STGlow: A Flow-based Generative Framework with Dual Graphormer for Pedestrian Trajectory Prediction

Rongqin Liang, *Student Member, IEEE,* Yuanman Li, *Member, IEEE,* Jiantao Zhou, *Senior Member, IEEE,* and Xia Li, *Member, IEEE*

基于 GAN 的方法可能无法完全支持数据分布，基于 CVAE 和扩散模型的方法优化的是观测对数似然的下界，可能导致学习到的分布偏离底层分布，这些方法常生成有偏差或不准确的轨迹。此外，基于图的方法可能存在节点特征过平滑问题，导致行人独特行为特征丢失

Abstract—The pedestrian trajectory prediction task is an essential component of intelligent systems. Its applications include but are not limited to autonomous driving, robot navigation, and anomaly detection of monitoring systems. Due to the diversity of motion behaviors and the complex social interactions among pedestrians, accurately forecasting their future trajectory is challenging. Existing approaches commonly adopt GANs or CVAEs to generate diverse trajectories. However, GAN-based methods do not directly model data in a latent space, which may make them fail to have full support over the underlying data distribution; CVAE-based methods optimize a lower bound on the log-likelihood of observations, which may cause the learned distribution to deviate from the underlying distribution. The above limitations make existing approaches often generate highly biased or inaccurate trajectories. In this paper, we propose a novel generative flow based framework with dual graphormer for pedestrian trajectory prediction (STGlow). Different from previous approaches, our method can more precisely model the underlying data distribution by optimizing the exact log-likelihood of motion behaviors. Besides, our method has clear physical meanings for simulating the evolution of human motion behaviors. The forward process of the flow gradually degrades complex motion behavior into simple behavior, while its reverse process represents the evolution of simple behavior into complex motion behavior. Further, we introduce a dual graphormer combining with the graph structure to more adequately model the temporal dependencies and the mutual spatial interactions. Experimental results on several benchmarks demonstrate that our method achieves much better performance compared to previous state-of-the-art approaches.

Index Terms—Generative flow, trajectory prediction, graph learning, attention mechanism, deep neural network.

I. INTRODUCTION

Trajectory prediction, as one of the most important future behavior modeling tasks, aims to predict the future trajectory based on the observed trajectory. It plays an important role in applications such as self-driving vehicles [1], autonomous navigation robots [2], anomaly behavior detection [3], [4], video surveillance [5]–[7] and so on. Despite that significant advances have been achieved recently [8]–[15], accurately predicting future trajectories of pedestrians remains challenging due to the inherent properties of pedestrians. First, due to the differences of human intent and unique behavior patterns, the future trajectories of pedestrians are full of diversity, even when they share the same historical trajectory. Second, influenced by surrounding agents, there are highly complex social interactions among pedestrians, which drive pedestrians to make decisions such as walking parallel, walking in groups, or changing direction / speed to avoid collisions. Faced with the challenge of diverse future trajectories, most previous works [8], [11]–[14], [16] applied generative models to model the multi-modality of human motion behaviors. For instance, some studies [8], [11], [13] employed generative adversarial networks (GANs) [17] to predict the distribution of future trajectories. However, GAN-based methods do not directly model data in a latent space, which may make them fail to have full support over the underlying data distribution, thus generating highly biased trajectories. In addition, the training process of GANs is often unstable due to the adversarial learning. Alternatively, some works exploited conditional variational auto-encoders (CVAEs) [16], [18]–[21] or diffusion model [14] to model the diversity of future trajectories. However, both of them optimize the variational lower bound on the log-likelihood of observations [22], which may cause the learned distribution to deviate from the underlying distribution. This means that using a lower bound criterion may yield a suboptimal solution with respect to the true log-likelihood, resulting in inaccurate future trajectories. Therefore, how to more precisely model the underlying distribution of pedestrian trajectories is very important for pedestrian trajectory prediction tasks.

To straightforwardly model the social interactions among pedestrians, some researches [9], [23]–[25] proposed to represent the social interactions between pedestrians utilizing the topology of graphs. However, graph-based methods may suffer from over-smoothing problems on node features [26], which means that when constructing graphs for crowded environments, the features of nodes will be smoothed by the aggregation of nodes. This may lead to the loss of unique behavior characteristics of pedestrians. Therefore, how to effectively model the social interactions while maintaining

Rongqin Liang, Yuanman Li and Xia Li are with the Guangdong Key Laboratory of Intelligent Information Processing, College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China. (Corresponding author: Yuanman Li, email: yuanmanli@szu.edu.cn.)

Jiantao Zhou is with the State Key Laboratory of Internet of Things for Smart City, and also with the Department of Computer and Information Science, University of Macau. e-mail: jtzhou@um.edu.mo.

unique behavioral features of pedestrians is still a challenge for pedestrian trajectory prediction tasks.

Faced with the above challenges on diverse trajectories and social interactions, in this paper, we propose a novel flow-based generative framework with dual graphormer for pedestrian trajectory prediction (STGlow). Firstly, to more precisely model the underlying distribution, we propose a generative flow framework with pattern normalization (Glow-PN) to produce multiple reasonable future trajectories. In contrast to previous GAN-based and CVAE-based models, our framework optimizes the exact log-likelihood of observations by mapping a complex data distribution into a simple and tractable one through a series of invertible transformations. Similar to the process of artists creating artworks, complex motion behaviors of pedestrians are not accomplished at one stroke, but based on simple motion behaviors (*e.g.*, every step of walking or every action) combined with individual behavior habits originating from various 'biases' such as people's walking habits, traffic awareness, and potential intentions. Hence, we propose to characterize the evolution of motion behavior from simple to complex by modeling the 'biases' progressively utilizing the generative flow with a series of simple and tractable functions. Secondly, to model the social interactions more effectively, we further propose a dual graphormer to extract the representations of motion behaviors and model the temporal dependencies and the mutual spatial interactions. The proposed dual graphormer combines graph structure with transformers, where the designed attention mechanism can adaptively focus on all other nodes. Our method not only enables intuitive and effective modeling of social interactions, but also greatly alleviates the problem of over-smoothing of nodes. Specifically, in the training phase, a Glow-PN is applied to learn the distribution of the motion behavior conditioned on the representation of social interactions. In the inference phase, simple behaviors are sampled from the standard normal distribution and formed into "evolved" representations of complex motion behaviors using the reverse process of Glow-PN conditioned on the representation of social interactions. These representations of complex motion behaviors are eventually decoded into predicted trajectories through a bidirectional trajectory prediction module. The main contributions of our work can be summarized as follows:

1) We present a novel diverse trajectory prediction framework based on generative flow, simulating the evolution of human motion behaviors from simple to complex. In contrast to previous approaches, our method can more precisely model the underlying distribution by optimizing the exact log-likelihood of motion behaviors. Besides, a pattern normalization is carefully developed to normalize the unique behavior pattern of pedestrians, greatly improving the prediction accuracy.

2) We further propose a dual graphormer to extract the representations of motion behaviors and model the social interactions in both temporal and spatial domains. Different from previous Transformer based methods, our dual graphormer combined with the graph structure more adequately models the temporal dependencies and the mutual spatial interactions.

3) The proposed framework achieves state-of-the-art performance on widely used pedestrians trajectory prediction benchmarks, providing a promising direction for generating diverse and reasonable trajectories.

The remainder of this paper is organized as follows. Section II gives a brief review of related works. Section III details our proposed STGlow for pedestrian trajectory prediction. Extensive experimental results are presented in Section IV, and we finally draw a conclusion in Section V.

II. RELATED WORKS

A. Pedestrian Trajectory Prediction

Traditional trajectory prediction methods mainly rely on designing handcrafted rules to model human interactions [27]–[31]. For instance, Social Force [27] introduced attractive and repulsive forces to avoid collisions. Although these methods demonstrate the importance of interaction modeling, they are limited by the handcrafted features and perform poorly in trajectory prediction.

With the great success of deep neural networks, the Recurrent Neural Network (RNN) and its variants are widely applied in trajectory prediction task [1], [32]–[34] and motion prediction task [35], [36], on the basis of their good performance on sequence learning [37]–[39]. Wherein, Social-LSTM [32] employed a Long Short-Term Memory (LSTM) to encode pedestrian trajectory and designed a Social Pooling to aggregate the global representation of neighboring pedestrians. To enhance the representation ability of social interaction features, many studies [1], [34], [40], [41] have followed this idea of transmitting information between pedestrians and proposed different effective message passing mechanisms. Although the RNN-based approaches approach the trajectory prediction task in a data-driven manner, they ignore the important fact that the future trajectories of pedestrians are full of diversity due to the differences of human intents and unique behavior patterns.

Besides, graph networks are utilized in various tasks such as action understanding [42], [43], recommendation systems [44], and text classification [45], due to their capability of modeling non-euclidean structured data. Recently, the intuitive modeling power of graph models has been applied to represent complex social interactions among pedestrians [9], [23], [24], [46]–[48]. For instance, the work GTPPO [24] explored a social graph attention module that combines specific obstacle avoidance experiences (OAEs) with the graph attention to capture pedestrians' social interactions. DMRGCN [48] proposed a disentangled multi-scale aggregation to better represent social interactions between pedestrians on a weighted graph. Though the topology of graphs seems to be a straightforward way to represent social interactions, graph-based methods may suffer from over-smoothing problems on node features [26], [48], which may lead to the loss of the unique behavior characteristics of pedestrians.

To generate diverse future trajectories, some researchers suggested employing generative models to model the diversity of human motion behaviors [8], [49], [50]. Part of these works were based on GANs [8], [11], [13]. Among them,

TABLE I
A SUMMARY OF THE DENOTATIONS.

Denotations	Descriptions	Denotations	Descriptions
$\mathcal{X}$	observed trajectories	R_i^t/SE	the embedding of the relative position
$\mathcal{Y}$	future trajectories	S_i^t/HE	the embedding of the relative steering angle
$TH_i^{1:T}$	the temporal embedding	SG	the spatial graphormer
G_{tmp}	the temporal graph	MB_{traget}	the representation of motion behavior
V_{tmp}	nodes of the G_{tmp}	ST_{traget}	the representation of social interaction
A_{tmp}	the adjacency matrix of the G_{tmp}	PN	pattern normalization
C_i^t/CE	the centrality embedding	BiD	the bidirectional decoder
P_i^t/PE	the positional embedding	$\hat{G}^{t_p}$	the goal of each pedestrian
TG	the temporal graphormer	$\hat{Y}_F^t$	the predicted forward trajectory
$SH_{1:N}^t$	the spatial-temporal embedding	$\hat{Y}_B^{t_b}$	the predicted backward trajectory
G_{spa}	the spatial graph	$\hat{Y}_{Both}^{t_b}$	the predicted bidirectional trajectory
V_{spa}	nodes of the G_{spa}	L_p	the loss of Glow-PN
A_{spa}	the adjacency matrix of the G_{spa}	L_{traj}	the loss of the bi-directional trajectory prediction

Social-GAN [8] applied GANs for the first time to generate diverse future trajectories and designed a pooling module to aggregate social interactions. Furthermore, TPNMS [11] proposed a temporal pyramid structure to model both global and local contexts of human motion behaviors. Another part of the works [12], [16], [18], [20], [21] applied CVAEs to explicitly encode the distribution of diverse future trajectories. For instance, Trajectron++ [16] utilized the latent variable framework of CVAEs to explicitly encode diversity and modeled social interactions in combination with a graph-structured recurrent model, while PECNet [18] embedded the distant trajectory endpoints into a latent space to assist in long-range diverse trajectory prediction. More recently, MID [14] devised a Transformer-based diffusion model for trajectory prediction with a reverse process of motion indeterminacy diffusion. Though previous generative models have achieved promising performance in modeling the diversity of human behaviors, these approaches still have inherent limitations, *e.g.*, methods based on GANs may not fully support the data distribution due to the lack of encoding latent variables, while methods based on CVAEs and diffusion models optimize a lower bound on the log-likelihood of observations. Such limitations could make them generate biased or inaccurate trajectories.

B. Normalizing Flow

Normalizing Flows (NFs) are invertible generative models that map complex data distributions to simple and tractable ones. Recently, NFs have been successfully applied to a variety of generation tasks such as image generation [22], [51]–[53], video generation [54], speech synthesis [55], [56]. For example, Glow [52] proposed an invertible 1×1 convolution for the generative flow and showed the efficiency of realistic-looking synthesis and manipulation of large images. WaveGlow [55] proposed a flow-based network for generating high-quality speech from mel-spectrograms. In our task, we propose a flow-based method for pedestrian trajectory prediction. Our proposed scheme has clear physical meanings to simulate the evolution of human motion behaviors.

C. Transformer

Transformer [57], which relies entirely on self-attention mechanisms to model global dependencies of the serialization

inputs, has recently made remarkable progress in a variety of natural language processing (NLP) tasks [58], vision tasks [59]–[62] and speech recognition [63]. For instance, in vision tasks [64], ViT [59] sequentialized the image into a series of tokens and modeled the global dependencies of the image through the Transformer encoder. More recently, some works [10], [12], [14] have applied Transformer to pedestrian trajectory prediction tasks. Among them, STAR [10] employed a temporal transformer and spatial transformer respectively to extract temporal dependencies and spatial interactions, while AgentFormer [12] exploited an agent-aware Transformer to learn representations from both temporal and spatial dimensions. Different from prior works, in this work, we devise a dual graphormer, which can more adequately model the temporal dependencies and the mutual spatial interactions.

III. PROPOSED APPROACH: STGLOW

The overall framework of STGlow model is illustrated in Fig. 1. It primarily consists of three components: 1) a dual graphormer to extract the representations of motion behaviors and model the temporal dependencies and the mutual spatial interactions; 2) a generative flow with pattern normalization (Glow-PN) to learn the underlying distribution of complex motion behaviors, conditioned on the social interactions; 3) a bi-directional trajectory prediction module to forecast diverse future trajectories. To facilitate check, we assemble the primary denotations and their accompanying explanations in Table I.

A. Problem Formulation

Given N pedestrians with observed trajectories $\mathcal{X} = \{X_1^{(-t_o+1:0)}, ..., X_N^{(-t_o+1:0)}\}$ from time steps T_{-t_o+1} to T_0 in the scene, the trajectory prediction algorithm aims to predict the future trajectories $\mathcal{Y} = \{Y_1^{(1:t_p)}, ..., Y_N^{(1:t_p)}\}$ of all pedestrians in the upcoming time steps T_1 to T_{t_p}, where $X_i^t = (x_i^t, y_i^t) \in \mathbb{R}^2$ is the position of the i-th pedestrian at the time step t. The trajectory prediction algorithm takes as input the observed trajectories with t_o time steps of all pedestrians in a scene, and aims to predict their future trajectories in the next t_p time steps by a model $f(\cdot)$, denoted by

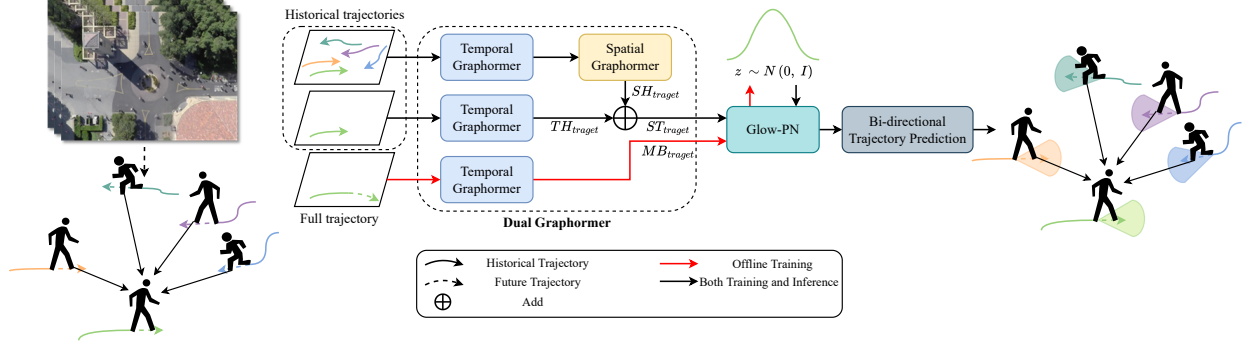$$\hat{\mathcal{Y}} = f(\mathcal{X}; W^*), \tag{1}$$

Fig. 1. The framework of our STGlow algorithm. STGlow primarily consists of a dual graphormer, a generative flow with pattern normalization (Glow-PN) and a bi-directional trajectory prediction module. 1) First, the full trajectory of the target pedestrian is encoded into the representation of complex motion behavior through the temporal graphormer, while the historical trajectories are encoded into the representation of social interaction through the temporal and spatial graphormer; 2) then, a Glow-PN is applied to learn the distribution of complex motion behaviors using a series of simple reversible transformation, conditioned on the social interactions; 3) simple behaviors sampled from the standard normal distribution evolve into complex motion behaviors through the reverse process of Glow-PN, which are eventually fed into the bi-directional trajectory prediction module to predict diverse future trajectories.
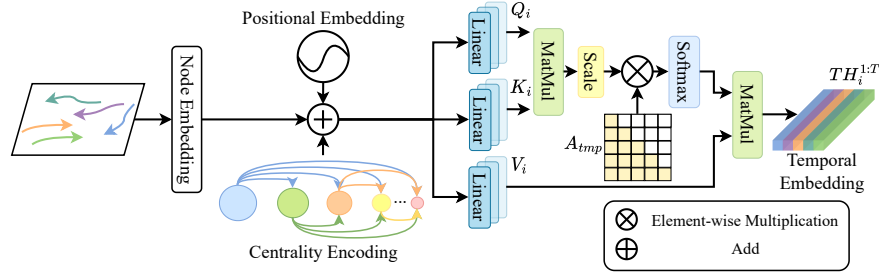

Fig. 2. Illustration of the temporal graphormer.

where $\hat{\mathcal{Y}}$ is the set of future trajectories predicted by $f(\cdot)$ and W^* represents the set of learnable parameters in the model.

For the sake of brevity, we hereafter drop the superscript when there is no ambiguity, *i.e.*, $X_i \triangleq X_i^{(-t_o+1:0)}$ and $Y_i \triangleq Y_i^{(1:t_p)}$. We further use X, Y to represent a generic history trajectory and the corresponding future trajectory, respectively.

B. Dual Graphormer

Influenced by surrounding agents, highly complex social interactions among pedestrians may force them to make decisions such as walking parallel, walking in groups, or changing direction and speed to avoid collisions. Obviously, such social interactions contain both temporal dependencies and spatial interactions, which are fundamentally important for accurately predicting trajectories. In this work, we design a dual graphormer to more adequately model the temporal dependencies and the mutual spatial interactions. As shown in Fig. 1, our dual graphormer primarily consists of two components: 1) a temporal graphormer, and 2) a spatial graphormer.

1) Temporal Graphormer: In this work, we decouple temporal dependencies into the behavior-independent temporal dependencies and behavior-dependent temporal dependencies. The former type of dependencies reveals the importance of each previous time step to the future trajectory. The other type of dependencies models the relationships among different previous motion behaviors across the temporal domain.

Assume that there are T time steps of each trajectory, denoted by $X_i^{1:T}$. As shown in Fig. 2, the temporal graphormer

takes $X_i^{1:T}$ as input and outputs a set of embeddings $TH_i^{1:T}$ with temporal dependencies, which we define as ***Temporal Embedding***. Specifically, we first construct a temporal graph by treating each time step as a node of the graph,

$$G_{tmp} = (V_{tmp}, A_{tmp}). \tag{2}$$

Here $V_{tmp} = \{V_t | t = 1, ..., T\}$ represents nodes of G_{tmp}. A_{tmp} is the adjacency matrix describing the temporal dependencies, i.e., the motion behavior at a given time can only be affected by previous motion states rather than future motion states. Based on this fact, we define A_{tmp} as

$$A_{tmp}^{ij} = \left\{ \begin{array}{ll} 1, & i \geq j \\ -inf, & i < j \end{array} \right., \ i, j \in [1, T]. \tag{3}$$

To model the behavior-independent temporal dependencies, we design a centrality encoding that learns a centrality embedding based on the length of the *influence duration* of each time step, which is formulated as

$$C_i^t = Linear\left(Deg\left(V_t\right), \Theta_C\right); t \in [1, T], \tag{4}$$

where $C_i^t \in \mathbb{R}^D$, Θ_C is the set of learnable parameters, and $Deg\left(V_t\right)$ calculates the outdegree of the node V_t. The centrality encoding adaptively explores the importance of different time steps based on the influence duration.

For behavior-dependent temporal dependencies, we first adopt a non-linear multi-layer perceptron (MLP) to embed the position X_i^t at each time step as

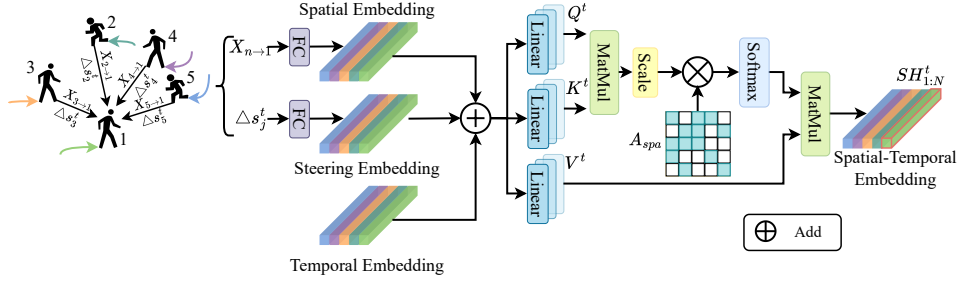$$H_i^t = MLP\left(X_i^t, \Theta_N\right), \tag{5}$$

Fig. 3. Illustration of the spatial graphormer.

where Θ_N contains the learnable parameters, and $H_i^t \in \mathbb{R}^D$. Besides, a positional embedding $P_i^t \in \mathbb{R}^D$ is applied to label the position of the motion state at each time step in a trajectory. In this paper, we encode the time step positions in a learnable way, as proposed in [65]. Then, we update the node embedding by

$$H_i^t := H_i^t + C_i^t + P_i^t. \tag{6}$$

According to the Transformer encoder [57], we further map H_i into three values

$$Q_i = H_i^{1:T} W_Q, \ K_i = H_i^{1:T} W_K, \ V_i = H_i^{1:T} W_V, \tag{7}$$

where W_Q, W_K, W_V are the parameters corresponding to the Query Q_i, Key K_i and Value V_i of the pedestrian i. The output of temporal graphormer can be further computed as

$$Att\left(Q_i, K_i, V_i\right) = Softmax\left(\frac{Q_i \left(K_i\right)^T}{\sqrt{d_k}} \odot A_{tmp}\right) V_i, \tag{8}$$

where A_{tmp} is the adjacency matrix, $\odot$ is the operation of dot product, and d_k is the dimension of Q_i. Eq. (8) characterizes the behavior-dependent temporal dependencies, which builds the relationships among different motion behaviors across the temporal domain. For brevity, we write the temporal graphormer as

$$TH_i^{1:T} = TG\left(X_i^{1:T}; \Theta_{tg}\right), \tag{9}$$

where $TH_i^{1:T}$ is the *Temporal Embedding* of the temporal graphormer, and Θ_{tg} contains learnable parameters. By resorting to the temporal graphormer, both the behavior-independent temporal dependencies and behavior-dependent temporal dependencies can be characterized.

2) Spatial Graphormer: Relative positions of other pedestrians are crucial for a target pedestrian to make decisions, such as adjusting velocity or direction to avoid collisions. However, considering only the relative positions to the neighbors is not sufficient for making a reasonable decision. For example, when a neighbor in the rear is walking toward the opposite direction from the target pedestrian, the target pedestrian generally does not need to make adjustments, despite their close proximity. In reality, the relative steering angle of a neighbor to the target pedestrian and the walking direction of the target pedestrian are also key elements that drive pedestrians to change their motion states.

To fully model the mutual spatial interactions between pedestrians, in this work, we propose a spatial graphormer by utilizing the relative motion states of the neighbors (*i.e.*, the

relative position and the relative steering angle to the target pedestrian) and the walking direction of the target pedestrian. Specifically, we take pedestrians in the scene as nodes and construct an undirected graph based on the walking direction of the target pedestrian and the relative motion states of the neighbors. First, at time step t, we can construct the spatial graph as

$$G_{spa} = (V_{spa}, A_{spa}), \tag{10}$$

where V_{spa} represents the nodes of G_{spa}, and A_{spa} is the adjacency matrix of G_{spa} describing the mutual spatial relationships among pedestrians. To better capture spatial interactions, we endow each node $V_i^t \in V_{spa}$ with the information of both relative positions and relative steering angles. Namely,

$$V_i^t = R_i^t + S_i^t + TH_i^t, \tag{11}$$

where TH_i^t denotes the temporal embedding of pedestrian i at time step t, R_i^t and S_i^t represent the embedding of the relative position and relative steering angle of pedestrian i to the target pedestrian, respectively. As shown in Fig. 3, we adopt a single layer MLP with ReLU activation to embed the relative position and relative steering angle of neighbors as

$$R_j^t = MLP\left(x_{j\rightarrow i}^t, y_{j\rightarrow i}^t; \Theta_P\right), \\ x_{j\rightarrow i}^t = x_j^t - x_i^t; \ y_{j\rightarrow i}^t = y_j^t - y_i^t, \tag{12}$$

$$S_j^t = MLP(\triangle s_j^t; \Theta_S), \\ \triangle s_j^t = \frac{\triangle X_i^t \cdot \triangle X_j^t}{|\triangle X_i^t||\triangle X_j^t|}, \tag{13}$$

where $(x_{j\rightarrow i}^t, y_{j\rightarrow i}^t)$ is the relative position of neighbor j to the target pedestrian i at time step t, $\triangle s_j^t$ denotes the corresponding relative steering angle, $\triangle X_i^t = (\triangle vx_i^t, \triangle vy_i^t) = (x_i^t - x_i^{t-1}, y_i^t - y_i^{t-1})$ denotes the walking direction of pedestrian i at time step t, and Θ_P, Θ_S are the parameters of MLP. We refer to R_j^t as the spatial embedding, and S_j^t as the steering embedding. Note that we do not employ positional encoding when modeling spatial interactions since there is no natural order for pedestrians in the scene.

Intuitively, people commonly pay little attention to pedestrians outside their field of vision. For simplicity, we set the maximum binocular field of view to $180°$ in the horizontal position for a pedestrian. Based on this fact, we design the adjacency matrix A_{spa} as

$$A_{spa}^{ij} = \begin{cases} 1, \ if \ x_{j\rightarrow i}^t \cdot \triangle vx_i^t \geq 0 \ and \ y_{j\rightarrow i}^t \cdot \triangle vy_i^t \geq 0, \\ -inf, \qquad\qquad Otherwise, \end{cases} \tag{14}$$

where $(\triangle vx_i^t, \triangle vy_i^t)$ is the walking direction of pedestrian i along x and y axes, and $(x_{j \to i}^t, y_{j \to i}^t)$ is the relative position of neighbor j to pedestrian i. A_{spa}^{ij} characterizes the mutual spatial relationships among pedestrians.

Similar to the temporal graphormer, the output of the spatial graphormer can be further computed as,

$$Q^t = SH_{1:N}^t W_{Q'}, \ K^t = SH_{1:N}^t W_{K'}, \ V^t = SH_{1:N}^t W_{V'},$$

$$Att\left(Q^t, K^t, V^t\right) = Softmax\left(\frac{Q^t\left(K^t\right)^T}{\sqrt{d_k}} \odot A_{spa}\right) V^t, \tag{15}$$

where N is the number of pedestrians in the scene, $W_{Q'}, W_{K'}, W_{V'}$ are the parameters. For brevity, we write the spatial graphormer as

$$SH_{1:N}^t = SG\left(X_{1:N}^{t-1:t}, TH_{1:N}^t; \Theta_{sh}\right), \tag{16}$$

where $SH_{1:N}^t$ is the output of the spatial graphormer, which we define as **Spatial-Temporal Embedding**. Θ_{sh} is the set of parameters. Similar to [57], the multi-head attention mechanism is employed in our framework. Note that different from graph-based methods, our dual graphormer takes the advantage of the self-attention mechanism to avoid direct aggregation of connected nodes, which could greatly alleviate the over-smoothing problem. Such a conclusion has been carefully justified in [26].

3) Proposed Dual Graphormer in Trajectory Prediction: In order to model the behavior pattern of pedestrians, we first apply the Dual Graphormer to extract the deep representations of **motion behaviors** and **social interactions**. As shown in Fig. 1, we primarily extract representations of two types of inputs, *i.e.*, the full trajectory and the observed historical trajectory. Note that the full trajectories are used only in the training phase.

The full trajectory reflects the motion behavior of each pedestrian in a time period, such as where to go and how to go. In our STGlow model, we define the representation of the full trajectory as the **motion behavior** of each pedestrian, which can be formulated as

$$MB_{target} = TG\left(Concat\left(X_{target}, Y_{target}\right), \Theta_{tg}\right), \tag{17}$$

where $TG(\cdot)$ is the temporal graphormer.

Besides, the observed historical trajectories of all pedestrians in the scene are encoded via the temporal and spatial graphormer to extract the **social interactions** of the target pedestrian, which can be written as

$$\begin{aligned} TH_{1:N} &= TG\left(X_{1:N}; \Theta_{tgh}\right), \\ SH_{target} &= SG\left(X_{1:N}^{-1:0}, TH_{1:N}^0; \Theta_{sgh}\right), \\ TH_{target} &= TG\left(X_{target}; \Theta_{tgy}\right), \\ ST_{target} &= TH_{target} + SH_{target}. \end{aligned} \tag{18}$$

where Θ_{tgh}, Θ_{sgh} and Θ_{tgy} are parameters. So far, we have obtained the representation MB_{target} of **motion behavior** of the target pedestrian and the mutual **social interaction** ST_{target}. In the inference phase, we first predict the latent motion behavior MB_{target} conditioned on the ST_{target}. The MB_{target} is then fed into a decoder to generate future trajectories.
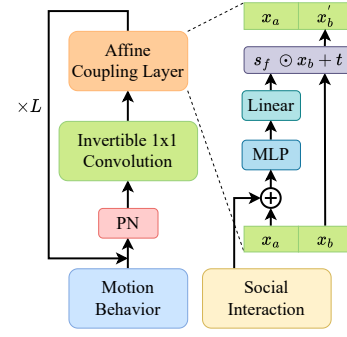

Fig. 4. Illustration of the proposed generative flow with Pattern Normalization (Glow-PN).

C. Proposed Generative Flow with Pattern Normalization (Glow-PN)

Due to the differences of human intent and unique behavior patterns, the motion behaviors of pedestrians are of high diversity. Namely, the future trajectories could be very different given the same historical trajectory. Thus, learning the underlying distribution of motion behaviors conditioned on social interactions is important to predict the diverse trajectories of pedestrians. Existing approaches commonly use GANs or CVAEs to generate diverse trajectories, where their limitations have been carefully discussed in the first section.

In our work, we simulate the process of degradation and evolution between the complex motion behavior and a simple behavior as follows:

$$MB_{target} \xleftrightarrow{f_1} h_1 \xleftrightarrow{f_2} h_2 \cdots \xleftrightarrow{f_k} z, \tag{19}$$

where $\{f_1, f_2, ..., f_k\}$ denote a set of *invertible* transformations to simulate the degrading and evolving process, and $h_1, ..., h_{k-1}$ denote the intermediate motion behaviors. We can see that the forward process of formula (19) gradually degrades the complex motion behavior MB_{target} into a simple behavior z, while its reverse process represents the evolution of a simple behavior to the complex motion behavior.

In reality, the complex motion behavior MB_{target} of pedestrians is not accomplished at one stroke, but based on simple motion behaviors z (*e.g.*, every step of walking or every action) combined with individual behavior habits which may originate from various aspects such as people's walking habits, traffic awareness, and potential intentions. For the sake of simplicity, we assume that z follows a standard normal distribution, *i.e.*,

$$z \sim \mathcal{N}\left(z; 0, I\right), \tag{20}$$

where I is an identity matrix. Letting $x \triangleq MB_{target}$, the log-likelihood of complex motion behavior can be written as:

$$\log p_\theta(x) = \log\left(p_\theta(z)\left|\det\left(dz/dx\right)\right|\right) \tag{21}$$

$$= \log p_\theta(z) + \sum_{i=1}^{K} \log\det\left(dh_i/dh_{i-1}\right)| \tag{22}$$

$$= \log p_\theta(z) + \sum_{i=1}^{K} \log\det\left(J\left(f_i^{-1}(x)\right)\right)|. \tag{23}$$

The equation (21) holds because $\int_x p_\theta(x)\,dx = \int_z p_\theta(z)\,dz$, and the equation (23) holds because $h_i = f_i^{-1}(h_{i-1})$. The first term in formula (23) is the log-likelihood of the standard normal distribution and the scalar value $\log|\det(J(f_i^{-1}(x)))|$ is the logarithm of the absolute value of the determinant of the Jacobian matrix $J(f_i^{-1}(x))$. This value reflects the transformation from h_{i-1} to h_i of the motion behavior under the transformation f_i. Then, our framework optimizes the parameters by minimizing the negative log-likelihood function, where the loss is

损失1：负对数似然函数

$$L_p = \min \ -\sum_{i=1}^{N} \log \ p_\theta(x_i). \tag{24}$$

In contrast to previous approaches, our method can more precisely model the underlying data distribution by optimizing the exact log-likelihood of motion behaviors as shown in (24). Obviously, how to design the invertible transformation functions $f_1, ..., f_k$ is essential for the optimization of (24). We should bear in mind that $f_1, ..., f_k$ should be differentiable thus allowing the end-to-end training.

In our work, we employ Glow [52] to learn the invertible transformations of motion behaviors from complex to simple. As shown in Fig. 4, for the forward process, we take ***motion behavior*** MB_{target} as input of Glow-PN. After several "steps of flow", the complex ***motion behavior*** is degraded into a simple behavior that can be represented by a simple distribution. A step of flow here consists of a pattern normalization (PN), an invertible 1×1 convolution, and an affine coupling layer, described below.

1) Pattern Normalization (PN): Considering the fact that diverse motion behaviors share the same behavior pattern (*e.g.*, the individual walking habits), we propose a pattern normalization (PN) for trajectory prediction as shown in Fig. 5. Different from the Actnorm adopted in the original Glow, PN performs normalization for multiple motion behaviors jointly at the sample axis. As will be shown in the experiments, our proposed PN greatly improves the prediction performance.

Specifically, the forward function, reverse function and log-determinant of our proposed PN are calculated as:

$$Function : y = s_g \odot x + b,$$
$$Reverse \ Function : x = (y - b)/s_g, \tag{25}$$
$$Log \ determinant : \log|s_g|,$$

where x indicates the input of PN, and y signifies its output. Both x and y are tensors of shape $(K \times C \times 1)$ with K motion behaviors and channel dimension C. $s_g, b \in \mathbb{R}^{K \times C}$ are learnable scales and bias parameters whose initialization depends on the input data, so that the data has zero mean and unit variance after PN. After initialization, the scale and bias are treated as regular trainable parameters that are independent of the data. Note that, different from Layer Normalization [66], to ensure that the transformation f is invertible, we design the forward and reverse process of normalization in PN. The forward process normalizes the behavior pattern of each pedestrian to promote model convergence, while the reverse process de-normalizes multiple motion behaviors to restore its original motion characteristics.
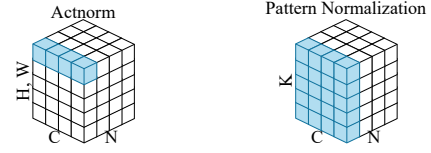

Fig. 5. In ActNorm, (H, W) as the size of the feature map and C as the channel axis, while in PN, K as the sample axis and C as the feature axis. N as the batch axis. The pixels in blue are normalized by the same learnable scales and bias.

2) Invertible 1×1 Convolution: In order that all channels of input in the forward transformation can be updated in subsequent coupling layers, we following Glow apply an invertible 1×1 convolution layer before coupling layers. The weights W of convolution are initialized as a random rotation matrix and hence invertible. Thus, the log-determinant of this transformation is easy to compute:

$$f_{conv}^{-1} = Wy, \tag{26}$$
$$\log|\det(J(f_{conv}^{-1}(y)))| = \log|\det(W)|, \tag{27}$$

where the log-determinant starts at zero and after one SGD step, the values start to diverge from zero.

3) Affine Coupling Layer: Generally, computing the determinants of high-dimensional Jacobian and large matrices is very expensive. Following Glow, we reduce the complexity by designing tractable and flexible invertible transformation. Specifically, we introduce an affine coupling layer, which can efficiently compute forward function, reverse function and log-determinant.

$$x_a, x_b = split(x_{affine}), \tag{28}$$
$$(\log s_f, t) = Linear(MLP(concat(x_a, ST_{target}))), \tag{29}$$
$$x_b' = s_f \odot x_b + t, \tag{30}$$
$$x' = concat(x_a, x_b'), \tag{31}$$

where $x_{affine} \in \mathbb{R}^{1 \times C \times K}$ represents the input of affine coupling layer, the $split(\cdot)$ function splits x into two havles along the channel dimension, while the $concat(\cdot)$ operation performs the restore operation. Since the inputs x_a of formula (29) remains unchanged during the affine transformation operation, formula (29) can be arbitrary transformation. Accordingly, in the reverse process of the affine coupling layer, s_f and t can be obtained from the output x_a through formula (29), and x_b' can be obtained by the reverse of x_b. Note that our affine coupling layer is conditioned on the social interaction ST_{target} to establish the mapping of complex to simple motion behaviors.

As we can see, only the s_f term changes the volume of the mapping in the affine coupling layer and adds a change of variables term to the loss. The log-determinant of the affine coupling layer and the final likelihood are hence computed as follows:

$$\log|\det(J(f_{coupling}^{-1}(x)))| = \log|s_f|. \tag{32}$$

$$\log p_\theta(x) = \log p_\theta(z) + \sum_{j=0}^{n} \left(\log|s_g^j| + \log\det|W^j| + \log|s_f^j|\right), \tag{33}$$
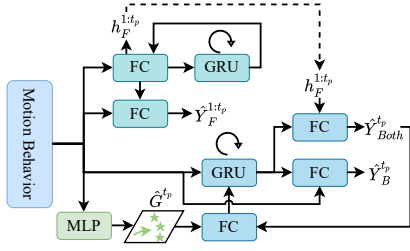
Fig. 6. Illustration of the bi-directional trajectory prediction module.

where $p_\theta(z) = \mathcal{N}(z; 0, I)$, and its log-likelihood can be easily computed as: $-z(x)^T z(x)/2\sigma^2$, where $\sigma = I$. So far, we have eventually completed the design of the invertible transformation of "step of flow". Note that, different from CVAE-based methods, our method optimizes the exact log-likelihood of motion behaviors.

D. Bi-directional Trajectory Prediction

Due to the invertible design of Glow-PN, a simple behavior drawn from the standard normal distribution can be directly evolved into a complex motion behavior MB_{target} through the reverse process of Glow-PN, conditioned on the social interaction ST_{target}. Then, we feed the obtained MB_{target} into the decoder to predict the future trajectory.

1) Bi-directional Decoder with Goal Estimation: To alleviate the error accumulation issue caused by the recurrent neural network in trajectory prediction, as shown in Fig. 6, we adopt the decoder proposed in Bitrap [20], which is designed in a bi-directional manner. We additionally predict and supervise the forward and backward future trajectory to strengthen the representation learning during bidirectional prediction. Specifically, we first predict the goal of each pedestrian as

$$\hat{G}^{t_p} = MLP\left(MB'_{target}; \Theta_g\right), \quad (34)$$

where MB'_{target} is the representation of complex motion behaviors evolved by the reverse process of Glow-PN from simple behaviors. Then, the forward trajectory $\hat{Y}_F^t$ is predicted as

$$
\begin{aligned}
f_i^{t-1} &= MLP\left(f_h^{t-1}; \Theta_{fi}\right), \\
f_h^t &= GRU\left(f_h^{t-1}, f_i^{t-1}; \Theta_{ff}\right), \\
\hat{Y}_F^t &= FC\left(concat\left(f_i^t, MB'_{target}\right); \Theta_{fy}\right),
\end{aligned}
\quad (35)
$$

where t from 1 to t_p represents the time step of the forward prediction, and $f_h^0 = MLP\left(MB'_{target}; \Theta_{fh}\right)$. Further, the backward prediction $\hat{Y}_B^{t_b}$ is formulated as

$$
\begin{aligned}
b_h^{t_b} &= GRU\left(b_i^{t_b+1}, b_h^{t_b+1}; \Theta_{bh}\right), \\
\hat{Y}_B^{t_b} &= FC\left(concat\left(b_h^{t_b}, MB'_{target}\right); \Theta_{by}\right),
\end{aligned}
\quad (36)
$$

where t_b from t_p-1 to 1 denotes the time step of the backward prediction, and $b_h^{t_p} = MLP\left(MB'_{target}; \Theta_{bh}\right)$. Last, the

bidirectional prediction $\hat{Y}_{Both}^{t_b}$ considering both forward and backward trajectories is predicted as

$$
\begin{aligned}
\hat{Y}_{Both}^{t_b} &= FC\left(concat\left(b_h^{t_b}, f_i^{t_b}\right); \Theta_{both}\right), \\
b_i^{t_b} &= MLP\left(\hat{Y}_{Both}^{t_b}, \Theta_{bi}\right),
\end{aligned}
\quad (37)
$$

where $b_i^{t_p} = MLP\left(\hat{G}^{t_p}, \Theta_{bi}\right)$. Besides, Θ_* indicates the set of parameters. We supervise the predicted forward, backward and bidirectional trajectories as

$$
\begin{aligned}
L_{traj} = \min_{k \in K} \alpha \|G_k^{t_p} - \hat{G}_k^{t_p}\| + \min_{k \in K} \sum_{t=1}^{t_p} (\lambda_1 \|Y_k^t - \hat{Y}_F^t\| \\
+ \lambda_2 \|Y_k^t - \hat{Y}_B^t\| + \lambda_3 \|Y_k^t - \hat{Y}_{Both}^t\|),
\end{aligned}
\quad (38)
$$

where K represents the number of samples of future trajectories, and α, λ_1, λ_2, λ_3 are the coefficients that balance different losses. Combined with the log-likelihood of the motion behaviors, our final loss is formulated as

$$L_{total} = L_p + \sum_{i=1}^{N} L_{traj}. \quad (39)$$

2) Inference: In the inference phase, we can easily generate the diverse trajectories by simple behaviors sampled from the standard normal distribution through the reverse process of Glow-PN conditioned on social interactions.

IV. EXPERIMENTS

In this section, we evaluate the performance of our proposed STGlow, which is implemented using the PyTorch framework. All the experiments are performed on Ubuntu 18.04 with an NVIDIA 3090 GPU. Our source code and trained models will be publicly available upon acceptance.

A. Implementation Details

The dimension of the node embedding is set as 256, and the number of coupling layers and invertible 1×1 convolutions are empirically set as 16. We also output 64 of the channels after every 4 coupling layers. In the training phase, the number of samples of predicted trajectories is set to 20 and the coefficients of the final loss are set to be 1, 0.25, 0.25, 0.5, respectively. We adopt Adam algorithm [68] to optimize the loss function (39) and train our network with the following hyper-parameter settings: batch size is 128; learning rate is 1e-3; betas are 0.9 and 0.999; weight decay is 1e-6 and the number of epochs is 400.

B. Datasets and Metrics   数据集

Datasets: We evaluate our method on two benchmark public pedestrian trajectory prediction benchmarks including ETH/UCY dataset [31], [69] and Stanford Drone Dataset (SDD) [70]. The ETH and UCY dataset group consists of 5 video sequences: ETH & HOTEL (from ETH) and UNIV, ZARA1, & ZARA2 (from UCY). All trajectories are converted to world coordinates, so the results we report are in meters. SDD comprise of more than 11000 unique pedestrians across

TABLE II
QUANTITATIVE RESULTS ON THE ETH/UCY DATASET WITH BEST-OF-20 STRATEGY IN ADE/FDE METRIC. LOWER IS BETTER.

Datasets / Methods	Sampling	ETH		Hotel		Univ		Zara1		Zara2		AVG	
		ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE	ADE	FDE
S-GAN [8]	20	0.87	1.62	0.67	1.37	0.76	1.52	0.35	0.68	0.42	0.84	0.61	1.21
Social-STGCNN [9]	20	0.64	1.11	0.49	0.85	0.44	0.79	0.34	0.53	0.30	0.48	0.44	0.75
GTPPO [24]	20	0.63	0.98	0.19	0.30	0.35	0.60	0.20	0.32	0.18	0.31	0.31	0.50
TPNMS [11]	20	0.52	0.89	0.22	0.39	0.55	1.13	0.35	0.70	0.27	0.56	0.38	0.73
TPNSTA [13]	20	0.51	0.87	0.22	0.39	0.52	1.09	0.34	0.68	0.26	0.54	0.37	0.71
STAR [10]	20	0.56	1.11	0.26	0.50	0.52	1.15	0.41	0.90	0.31	0.71	0.41	0.87
Trajectron++ [16]	20	0.39	0.83	0.12	0.21	0.20	0.44	0.15	0.33	0.11	0.25	0.19	0.41
AgentFormer [12]	20	0.45	0.75	0.14	0.22	0.25	0.45	0.18	0.30	0.14	0.24	0.23	0.39
PECNet [18]	20	0.54	0.87	0.18	0.24	0.35	0.60	0.22	0.39	0.17	0.30	0.29	0.48
SGCN [23]	20	0.63	1.03	0.32	0.55	0.37	0.70	0.29	0.53	0.25	0.45	0.37	0.65
DMRGCN [48]	20	0.60	1.09	0.21	0.30	0.35	0.63	0.29	0.47	0.25	0.41	0.34	0.58
Y-Net+TTST [19]	**10000**	**0.28**	**0.33**	0.10	**0.14**	0.24	0.41	0.17	0.27	0.13	0.22	0.18	0.27
CAGN [67]	20	0.41	0.65	0.13	0.23	0.32	0.54	0.21	0.38	0.16	0.33	0.25	0.43
MID [14]	20	0.39	0.66	0.13	0.22	0.22	0.45	0.17	0.30	0.13	0.27	0.21	0.38
BiTraP [20]	20	0.37	0.69	0.12	0.21	0.17	0.37	0.13	0.29	0.10	0.21	0.18	0.35
STGlow (ours)	20	**0.31**	**0.49**	**0.09**	**0.14**	**0.16**	**0.33**	**0.12**	**0.24**	**0.09**	**0.19**	**0.15**	**0.28**
Improvement	20	16%	25%	25%	33%	6%	11%	8%	17%	10%	10%	13%	19%

评价指标

20 top-down scenes captured on the stanford university campus in bird's eye view containing several moving agents like humans and vehicles. We use the standard test train split as used in [18] and other previous works.

Metrics: For the sake of fairness, we adopt the evaluation metrics Average Displacement Error (ADE) and Final Displacement Error (FDE) which are commonly used in literature [8], [9], [12], [32]. ADE computes the average $\ell 2$ distance between the predictions and the ground truth future while FDE computes the $\ell 2$ distance between the predicted and ground truth at the last observed point. The number of observed time steps is 8 (3.2 seconds) of each person and the upcoming trajectory of 12 time steps (4.8 seconds) is used to predict. For the ETH/UCY dataset, we use the widely adopted leave-one-out approach evaluation methodology such that we train our model on four scenes and test on the remaining one [8], [11], [13]. Considering the diversity of future trajectories, we use the Best-of-K strategy to compute the final ADE and FDE with $K = 20$.

C. Baselines

We compare with the following baselines including previous state-of-the-art methods:

GAN-based methods: S-GAN [8]: a model that employs GAN with a global pooling module to generate diverse pedestrian trajectories; TPNMS [11] and TPNSTA [13]: methods based on the temporal pyramid network to model global and local context of motion behavior, the latter further designs the spatial-temporal attention mechanism.

CVAE-based methods: Trajectron++ [16]: a recurrent graph based forecasting model incorporating dynamic constrains; PECNet [18]: a goal conditioned trajectory prediction network; BiTraP [20]: a goal-conditioned bidirectional trajectory prediction method based on the CVAEs.

Graph-based methods: Social-STGCNN [9]: an approach that models the social behavior of pedestrians using a graph; SGCN [23]: an approach that models the sparse directed interaction with a sparse directed spatial graph; DMRGCN [48]: a model that introduce a disentangled multi-scale

aggregation to represent social interactions; GTPPO [24]: a Graph-based Trajectory Predictor with Pseudo-Oracle.

Transformer-based methods: AgentFormer [12]: a transformer-based approach that models the time and spatial dimensions simultaneously; STAR [10]: a spatial-temporal graph transformer framework.

Other methods: Y-Net [19]: method based on position and visual image information. 'TTST' stands for the Test-Time Sampling Trick (TTST) in post-processing, which first samples 10000 trajectories and then clusters them into 20 trajectories; CAGN [67]: a complementary attention gated network for pedestrian trajectory prediction; SIT [71]: a tree-based method for pedestrian trajectory prediction; MID [14]: a method based on the diffusion model.

D. Quantitative Analysis

We quantitatively compare our ***STGlow*** with a wide range of current methods. Table II compares our method with existing algorithms on the ETH/UCY dataset. Besides the performance on each dataset, we report the average results for each method in the last two columns. Noted that, we also report the sampling number since adding the sampling number can effectively promote the performance [14]. Based on the results, we draw the following conclusions:

- In general, with the same sampling number of 20, our method STGlow outperforms all the previous approaches in terms of ADE and FDE for all datasets. The last row of Table II shows the performance improvement of our method over the previous best methods (marked with red underline), where our method improves the ADE/FDE metrics by an average of 13%/19% on ETH/UCY datasets, respectively.
- Compared with the GAN-based method [8], [11], [13], our method achieves significant performance gains on ADE and FDE metrics. For example, Our method achieves 59% and 61% relative improvements in average ADE and FDE metrics over the GAN-based method TPNSTA. In addition, compared with the methods based on CVAEs [12], [16], [18], [20] and diffusion model

TABLE III
QUANTITATIVE RESULTS ON THE SDD DATASET WITH BEST-OF-20 STRATEGY IN ADE/FDE METRIC. ∗ MEANS THE RESULTS ARE REPRODUCED BY [14] WITH THE OFFICIAL RELEASED CODE. LOWER IS BETTER.

Methods	Sampling	ADE	FDE
S-GAN [8]	20	27.23	41.44
PECNet [18]	20	9.96	15.88
Y-Net + TTST [19]	10000	7.85	11.85
Y-Net* [19]	20	8.97	14.61
GTPPO [24]	20	10.13	15.35
Trajectron++* [16]	20	8.98	19.02
SIT [71]	20	8.59	15.27
MID [14]	20	7.61	14.30
STGlow (ours)	20	**7.20**	**11.20**

TABLE IV
THE ADE AND FDE PERFORMANCE OF VARIANTS ON SDD DATASET. ◯ AND △ INDICATE GRU AND TRANSFORMER, RESPECTIVELY

Components				Variants
SG	TG	PN	BiD	ADE/FDE
✓	✓	✓		7.30/11.38
×	◯		✓	8.80/14.75
×	◯	✓	✓	7.45/11.88
△	△	✓	✓	7.33/11.56
×	✓	✓	✓	7.39/11.70
✓	◯	✓	✓	7.53/11.84
✓	✓		✓	8.60/14.35
✓	✓	✓	✓	**7.20/11.20**

[14], our method also achieves a greater performance improvement due to the optimization of the exact log-likelihood of motion behavior rather than the variational lower bound. For instance, our method improves the average ADE and FDE metrics by 17% and 20% respectively compared with the best CVAE-based method BiTraP [20], and 29% and 26% respectively compared with MID [14].

- Compared with the graph-based approach [9], [23], [24], [48] that models social interactions in an intuitive way, we integrate the graph structure with the Transformer structure in modeling social interactions and achieve significant performance improvements. For example, compared with DMRGCN [48], our STGlow improves 56% and 52% in average ADE and FDE metrics respectively.
- Despite the unfair experimental settings, our method still achieves 17% and 4% performance improvement over Y-Net+TTST on average ADE/FDE metrics respectively.

We can draw similar conclusions above on SDD dataset, as shown in Table III, which indicates that the proposed STGlow has better generalization performance. It is worth noting that STGlow achieved a relative improvement of 20% and 23% in ADE and FDE metrics compared to Y-Net [19] without using 'TTST' post-processing. Furthermore, compared with the current state-of-the-art method MID [14], our method achieves the best performance on ADE and FDE metrics.

E. Qualitative Analysis

In this subsection, we present visual examples to further illustrate the ability of our STGlow to fully explore complex social interactions between pedestrians and generate reasonable and diverse future trajectories.

1) Results in different scenarios: As can be drawn from Table II, methods based on CVAEs perform significantly better than methods based on GANs. Thus, we compare the most-likely predictions between STGlow and the previous state-of-the-art CVAE-based method, BiTraP [20], qualitatively on all five scenes of the ETH/UCY dataset.

Overall, as shown in Fig. 7, our prediction results are significantly closer to the ground truth trajectory compared with BiTraP, regardless of simple scenarios or scenarios with complex interactions. Specifically, the first row illustrates simple motion behaviors in five scenarios, including uniform walking and simple interactions. In these scenarios, the trajectories predicted by our method are significantly closer to the ground truth trajectories, since we more precisely model the underlying distribution by optimizing the exact log-likelihood of motion behaviors. The second and third rows show more complex motion behaviors, including slowing down, speeding up, making-a-turn, avoiding collision, and complex interactions. In these scenarios, compared to BiTraP, our method is able to make predictions that are more consistent with the laws of human motion, as we intuitively model the evolution of human motion behavior from simple to complex. For example, in the second and third rows of ETH and HOTEL, BiTraP does not handle well with slowing down and speeding up, and even predicts the future trajectory of possible collisions for parallel pedestrians, whereas our method does a good job of forecasting these challenging motion behaviors. Besides, our method adequately models temporal dependencies and the mutual spatial interactions to predict a more reasonable future trajectory. For instance, in the second row of UNIV, our method can accurately predict the making-a-turn behavior for parallel pedestrians in complex scenarios. In contrast, BiTraP's prediction results show certain deviations in both velocity and direction. Similarly, in the third row, our method can accurately deal with the situation of avoiding collision, because we intuitively model social interactions in both temporal and spatial domain. A similar conclusion can be drawn from the second and third rows of ZARA1 and ZARA2.

2) Results of diverse predictions: We further investigate the ability of our method to generate diverse predictions by comparing the case with and without the dual graphormer. As shown in the walkable area predicted in Fig. 8, regardless of whether there is social interaction modeling or not, our generative flow-based method can generate diverse and reasonable future trajectories well. Notably, compared with the model without the dual graphormer (as shown in the second row), our STGlow not only avoids collisions to a certain extent but also predicts more concentrated walking areas. This means that our method STGlow considering social interaction can better measure the diversity and stability of forecasted future trajectories.

F. Ablation Experiments

In this subsection, we conduct ablation experiments to investigate the effectiveness of each key component including spatial graphormer, temporal graphormer, PN in Glow, and bidirectional decoder.

1) Componets of our architecture: We first explore the impact of each component of our architecture, including
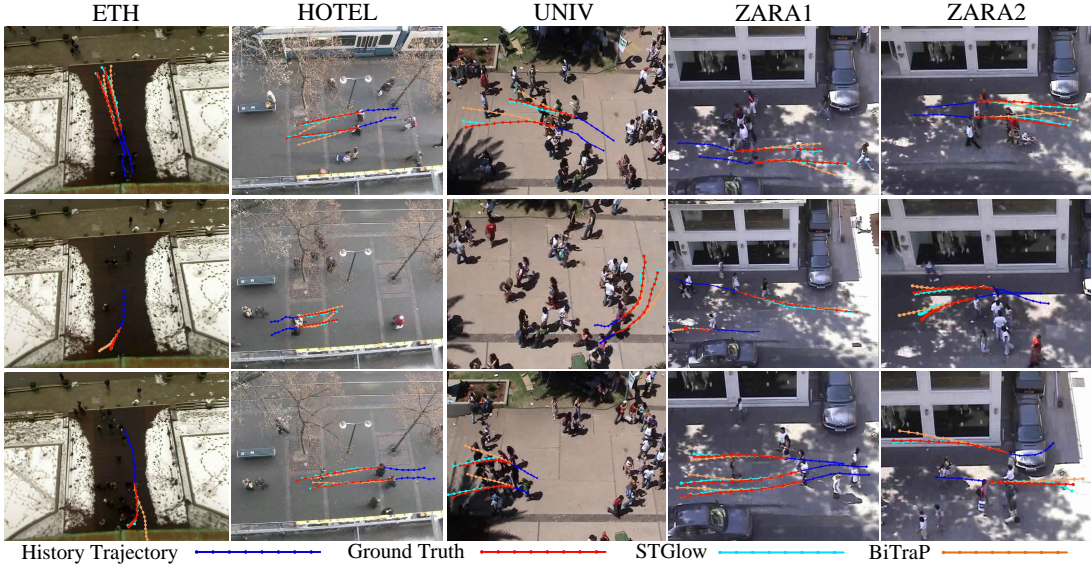
Fig. 7. Visualization of predicted trajectories on the ETH/UCY Dataset. Given the history trajectories (blue line), we illustrate the ground truth paths (red line) and predicted future trajectories by our STGlow (green line) and the previous state-of-the-art BiTrap (orange line) for five different scenes, and dots are the locations of pedestrians at different time steps. Best viewed in color and zoom-in for more clarity.
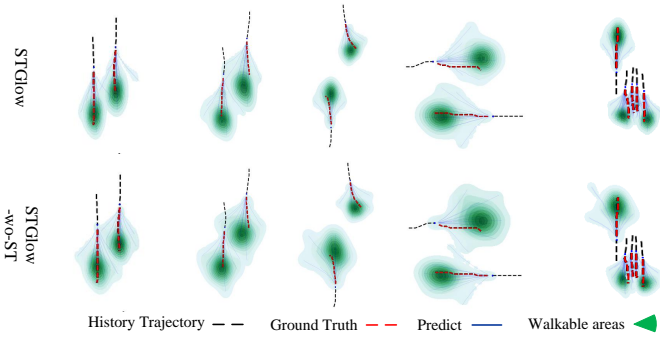

Fig. 8. Examples of diverse predictions of our method with and without dual graphormer (ST) on SDD. Black dash represents pedestrian historical trajectory, red dash denotes the ground truth trajectories, thin solid blue line represents the predicted diverse trajectories, and the green area represents the walkable area, which is visualized by the kernel density estimation map drawn from the predicted 20 trajectories. Better viewed in color.

TABLE VI
ABLATION EXPERIMENTS ON HYPERPARAMETERS OF THE LOSS
FUNCTION ON SDD DATASET.

α	λ_1	λ_2	λ_3	ADE/FDE
	0.0	0.0	1.0	7.25/11.39
	0.0	1.0	0.0	7.28/11.50
0.0	1.0	0.0	0.0	7.27/11.54
	0.25	0.25	0.5	7.23/11.35
	0.5	0.5	0.0	7.30/11.47
	0.0	0.0	1.0	7.24/11.30
	0.0	1.0	0.0	7.28/11.37
0.5	1.0	0.0	0.0	7.26/11.45
	0.25	0.25	0.5	7.23/11.28
	0.5	0.5	0.0	7.27/11.44
	0.0	0.0	1.0	7.22/11.25
	0.0	1.0	0.0	7.27/11.28
1.0	1.0	0.0	0.0	7.25/11.40
	0.25	0.25	0.5	**7.20/11.20**
	0.5	0.5	0.0	7.25/11.25

TABLE V
THE PERFORMANCE OF VARIANTS OF DUAL GRAPHORMER ON SDD
DATASET.

TG			Results	SG			Results
CE	PE	A_{tmp}	ADE/FDE	SE	HE	A_{spa}	ADE/FDE
✓	×	×	7.27/11.42	✓	×	×	7.25/11.35
×	✓	×	7.22/11.43	×	✓	×	7.25/11.45
×	×	✓	7.24/11.48	×	×	✓	7.24/11.36
×	×	×	7.30/11.46	×	×	×	7.29/11.47
✓	✓	×	7.24/11.38	✓	✓	×	7.23/11.42
✓	✓	✓	**7.20/11.20**	✓	✓	✓	**7.20/11.20**

spatial graphormer (SG), temporal graphormer (TG), pattern normalization (PN), and bidirectional decoders (BiD). In the variations of our approach, we choose GRU or Transformer to replace our TG to extract the representation lef motion behaviors and use a widely-used forward decoder to replace our BiD. The ablation results are summarized in Table IV. Obviously,

each component in our framework improves performance to some extent. Compared with the third and fifth rows, it can be seen that our TG can extract temporal dependencies better than GRU due to considering the temporal dependencies in both behavior-independent and behavior-dependent situations. Besides, considering the uniqueness of behavior pattern, our PN normalizes the behavior pattern of each pedestrian, bringing a huge performance gain compared to the last two rows. Note that our spatial and temporal graphormer is significantly better than the standard Transformer on modeling social interactions in the fourth row, which further demonstrates the effectiveness of our proposed dual graphormer. Other ablation results also demonstrated the importance of our proposed components.

2) Dual graphormer analysis: To investigate the impact of each component in the developed temporal graphormer (TG) and spatial graphormer (SG) on trajectory prediction, including the centrality encoding (CE), position embedding (PE), adjacency matrix in temporal (A_{tmp}), spatial embedding

Methods	BiTrap	Trajectorn++	Y-Net+TTST	AgentFormer	DMRGCN	Ours
Time (ms/step)	**0.014**	0.864	13.294	1.882	0.852	0.174

(SE), steering embedding (HE), and adjacency matrix in spatial (A_{spa}), we conducted corresponding ablation experiments which are summarized in Table V. As shown by the left and right groups in the table, in general, each component we designed in TG and SG help to better model social interactions in both temporal and spatial domains, resulting in improved performance. Note that when no components are adopted, our TG and SG degenerate into commonly used Transformers, where the time step nodes and pedestrians in the scene are treated as fully connected undirected graphs. Obviously, our SG and TG significantly outperform the widely used Transformer that does not consider temporal dependencies and mutual spatial interactions.

3) Hyperparameters analysis: To ensure the rationality of the hyperparameter settings in the loss function L_{traj}, we further performed ablation experiments for hyperparameters α, λ_1, λ_2 and λ_3. The experimental results are summarized in Table VI. Since the forward and backward trajectory predictions have the same importance, we generally keep the balance coefficients (i.e., λ_1 and λ_2) consistent for both in ablation experiments. Note that when $\lambda_1 = 1.0$, we adopt the forward predicted trajectories as the final trajectories. Similarly, when $\lambda_2 = 1.0$, the backward predicted trajectories served as the final trajectories. In addition, we adopt the forward predicted trajectories as the final trajectories when $\lambda_1 = \lambda_2 = 0.5, \lambda_3 = 0.0$ and the bidirectional predicted trajectories as the final trajectories when $\lambda_3 \neq 0.0$. As shown in Table VI, in general, each part of the loss function L_{traj} (*i.e.*, goal estimation, forward trajectory prediction, backward trajectory prediction, and bidirectional trajectory prediction) contributes to the proposed model. Furthermore, we observed that bidirectional trajectory prediction ($\lambda_3 = 1.0$) is slightly superior to forward trajectory prediction ($\lambda_1 = 1.0$) and backward trajectory prediction ($\lambda_2 = 1.0$). When supervising unidirectional and bidirectional trajectory prediction simultaneously, the model achieves the best performance. Based on the experimental results, we finally adopt: $\alpha = 1.0$, $\lambda_1 = 0.25$, $\lambda_2 = 0.25$ and $\lambda_3 = 0.5$ in our experiments.

G. Inference Time Analysis

To verify the efficiency of our proposed method, we conduct a comparison experiment on inference time with existing mainstream trajectory prediction frameworks. As demonstrated in Table VII, our method is inferior only to BiTrap in inference time but has significant performance improvements.

H. Discussion

In this subsection, we discuss some limitations of our approach. First, we experimentally find that our method fails to accurately predict the scenes when the motion behaviors of pedestrians change drastically in a short period of time.
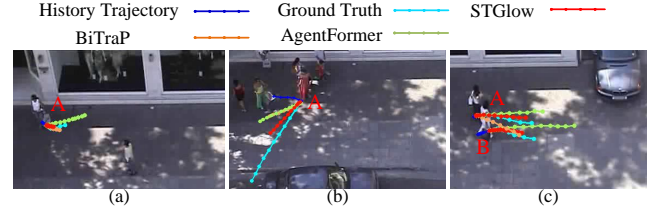

Fig. 9. Cases with poor performance on the ETH/UCY dataset. Better viewed in color.

未来研究方向

Fig. 9 provides several examples where pedestrians abruptly transition their motion behaviors. Specifically, pedestrian A in Fig. (a) was in a standing state for the first 6 time steps within the 8 observed time steps, and started to move slowly in the last two time steps. In Fig. (b), pedestrian A walked to the right at a constant speed for the first 5 time steps of the observation, but suddenly reduced her speed and changed her walking direction in the last three time steps. In Fig. (c), pedestrian B walked at a very slow speed for the first 7 time steps of the observation and then started walking at a normal speed in the last time step. Meanwhile, pedestrian A was stationary until the last time step of the observation, when it suddenly began walking. Though our approach can still cope with such scenarios somewhat better than existing methods such as BiTrap and AgentFormer, it still has a large error compared to the ground truth trajectory. How to address the scenes when the motion behaviors of pedestrians change drastically still needs more effort in future research.

Besides, modeling complex social interactions between pedestrians increases the time cost in the inference phase. When evaluated on ETH/UCY datasets, our method required 0.174 ms/step, BiTrap required 0.014 ms/step, Trajectron++ required 0.864 ms/step, and DMRGCN required 0.852 ms/step. Although our method is faster than Trajectron++ and DMRGCN, it is much slower than BiTrap. Fortunately, there has been a lot of recent work focusing on improving the efficiency of Transformers [72], [73]. We leave it as future work to build more efficient interaction modules.

V. CONCLUSION

In this paper, we have introduced a novel STGlow framework for trajectory prediction. Different from previous approaches, our method can more precisely model the underlying data distribution by optimizing the exact log-likelihood of observations. Besides, our method has clear physical meanings to simulate the evolution of human motion behaviors, where the forward process of the flow gradually decouples the complex motion behavior into a series of simple behaviors, while its reverse process represents the evolution of simple behaviors to the complex motion behavior. In addition, we have designed a novel dual graphormer to extract the global

social interaction of pedestrians in both temporal and spatial domains. Both quantitative and qualitative experimental results demonstrate the superiority of our approach under various situations.

REFERENCES

[1] 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 Conf. Comput. Vis. Pattern Recogn.*, 2019, pp. 5725–5734.

[2] P. T. Szemes, H. Hashimoto, and P. Korondi, "Pedestrian-behavior-based mobile agent control in intelligent space," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 6, pp. 2250–2257, 2005.

[3] B. Musleh, F. García, J. Otamendi, J. M. Armingol, and A. De la Escalera, "Identifying and tracking pedestrians based on sensor fusion and motion stability predictions," *Sensors*, vol. 10, no. 9, pp. 8028–8053, 2010.

[4] C. Huang, J. Wen, Y. Xu, Q. Jiang, J. Yang, Y. Wang, and D. Zhang, "Self-supervised attentive generative adversarial networks for video anomaly detection," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2022.

[5] V. Bastani, L. Marcenaro, and C. S. Regazzoni, "Online nonparametric bayesian activity mining and analysis from surveillance video," *IEEE Trans. Image Process.*, vol. 25, no. 5, pp. 2089–2102, 2016.

[6] L. Lin, Y. Lu, Y. Pan, and X. Chen, "Integrating graph partitioning and matching for trajectory analysis in video surveillance," *IEEE Trans Image Process.*, vol. 21, no. 12, pp. 4844–4857, 2012.

[7] F. Jiang, Y. Wu, and A. K. Katsaggelos, "A dynamic hierarchical clustering method for trajectory-based unusual video event detection," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 907–913, 2009.

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

[9] 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 Conf. Comput. Vis. Pattern Recogn.*, 2020, pp. 14 424–14 432.

[10] 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.

[11] R. Liang, Y. Li, X. Li, Y. Tang, J. Zhou, and W. Zou, "Temporal pyramid network for pedestrian trajectory prediction with multi-supervision," in *Proc. AAAI Conf. Art. Intel.*, vol. 35, 2021, pp. 2029–2037.

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

[13] Y. Li, R. Liang, W. Wei, W. Wang, J. Zhou, and X. Li, "Temporal pyramid network with spatial-temporal attention for pedestrian trajectory prediction," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1006–1019, 2022.

[14] T. Gu, G. Chen, J. Li, C. Lin, Y. Rao, J. Zhou, and J. Lu, "Stochastic trajectory prediction via motion indeterminacy diffusion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2022, pp. 17 113–17 122.

[15] H. Gao, Y. Qin, C. Hu, Y. Liu, and K. Li, "An interacting multiple model for trajectory prediction of intelligent vehicles in typical road traffic scenario," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–12, 2021.

[16] 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.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[18] 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.

[19] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 15 233–15 242.

[20] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, "Bitrap: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *IEEE Robot. Autom. Let.*, vol. 6, no. 2, pp. 1463–1470, 2021.

[21] G. Chen, J. Li, N. Zhou, L. Ren, and J. Lu, "Personalized trajectory prediction via distribution discrimination," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 15 580–15 589.

[22] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," *arXiv preprint arXiv:1410.8516*, 2014.

[23] 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 Conf. Comput. Vis. Pattern Recogn.*, 2021, pp. 8994–9003.

[24] B. Yang, G. Yan, P. Wang, C.-Y. Chan, X. Song, and Y. Chen, "A novel graph-based trajectory predictor with pseudo-oracle," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2021.

[25] M. Li, S. Chen, Y. Shen, G. Liu, I. W. Tsang, and Y. Zhang, "Online multi-agent forecasting with interpretable collaborative graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2022.

[26] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu, "Do transformers really perform badly for graph representation?" in *Proc. Adv. Neu. Inf. Process. Syst.*, vol. 34, 2021, pp. 28 877–28 888.

[27] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Phy. rev. E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[28] 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.

[29] M. K. C. Tay and C. Laugier, "Modelling smooth paths using gaussian processes," in *Proc. Field and Service Robotics*, 2008, pp. 381–390.

[30] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2009, pp. 935–942.

[31] 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 Int. Conf. Comput. Vis.*, 2009, pp. 261–268.

[32] 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 Recogn.*, 2016, pp. 961–971.

[33] H. Xue, D. Q. Huynh, and M. Reynolds, "Poppl: Pedestrian trajectory prediction by lstm with automatic route class clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 77–90, 2021.

[34] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2019, pp. 12 085–12 094.

[35] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2017, p. 4674–4683.

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

[37] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2980–2988.

[38] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. Int. conf. mach. learn.*, 2014, pp. 1764–1772.

[39] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2015, pp. 3156–3164.

[40] Y. Hu, S. Chen, Y. Zhang, and X. Gu, "Collaborative motion prediction via neural motion message passing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2020, pp. 6319–6328.

[41] Y. Xu, J. Yang, and S. Du, "Cf-lstm: cascaded feature-based long short-term networks for predicting pedestrian trajectory," in *Proc. AAAI Conf. Art. Intel.*, vol. 34, no. 07, 2020, pp. 12 541–12 548.

[42] B. Xu, X. Shu, and Y. Song, "X-invariant contrastive augmentation and representation learning for semi-supervised skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 31, no. 5, pp. 3852–3867, 2022.

[43] X. Shu, B. Xu, L. Zhang, and J. Tang, "Multi-granularity anchor-contrastive representation learning for semi-supervised skeleton-based action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7559–7576, 2022.

[44] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2018, pp. 974–983.

[45] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proc. AAAI Conf. Art. Intel.*, vol. 33, no. 01, 2019, pp. 7370–7377.

[46] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *Proc. IEEE int. Conf. Robot. and Auto.*, 2018, pp. 1–7.

[47] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, "STGAT: Modeling spatial-temporal interactions for human trajectory prediction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6272–6281.

[48] I. Bae and H.-G. Jeon, "Disentangled multi-relational graph convolutional network for pedestrian trajectory prediction," in *Proc. AAAI Conf. Art. Intel.*, vol. 35, 2021, pp. 911–919.

[49] T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2019, pp. 12 126–12 134.

[50] J. Li, H. Ma, and M. Tomizuka, "Conditional generative neural system for probabilistic trajectory prediction," in *Proc. IEEE Int. Conf. Intel. Robots and Sys.*, 2019.

[51] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[52] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Proc. Adv. Neu. Inf. Process. Syst.*, vol. 31, 2018, pp. 10 236–10 245.

[53] A. Lugmayr, M. Danelljan, L. V. Gool, and R. Timofte, "Srflow: Learning the super-resolution space with normalizing flow," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 715–732.

[54] M. Kumar, M. Babaeizadeh, D. Erhan, C. Finn, S. Levine, L. Dinh, and D. Kingma, "Videoflow: A flow-based generative model for video," *arXiv preprint arXiv:1903.01434*, 2019.

[55] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *IEEE Int. Conf. Acoust. Speech Signal Process.*, 2019, pp. 3617–3621.

[56] S. Kim, S.-g. Lee, J. Song, J. Kim, and S. Yoon, "Flowavenet: A generative flow for raw audio," *arXiv preprint arXiv:1811.02155*, 2018.

[57] 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.

[58] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[59] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[60] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z.-H. Jiang, F. E. Tay, J. Feng, and S. Yan, "Tokens-to-token vit: Training vision transformers from scratch on imagenet," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 558–567.

[61] J. You, Y. Li, J. Zhou, Z. Hua, W. Sun, and X. Li, "A transformer based approach for image manipulation chain detection," in *Proc. ACM Int. Conf. Multi.*, 2021, pp. 3510–3517.

[62] Y. Li, J. You, J. Zhou, W. Wang, X. Liao, and X. Li, "Image operation chain detection with machine translation framework," *IEEE Trans. Multimed.*, pp. 1–16, 2022.

[63] Q. Song, B. Sun, and S. Li, "Multimodal sparse transformer network for audio-visual speech recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–11, 2022.

[64] Y. Li, J. Zhou, J. Tian, X. Zheng, and Y. Y. Tang, "Weighted error entropy-based information theoretic learning for robust subspace representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 9, pp. 4228–4242, 2022.

[65] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[66] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[67] 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. Art. Intel.*, 2022.

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[69] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," *Computer graphics forum*, vol. 26, no. 3, pp. 655–664, 2007.

[70] 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.

[71] L. Shi, L. Wang, C. Long, S. Zhou, F. Zheng, N. Zheng, and G. Hua, "Social interpretable tree for pedestrian trajectory prediction," *arXiv preprint arXiv:2205.13296*, 2022.

[72] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *ACM Comput. Surv.*, vol. 55, no. 6, pp. 1–28, 2022.

[73] K. M. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Q. Davis, A. Mohiuddin, L. Kaiser, D. B. Belanger, L. J. Colwell, and A. Weller, "Rethinking attention with performers," in *Int. Conf. Learn. Represent.*, 2021.
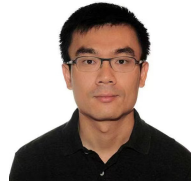
Rongqin Liang (Student Member, IEEE) received the B.Eng. degree in communication engineering from Wuyi University, Guangdong, China, in 2018 and M.S. degree in Information and Communication Engineering from Shenzhen University, Shenzhen, China, in 2021. He is currently a Ph.D. candidate at the College of Electronics and Information Engineering from Shenzhen University. His current research interests include trajectory prediction, anomaly detection, computer vision and deep learning.

Yuanman Li (Member, IEEE) received the B.Eng. degree in software engineering from Chongqing University, Chongqing, China, in 2012, and the Ph.D. degree in computer science from University of Macau, Macau, 2018. From 2018 to 2019, he was a Post-doctoral Fellow with the State Key Laboratory of Internet of Things for Smart City, University of Macau. He is currently an Assistant Professor with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China. His current research interests include multimedia security and forensics, data representation, computer vision and machine learning.

Jiantao Zhou (Senior Member, IEEE) received the B.Eng. degree from the Department of Electronic Engineering, Dalian University of Technology, in 2002, the M.Phil. degree from the Department of Radio Engineering, Southeast University, in 2005, and the Ph.D. degree from the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, in 2009. He held various research positions with University of Illinois at Urbana-Champaign, Hong Kong University of Science and Technology, and McMaster University. He is an Associate Professor with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, and also the Interim Head of the newly established Centre for Artificial Intelligence and Robotics. His research interests include multimedia security and forensics, multimedia signal processing, artificial intelligence and big data. He holds four granted U.S. patents and two granted Chinese patents. He has co-authored two papers that received the Best Paper Award at the IEEE Pacific-Rim Conference on Multimedia in 2007 and the Best Student Paper Award at the IEEE International Conference on Multimedia and Expo in 2016. He is serving as the Associate Editors of the IEEE TRANSACTIONS on IMAGE PROCESSING and the IEEE TRANSACTIONS on MULTIMEDIA.

Xia Li (Member, IEEE) received her B.S. and M.S. in electronic engineering and SIP (signal and information processing) from Xidian University in 1989 and 1992 respectively. She was later conferred a Ph.D. in Department of information engineering by the Chinese University of Hong Kong in 1997. Currently, she is a member of the Guangdong Key Laboratory of Intelligent Information Processing. Her research interests include intelligent computing and its applications, image processing and pattern recognition.