aligned} \text{ADE} &= \frac{1}{T_{pred}} \sum_{t=T_{obs}}^{T} \sqrt{(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2}, \\ \text{FDE} &= \sqrt{(x_T - \hat{x}_T)^2 + (y_T - \hat{y}_T)^2}. \end{aligned} \tag{7}$$

brier-ADE and brier-FDE [32] are similar to ADE and FDE but add the probability p of the closest predicted trajectory, as shown in Eq. (8):

Method	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Social GAN [6]	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 [23]	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
STAR [36]	ECCV2020	**0.36**/0.64	0.17/0.36	0.31/0.62	0.29/0.52	0.22/0.46	0.26/0.53
SGCN [27]	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
CAGN [5]	AAAI2022	0.41/0.65	0.13/0.23	0.32/0.54	0.21/0.38	0.16/0.33	0.25/0.43
SIT [26]	AAAI2022	0.39/0.62	0.14/0.22	0.27/0.47	0.19/**0.33**	0.16/0.29	0.23/0.38
SocialVAE [34]	ECCV2022	0.47/0.76	0.14/0.22	0.25/0.47	0.20/0.37	0.14/0.28	0.24/0.42
PECNet [16]	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
AgentFormer [37]	ICCV2021	0.45/0.75	0.14/0.22	0.25/0.45	0.18/0.30	0.14/0.24	0.23/0.39
MemoNet [33]	CVPR2022	**0.40**/0.61	**0.11/0.17**	0.24/0.43	0.18/0.32	0.14/0.24	**0.21**/0.35
SocialVAE+FPC [34]	ECCV2022	0.41/**0.58**	0.13/0.19	**0.21/0.36**	**0.17/0.29**	**0.13/0.22**	**0.21/0.32**
Ours (TUTR)	-	**0.40**/0.61	**0.11**/0.18	0.23/0.42	0.18/0.34	**0.13**/0.25	**0.21**/0.36

Table 1. Comparison with state-of-the-art methods on ETH-UCY in ADE/FDE. The first block is the comparisons against the methods without the post-processing step, while the second block is the comparisons against the methods with the post-processing step.

$$\begin{aligned} \text{brier-ADE} &= \text{ADE} + (1-p)^2, \\ \text{brier-FDE} &= \text{FDE} + (1-p)^2. \end{aligned} \tag{8}$$

Implementation Details. In our conducted experiments, the trajectories are translated to the origin and then rotated to the X-axis to be consistent with the general motion modes. On ETH-UCY, the number of general motion modes $L = 50, 90, 50, 70, 50$ for the ETH, HOTEL, UNIV, ZARA1, and ZARA2, respectively. The embedding dimension D_e is equal to 128. We stack 2 mode-level transformer encoders with 4 attention heads and 128 FFN hidden dimensions. We stack 1 social-level transformer decoder with 4 attention heads and 128 FFN hidden dimensions. On SDD, the number of general motion modes $L = 100$ and the embedding dimension $D_e = 64$. We stack 2 mode-level transformer encoders with 4 attention heads and 128 FFN hidden dimensions. We stack 1 social-level transformer decoder with 4 attention heads and 128 FFN hidden dimensions. All experiments are conducted on a single RTX 3090 GPU.

4.2. Comparison with State-of-art Methods

Comparison in ADE/FDE on ETH-UCY. As shown in Table 1, the first block shows the comparisons against methods without post-processing. TUTR achieves state-of-the-art performance in both average ADE and average FDE. Specifically, TUTR improves the average ADE/FDE from $0.23/0.38$ to $0.21/0.35$ compared to the previous best method, SIT [26]. The second block in Table 1 shows the comparison against the methods with the post-processing step. TUTR shows competitive performance in average ADE metrics, being on par with the methods (MemoNet [33] and SocialVAE+FPC [34]) with a post-processing step. However, TUTR still shows a performance gap (0.04) in average FDE metrics, against the previous best methods, SocialVAE+FPC [34].

Method	Venue/Year	ADE/FDE
Social GAN [6]	CVPR2018	27.23/41.44
SoPhie [23]	CVPR2019	16.27/29.38
CAGN [5]	AAAI2022	9.42/15.93
SIT [26]	AAAI2022	9.13/15.42
MemoNet [33]	CVPR2022	9.50/14.78
SocialVAE [34]	ECCV2022	8.88/14.81
PECNet [16]	ECCV2020	9.96/15.88
MemoNet [33]	CVPR2022	8.56/12.66
SocialVAE+FPC [34]	ECCV2022	8.10/**11.72**
Ours (TUTR)	-	**7.76**/12.69

Table 2. Comparison with state-of-the-art methods on SDD in ADE/FDE. The first block is the comparisons against the methods without the post-processing step, while the second block is the comparisons against the methods with the post-processing step.

Comparison in ADE/FDE on SDD. As shown in Table 2, the first block shows the comparisons against the methods without post-processing. TUTR also achieves state-of-the-art performance both in ADE and FDE. Specifically, TUTR improves the ADE/FDE from $8.88/14.81$ to $7.79/12.73$ compared with the previous best method, SocialVAE [34]. The second block in Table 2 shows the comparisons against the methods with the post-processing step. TUTR shows state-of-the-art performance in ADE metrics, improving the ADE from 8.10 to 7.79 compared with previous methods, SocialVAE+FPC [34]. However, TUTR also shows a performance gap (1.01) in FDE metrics, against the previous best method, SocialVAE+FPC [34].

Comparison in brier-ADE/FDE. Since many works ne-

Method	Venue/Year	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
CAGN* [5]	AAAI2022	1.43/1.78	1.18/1.44	1.47/2.04	1.29/1.78	1.23/1.65	1.32/1.73
SIT [26]	AAAI2022	1.29/1.49	1.03/1.14	1.38/1.82	1.08/1.23	0.99/1.13	1.15/1.36
SocialVAE+FPC* [34]	ECCV2022	1.37 / 1.61	1.02/1.09	1.12/1.31	1.07/1.20	1.04/1.17	1.12/1.27
Ours (TUTR)	-	**1.21/1.41**	**0.8/0.86**	**0.99/1.19**	**1.03/1.19**	**0.73/0.85**	**0.95/1.10**

Table 3. Comparisons on ETH-UCY in brier-ADE/brier-FDE. * represents the conducted model variant.

glect probabilities, we select three methods with different multimodal trajectory prediction strategies to make a comparison in brier-ADE/FDE. SIT [26] provides probability information without post-processing. CAGN [5] uses the Gaussian Mixture Model (GMM) to model diverse future trajectories but without probability information. SocialVAE+FPC [34] is the previous state-of-the-art method with post-processing but without probability information. We conduct two variants of VAGN and SocialVAE+FPC to make a comparison with TUTR. CAGN predicts 20 Gaussian components and the weights of each component as probabilities. SocialVAE+FPC predicts abundant trajectories and clusters them into a GMM, where the weights of each component are considered as probabilities.

As shown in Table 3 and Table 4, TUTR achieves state-of-the-art performance in both brier-ADE and brier-FDE. Specifically, TUTR reduces the brier-ADE/brier-FDE from 1.12/1.17 to 0.95/1.1 on ETH-UCY compared to SocialVAE+FPC [34]. On SDD, TUTR reduces brier-ADE/brier-FDE from 9.57/14.75 to 8.44/13.53 compared to SocialVAE+FPC.

Comparison in Inference Speed. We compare the inference speed with previous state-of-the-art methods in sparse and dense pedestrian motion scenes, respectively. We set the number of pedestrians N as equal to $5, 10, 20, 40$, and 80, respectively. The larger N represents more dense scenes. As shown in Table 5, TUTR significantly outperforms the methods (MemoNet [33], SocialVAE+FPC [34]) with post-processing step significantly. Specifically, MemoNet and SocialVAE+FPC suffer from the higher prediction delays that they cost $1.2989s$ and $2.0939s$ to predict a $4.8s$ trajectory in a dense scene, respectively. In contrast, TUTR achieves about $10\times$ speed improvement in sparse scenes and $40\times$ speed improvement in dense scenes. The inference speed variance is also shown in Figure 1 (B). In conclusion, TUTR achieves a balance between accuracy performance and inference speed.

4.3. Ablation Studies

Importance of Global Prediction. We conduct a variant to evaluate the importance of global prediction, referring to object queries [3]. Specifically, the L general motion modes obtained from the global prediction are replaced by L learnable latent vectors. In this case, the nearest neighbor hypothesis is not available because the latent vectors cannot provide information on which latent vector is closest to the ground truth. Therefore, we use a variety loss [6] to predict trajectories. The experimental results demonstrate that the latent vectors suffer from a large performance reduction, enlarging the average ADE/FDE from 0.21/0.36 to 0.34/0.64 on ETH-UCY and from 7.76/12.69 to 17.26/34.64 on SDD. The reason lies that latent vectors cannot provide useful information to guide neural networks to generate diverse trajectories compared with general motion modes.

Number of General Motion Modes. The general motion modes are used to represent the common motion behaviors of a pedestrian. Here. we analyze the impact of the number of general motion modes L as shown in Figure 4, where the experimental results show that $L = 100$ achieves the best performance. The reason could be that too few general motion modes can not cover common motion behaviors, and too many general motion modes disturb the neural network to search for effective modes.

Importance of Model Components. We conduct three variants to evaluate the components of TUTR. As shown in Table 6, GP is the global prediction, MTE is the mode-level transformer encoder, and STD is the social-level transformer decoder. GP is replaced by multiple learnable latent vectors similar to the before-mentioned ablation study of global prediction. The MTE is replaced by a feed-forward network [29] to perform an ablation study. The experimental results show that each component is effective in predicting diverse future trajectories.

4.4. Qualitative Analysis

General Motion Modes. Here, we provide an intuitive visualization of general motion modes to evaluate their ability to cover common motion behaviors of a pedestrian. Note that the general motion modes are obtained on normalized trajectories, *i.e.*, the direction of pedestrian motion is from right to left. As shown in Figure 3, the general motion modes could represent the common motion behaviors, *e.g.*, going straight, turning left/right, or turning back.

Predicted Diverse Trajectories. As shown in Figure 5, the predicted trajectories have a good diversity to cover various motion behaviors of pedestrians, such as turning left/right (1,4), going straight (3), keeping standing (6) and sharp turning (2, 5). Moreover, TUTR can predict the best

Method	Venue/Year	ADE/FDE
CAGN* [5]	AAAI2022	17.82/35.78
SIT [26]	AAAI2022	10.06/16.33
SocialVAE+FPC* [34]	ECCV2022	9.57/14.75
Ours (TUTR)	-	**8.44/13.53**

Table 4. Comparisons on SDD in brier-ADE/brier-FDE. * represents the methods without probability prediction.

N	MemoNet [33]	SocialVAE+FPC [34]	Ours
5	0.3221	0.6127	**0.0577**
10	0.4058	0.6869	**0.0561**
20	0.5358	1.1807	**0.0586**
40	0.7784	1.4053	**0.0582**
80	1.2989	2.0939	**0.0533**

Table 5. Comparisons in inference time recorded by seconds. Our method significantly outperforms the compared methods.

Variant	GP	MTE	STD	ADE/FDE
(1)	✗	✓	✓	17.26/34.64
(2)	✓	✗	✗	8.14/13.46
(3)	✓	✓	✗	7.85/12.91
(4)	✓	✓	✓	**7.76/12.69**

Table 6. Ablation study of TUTR on SDD dataset in ADE/FDE.

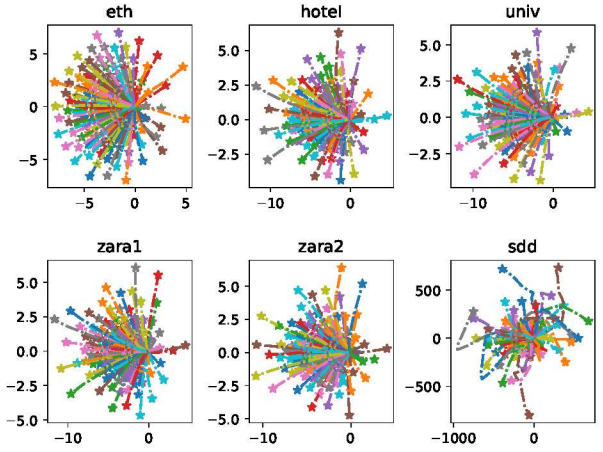


Figure 3. Visualizations of the general motion modes on ETH-UCY and SDD. The motion direction is from right to left.

trajectory with high probability.

5. Conclusion

In this paper, we present a trajectory-unified framework named TUTR, which unifies the social interaction and multimodal trajectory prediction into an encoder-decoder transformer architecture to remove the need for post-processing. The experimental results show that TUTR achieves competitive accuracy performance compared with previous state-of-the-art methods that gain from the well-designed post-processing. What's more, TUTR performs about $10\times-40\times$ inference speed improvements. However, the clustering algorithm is hard to match complex data structures, such as map information. How to learn more robust mode representations is worth exploring in the future.

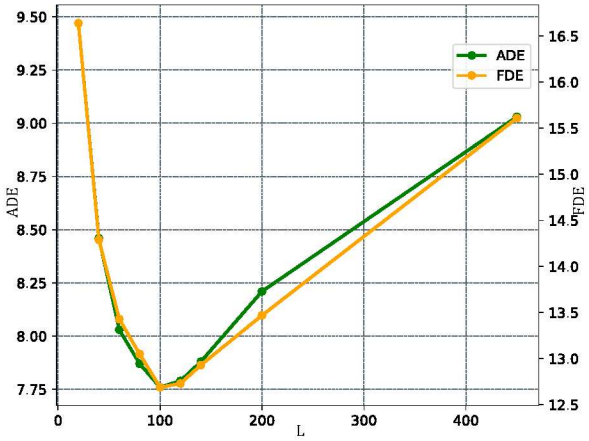


Figure 4. Ablation study of the number of general motion modes L on SDD dataset. $L = 100$ is the best performance.

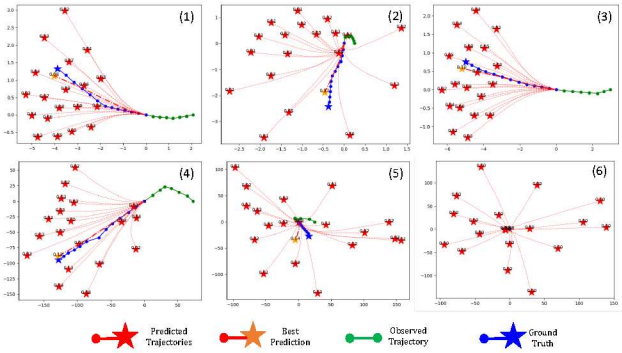


Figure 5. Visualization of predicted trajectories and corresponding probabilities.

Acknowledgement

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

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *CVPR*, pages 961–971, 2016. 2

[2] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *ECCV*, pages 387–404. Springer, 2020. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020. 3, 7

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[5] Jinghai Duan, Le Wang, Chengjiang Long, Sanping Zhou, Fang Zheng, Liushuai Shi, and Gang Hua. Complementary attention gated network for pedestrian trajectory prediction. In *AAAI*, pages 542–550, 2022. 1, 2, 6, 7, 8

[6] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, pages 2255–2264, 2018. 1, 2, 6, 7

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4

[8] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282, 1995. 2

[9] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *CVPR*, pages 4507–4515, 2017. 1

[10] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *ICML*, pages 2342–2350, 2015. 3

[11] Ioannis Karamouzas, Brian Skinner, and Stephen J Guy. Universal power law governing pedestrian interactions. *Physical Review Letters*, 113(23):238701, 2014. 2

[12] Kihwan Kim, Dongryeol Lee, and Irfan Essa. Gaussian process regression flow for analysis of motion trajectories. In *ICCV*, pages 1164–1171, 2011. 2

[13] Florin Leon and Marius Gavrilescu. A review of tracking, prediction and decision making methods for autonomous driving. *arXiv preprint arXiv:1909.07707*, 2019. 1

[14] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007. 2, 5

[15] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From goals, waypoints & paths to long term human trajectory forecasting. In *ICCV*, pages 15233–15242, 2021. 2

[16] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *ECCV*, pages 759–776, 2020. 1, 2, 5, 6

[17] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *CVPR*, pages 14424–14432, 2020. 1, 2

[18] Sajjad Mozaffari, Omar Y Al-Jarrah, Mehrdad Dianati, Paul Jennings, and Alexandros Mouzakitis. Deep learning-based vehicle behavior prediction for autonomous driving applications: A review. *IEEE T-ITS*, 23(1):33–47, 2020. 1

[19] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, pages 261–268, 2009. 2, 5

[20] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 3

[21] Amir Rasouli, Mohsen Rohani, and Jun Luo. Bifold and semantic reasoning for pedestrian behavior prediction. In *ICCV*, pages 15600–15610, 2021. 2

[22] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning Social Etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, pages 549–565, 2016. 2, 5

[23] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. SoPhie: An attentive gan for predicting paths compliant to social and physical constraints. In *CVPR*, pages 1349–1358, 2019. 2, 6

[24] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *ECCV*, pages 683–700, 2020. 2

[25] Nasim Shafiee, Taskin Padir, and Ehsan Elhamifar. Introvert: Human trajectory prediction via conditional 3d attention. In *CVPR*, pages 16815–16825, 2021. 2

[26] Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Fang Zheng, Nanning Zheng, and Gang Hua. Social interpretable tree for pedestrian trajectory prediction. In *AAAI*, pages 2235–2243, 2022. 1, 2, 6, 7, 8

[27] Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Mo Zhou, Zhenxing Niu, and Gang Hua. SGCN: Sparse graph convolution network for pedestrian trajectory prediction. In *CVPR*, pages 8990–8999, 2021. 1, 2, 6

[28] Jur Van Den Berg, Stephen J Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *ISRR*, pages 3–19, 2011. 2

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 3, 4, 7

[30] Chuhua Wang, Yuchen Wang, Mingze Xu, and David J Crandall. Stepwise goal-driven networks for trajectory prediction. *IEEE RA-L*, 7(2):2716–2723, 2022. 2

[31] Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian process dynamical models for human motion. *IEEE T-PAMI*, 30(2):283–298, 2007. 2

[32] Mingkun Wang, Xinge Zhu, Changqian Yu, Wei Li, Yuexin Ma, Ruochun Jin, Xiaoguang Ren, Dongchun Ren, Mingxu Wang, and Wenjing Yang. GANet: Goal area network for motion forecasting. In *ICRA*, pages 1609–1615, 2023. 5

[33] Chenxin Xu, Weibo Mao, Wenjun Zhang, and Siheng Chen. Remember intentions: Retrospective-memory-based trajectory prediction. In *CVPR*, pages 6488–6497, 2022. 1, 2, 5, 6, 7, 8

[34] Pei Xu, Jean-Bernard Hayet, and Ioannis Karamouzas. SocialVAE: Human trajectory prediction using timewise latents. In *ECCV*, pages 511–528, 2022. 1, 2, 5, 6, 7, 8

[35] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. XLNet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, pages 5753–5763, 2019. 3

[36] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *ECCV*, pages 507–523, 2020. 2, 3, 6

[37] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris Kitani. AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *ICCV*, pages 9793–9803, 2021. 1, 2, 3, 5, 6

[38] Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. In *ECCV*, pages 376–394, 2022. 2