BCDiff: Bidirectional Consistent Diffusion for Instantaneous Trajectory Prediction

Rongqing Li
Beijing Institute of Technology
lirongqing99@gmail.com

Changsheng Li *
Beijing Institute of Technology
lcs@bit.edu.cn

Dongchun Ren
ALLRIDE.AI
Dongchun.ren@allride.ai

Guangyi Chen
CMU & MBZUAI
guangyichen1994@gmail.com

Ye Yuan
Beijing Institute of Technology
yuan-ye@bit.edu.cn

Guoren Wang
Beijing Institute of Technology
wanggrbit@126.com

Abstract

The objective of pedestrian trajectory prediction is to estimate the future paths of pedestrians by leveraging historical observations, which plays a vital role in ensuring the safety of self-driving vehicles and navigation robots. Previous works usually rely on a sufficient amount of observation time to accurately predict future trajectories. However, there are many real-world situations where the model lacks sufficient time to observe, such as when pedestrians abruptly emerge from blind spots, resulting in inaccurate predictions and even safety risks. Therefore, it is necessary to perform trajectory prediction based on instantaneous observations, which has rarely been studied before. In this paper, we propose a **Bi**-directional **C**onsistent **Diff**usion framework tailored for instantaneous trajectory prediction, named **BCDiff**. At its heart, we develop two coupled diffusion models by designing a mutual guidance mechanism which can bidirectionally and consistently generate unobserved historical trajectories and future trajectories step-by-step, to utilize the complementary information between them. Specifically, at each step, the predicted unobserved historical trajectories and limited observed trajectories guide one diffusion model to generate future trajectories, while the predicted future trajectories and observed trajectories guide the other diffusion model to predict unobserved historical trajectories. Given the presence of relatively high noise in the generated trajectories during the initial steps, we introduce a gating mechanism to learn the weights between the predicted trajectories and the limited observed trajectories for automatically balancing their contributions. By means of this iterative and mutually guided generation process, both the future and unobserved historical trajectories undergo continuous refinement, ultimately leading to accurate predictions. Essentially, BCDiff is an encoder-free framework that can be compatible with existing trajectory prediction models in principle. Experiments show that our proposed BCDiff significantly improves the accuracy of instantaneous trajectory prediction on the ETH/UCY and Stanford Drone datasets, compared to related approaches.

*Changsheng Li (lcs@bit.edu.cn) is the corresponding author

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

https://doi.org/10.52202/075280-0633

1 Introduction

Pedestrian trajectory prediction aims to predict future trajectories conditioned on their past movements, which is an important task for autonomous driving [25, 57] and navigation robot [4]. Previous pedestrian trajectory prediction approaches usually rely on long enough observation time (typically, 2 to 3 seconds) for a pedestrian to precisely predict the future trajectories [53, 48, 54, 13]. However, in many real-world situations, e.g., when pedestrians suddenly emerge from blind spots and are in close proximity to autonomous vehicles, traditional trajectory prediction methods do not have ample time to collect a sufficient number of locations. This leads to sub-optimal prediction performance and potentially unsafe behaviors in the decision-making of autonomous vehicles and robots. Therefore, it is quite necessary to forecast future trajectories based on limited or instantaneous observations.

The prediction of instantaneous trajectories for pedestrians is a highly challenging task due to the limited observation time. In some cases, as extreme as it can be, merely two frames of locations can be observed. In the face of such a challenging task, there have been only a few academic works to date. MOE [46] is the first to propose the problem of instantaneous trajectory prediction, and thus is the most relevant to ours. MOE incorporates scene context information into limited observations and introduces the masked trajectory complement and context restoration as self-supervised tasks to pre-train the model. However, since MOE only utilizes instantaneous temporal information acquired from limited trajectories, it might be hard to accurately predict the future trajectories of a pedestrian with complex behavior such as turning and yielding. DTO [34] focuses on lowering the influence of noise introduced by incorrect detection and tracking, and attempts to employ limited observed trajectory to alleviate this problem. It utilizes the knowledge distillation technique to distill knowledge from a teacher model trained with an ample amount of long observations, and transfer the knowledge to a student model receiving fewer observations as input. Although these approaches have shown some effectiveness in instantaneous trajectory prediction, the representation of a pedestrian is restricted to two frames of locations, which contains extremely limited temporal information.

In this paper, we propose BCDiff, a bidirectional consistent diffusion framework specifically designed for the instantaneous trajectory prediction task. As we know, the diffusion model is a generative model, which has been successfully applied to various generation tasks, including image synthesis [39, 33], image denoising [23], etc. Different from them, we leverage the diffusion model for instantaneous trajectory prediction. We devise two coupled diffusion models to bidirectionally generate previous unobserved trajectories and future trajectories from random noises, which can address the issue of temporal information scarcity in limited observations. The underlying intuition behind this idea is: Both previous unobserved historical trajectories and future trajectories contain information of the same pedestrian at different timesteps, and thereby they provide complementary information to each other. It will be beneficial for the prediction of future trajectories if we can design an elegant method to simultaneously generate previous unobserved historical trajectories and future trajectories by fully leveraging the complementary information between them.

To accomplish this, we devise a step-by-step mutual guidance mechanism in two coupled diffusion models to simultaneously generate previous unobserved historical trajectories and future trajectories. Specifically, at each step, the predicted unobserved historical trajectories, together with the limited observed trajectories serve as a guidance for one diffusion model to predict a denoising intensity, which is then used to generate future trajectories of the subsequent step. Likewise, the predicted future trajectories, together with the observed trajectories, guide the other diffusion model to predict unobserved historical trajectories of the next step. Meanwhile, considering there exists relatively high noise in the generated trajectories during the initial steps, we devise a gating mechanism to learn the weights between the predicted trajectories and the limited observed trajectories for automatically controlling the proportion of the guidance information from two kinds of trajectories during each generation step. Through this iterative and mutually guided generation process, the future and unobserved historical trajectories are continuously refined, ultimately leading to precise predictions. Notably, our proposed BCDiff is encoder-free and is compatible with existing trajectory prediction encoders in principle, allowing them to gracefully handle cases with instantaneous observations.

Our contributions can be summarized as follows: 1) We propose BCDiff, a diffusion model based framework tailored for instantaneous trajectory prediction. BCDiff can simultaneously generate both future and unobserved historical trajectories in a consistent manner, which can effectively leverage complementary information between them. 2) We devise a step-by-step mutual guidance mechanism to couple two diffusion models for trajectory generation, and present a gating strategy to adaptively

https://doi.org/10.52202/075280-0633

adjust the contributions of the guidance information between two kinds of trajectories. 3) Experiments demonstrate our proposed BCDiff significantly improves the accuracy of instantaneous prediction and outperforms the state-of-the-art methods on ETH/UCY and Stanford Drone datasets.

2 Related Works

2.1 Traditional Trajectory Prediction

Traditional trajectory prediction methods aim to predict future trajectories given sufficient observation time. To capture complex interactions between pedestrians, many methods have been proposed [1, 15, 40]. These models utilize a social mechanism to aggregate neighboring actors and broadcast information to each actor. In addition, graph neural networks [49, 11, 19, 22, 27, 28] and transformer architectures [52, 36, 35, 51] are introduced to encapsulate implicit interactions among pedestrians. To resolve the problem of high uncertainty in pedestrians, researchers propose stochastic generative models, such as GAN [15, 22, 40, 45, 55], VAE [25, 24, 31, 53], and Diffusion Models [14, 32], to better capture the variability in future trajectories. Various sampling strategies are designed to avoid purely random sampling [6, 3, 18, 5, 30]. However, the aforementioned methods can accurately predict trajectories only when a sufficient amount of long observation trajectories are available. The accuracy cannot be guaranteed given the limited observation time. In contrast to these methods, our goal is to address the trajectory prediction problem under instantaneous observation scenarios.

2.2 Instantaneous Trajectory Prediction

Instantaneous trajectory prediction aims to predict future trajectories given a limited number of observed trajectory points. In the most extreme scenarios, merely two frames of locations can be observed. This task poses significant challenges due to the exceedingly short observation period. MOE [46] and DTO [34] are two trajectory prediction methods based on limited observed trajectory points. MOE proposes a feature extractor to incorporate image semantic information and develops a self-supervised task to enhance the representational ability of instantaneous observations. Meanwhile, DTO investigates the influence of noise introduced by detection or tracking to trajectory prediction, and intends to use fewer trajectory points in a knowledge distillation framework to address the issue. Despite these advances, they fail to address the inherent lack of temporal information in instantaneous trajectory prediction. In our study, we attempt to predict and leverage previous unobserved trajectories to capture more temporal information, thereby improving the prediction accuracy of future trajectories.

2.3 Diffusion Models

The diffusion model is a class of stochastic generation models, which exhibits amazing performance in a diverse range of fields such as image synthesis [9, 39, 33], audio synthesis [7, 21] and text generation [2, 12, 8]. Among these works, a typical diffusion model is the Denoising diffusion probabilistic model (DDPM) [44, 16]. DDPM is inspired by the non-equilibrium thermodynamics, in which a forward Markov process perturbs real data into noise, and a reverse Markov process converts noise back to real data. DDPM has been widely used in various tasks, including image super-resolution [41, 17], 3D point cloud generation [29, 56] etc. For instance, CDM [17] cascades multiple DDPM models to generate images of increasing resolution. The work in [29] employs a heat bath mechanism on DDPM to facilitate the generation of 3d point clouds. Different from these works, we leverage DDPM to solve the problem of instantaneous trajectory prediction.

3 Methods

3.1 Problem Formulation

In this work, we aim to tackle the task of instantaneous trajectory prediction, where we assume only two frames are observed, i.e., the most extreme case. We denote $X_{obs} = \{x_1, x_2\}$ as the observation locations. The ground-truth future trajectory is symbolized as $X_{fut} = \{x_3, x_4, ..., x_{T_{fut}+2}\}$, where T_{fut} is the prediction length, and $x_i \in \mathbb{R}^2$ is the 2D coordinate of a trajectory location. Moreover, we characterize the previous unobserved trajectories as $X_{unobs} = \{x_{1-T_{unobs}}, ..., x_{-1}, x_0\}$, where T_{unobs} represents the length of the unobserved trajectories. Our objective is to develop a diffusion

https://doi.org/10.52202/075280-0633

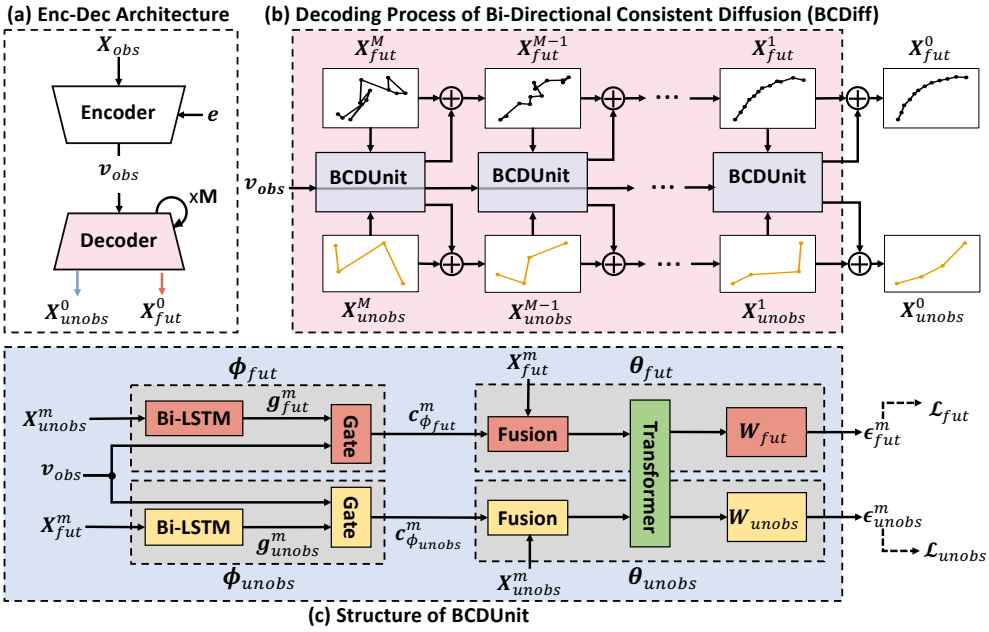

Figure 1: An illustration of our BCDiff framework. (a) The overall architecture comprises an encoder and a decoder. The encoder is utilized to generate features of trajectories by incorporating social and scene context. The decoder is depicted in Figure 1 (b). The decoder, i.e., our proposed BCDiff, generates previous unobserved historical trajectories and future trajectories step-by-step through two coupled diffusion models. (c) The BCDUnit describes the details of two diffusion models marked in red and yellow. We design a mutual guidance mechanism and predict denoise intensities that are used for generating unobserved historical trajectories and future trajectories at the next step.

models based method to generate both previous unobserved historical and future trajectories, so that more temporal information can be captured to better predict future trajectories, given only two observed frames. Since we utilize the diffusion model, we denote the maximum diffusion steps as M, and use X^m_{fut} and X^m_{unobs} to represent the future and unobserved trajectories following a diffusion of m steps or a denoising of $M-m$ steps, respectively. Note that $X_{unobs} = X^0_{unobs}$ and $X_{fut} = X^0_{fut}$.

3.2 Overall Architecture

The overall architecture consists of an encoder and a decoder, as shown in Figure 1 (a). The encoder encodes X_{obs} as v_{obs}, and captures social and scene context $\mathbf{e}$. The decoder is our proposed BCDiff framework, as shown in Figure 1 (b), which simultaneously generates previous observed historical trajectories and future trajectories step-by-step through the Bidirectional Consistent Denoising Unit (BCDUnit). The BCDUnit comprises two coupled diffusion models, As shown in Figure 1 (c), the diffusion model $\{\phi_{fut}, \theta_{fut}\}$ marked in red color, is responsible for generating future trajectories, while the diffusion model $\{\phi_{unobs}, \theta_{unobs}\}$, depicted in yellow, is used for generating unobserved historical trajectories. The two diffusion models are coupled through a mutual guidance mechanism. To be specific, as the m^{th} step, the network $\phi = \{\phi_{fut}, \phi_{unobs}\}$ leverages two bidirectional LSTMs to encode unobserved trajectories and future trajectories as mutual guidance g^m_{fut} and g^m_{unobs} used for generating each other in the $m-1^{th}$ step. Considering the guidance information contains relatively high noise during the initial steps, we introduce a gate mechanism to balance the contributions between the observed guidance v_{obs} and future guidance g^m_{fut}, as well as the observed guidance v_{obs} and unobserved guidance g^m_{unobs}, ultimately producing the appropriate guidance $\mathbf{c}_{\phi,fut}$ and $\mathbf{c}_{\phi,unobs}$. Then the network $\theta = \{\theta_{fut}, \theta_{unobs}\}$ fuses the guidance with both predicted unobserved historical and future trajectories at current steps to generate the denoise intensities, which is used to simultaneously generate X^{m-1}_{fut} and X^{m-1}_{unobs} of the next step. It is noteworthy that both previous unobserved historical trajectories and future trajectories contain the information of the same pedestrian at different timesteps, thus we adopt a parameter-shared transformer across θ_{fut} and θ_{unobs}.

https://doi.org/10.52202/075280-0633

3.3 Bidirectional Consistent Diffusion

In this section, we introduce our proposed BCDiff framework, which contains two coupled diffusion models to consistently generate trajectories in two directions, i.e., simultaneously predicting previous unobserved trajectories and future trajectories. In this paper, we utilize DDPM [16] as our basic diffusion model, because of its excellent performance in various tasks. For the generation of each direction, the diffusion model executes diffusion and conditional denoising processes. The diffusion process aims to intentionally add a series of noises to a ground-truth trajectory, while the conditional denoising process recovers the trajectory from noise inputs conditioned on the guidance.

Diffusion Process. The diffusion process is defined as a Markov chain, conditioned on the ground-truth trajectories X^0_{fut}, X^0_{unobs}. To write conveniently, we omit the subscripts, allowing X^0 to represent unobserved historical trajectories or future trajectories. The diffusion process generates the sequence $\{X^i\}_{i=1}^M$ by accumulating noise M times, i.e.,

$$q(X^{1:M}|X^0) = \prod_{m=1}^{M} q(X^m|X^{m-1}), \quad q(X^m|X^{m-1}) = \mathcal{N}(X^m; \sqrt{\alpha^m}X^{m-1}, (1-\alpha^m)\mathbf{I}), \quad (1)$$

where $\mathcal{N}$ denotes the Gaussian distribution, and α^m represents the noise intensity from X^m to X^{m-1}. Typically, α^m is equal to $1-\beta^m$, where β^m is a pre-defined value belonging to the interval [0,1]. Due to the additivity of Gaussian distributions, we are able to directly obtain X^m from X^0:

$$q(X^m|X^0) = \mathcal{N}(X^m; \sqrt{\overline{\alpha}^m}X^0, (1-\overline{\alpha}^m)\mathbf{I}), \quad \overline{\alpha}^m = \prod_{i=1}^{m} \alpha^i, \quad (2)$$

where $\overline{\alpha}^m = \prod_{i=1}^{m} \alpha^i$. Note that as M becomes sufficiently large, $\overline{\alpha}$ approaches to zero, making $q(X^M|X^0)$ converge to the standard Gaussian distribution. By employing the above process, the ground-truth trajectory is transformed into the Gaussian noise $X^M \sim \mathcal{N}(0, I)$.

Conditional Denoising Process. In this process, we aim to generate X^0 from the Gaussian noise $X^M \sim \mathcal{N}(0, I)$. We can reverse the aforementioned diffusion process, to gradually denoise from the Gaussian noise X^M and reconstruct X^0. Based on the proof in [10]: If $q(X^m|X^{m-1})$ follows a Gaussian distribution and β^m is sufficiently small, $q(X^{m-1}|X^m)$ also satisfies a Gaussian distribution. Therefore, we formulate $q(X^{m-1}|X^m)$ as a Gaussian Markov process. However, it is intractable to directly obtain $q(X^{m-1}|X^m)$. Consequently, we employ a denoising neural network to estimate its mean and variance:

$$p_\Theta(X^{0:M}|\mathbf{c}^m_\phi) = p(X^M|\mathbf{c}^m_\phi) \prod_{m=1}^{M} p_\Theta(X^{m-1}|X^m, \mathbf{c}^m_\phi), \quad (3)$$

$$p_\Theta(X^{m-1}|X^m, \mathbf{c}^m_\phi) = \mathcal{N}(X^{m-1}|\mathbf{c}^m_\phi; \mu_\Theta(X^m, m, \mathbf{c}^m_\phi), \mathbf{\Sigma}_\Theta(X^m, m, \mathbf{c}^m_\phi)), \quad (4)$$

where $\Theta = \{\phi, \theta\}$ is the parameter of the neural network. μ_Θ and Σ_Θ are the predicted mean and variance by Θ. $\mathbf{c}^m_\phi$ is produced by network ϕ, serving as conditions to guide the denoising (we will introduce it in detail later). Our ultimate goal in the denoising process is to ensure that the denoising step becomes the inverse process of the diffusion step, thus enabling $p_\Theta(X^{m-1}|X^m, \mathbf{c}^m_\phi)$ and $q(X^{m-1}|X^m)$ to have the same distribution. For more details, please refer to Section 3.4.

Bidirectional Consistent Denoising Unit. To utilize more temporal complementary information in trajectories, we propose the BCDUnit to couple two diffusion models. One is called the backward model, used to generate previous unobserved historical trajectories. The other, named as the forward model, is employed to produce future trajectories. We design a mutual guidance mechanism in the two diffusion models: At each step, the predicted unobserved historical trajectories and observed trajectories are jointly utilized to guide the forward model to generate future trajectories of the next step. Likewise, the predicted future trajectories together with observed trajectories are responsible for guiding the backward model to generate the unobserved historical trajectories. In this way, the BCDUnit continuously refines the future and unobserved historical trajectories, ultimately leading to accurate predictions.

We denote the network in BCDUnit as $\Theta = \{\{\phi_{fut}, \theta_{fut}\}, \{\phi_{unobs}, \theta_{unobs}\}\}$. As illustrated in Figure 1 (c), the forward model $\{\phi_{fut}, \theta_{fut}\}$, marked as red, is used to generate future trajectories, while the backward model $\{\phi_{unobs}, \theta_{unobs}\}$, depicted in yellow, is responsible for generating

https://doi.org/10.52202/075280-0633

unobserved historical trajectories. Here, the guidance information can be obtained by the network $\phi = \{\phi_{fut}, \phi_{unobs}\}$. At the m^{th} step, ϕ first generates future guidance g^m_{fut} and unobserved guidance g^m_{unobs} by sending the unobserved historical trajectories X^m_{unobs} and future trajectories X^m_{fut} to Bidirectional Long-Short Term Memory (Bi-LSTM), respectively:

$$g^m_{fut} = \textbf{Bi-LSTM}(X^m_{unobs}),\ g^m_{unobs} = \textbf{Bi-LSTM}(X^m_{fut}). \tag{5}$$

We further incorporate observed trajectory guidance v_{obs} into future and unobserved guidance to obtain more informative guidance $\mathbf{c}^m_{\phi,fut}$ and $\mathbf{c}^m_{\phi,unobs}$, $\mathbf{c}^m_{\phi,fut} = [g^m_{fut}, v_{obs}, m],\ \mathbf{c}^m_{\phi,unobs} = [g^m_{unobs}, v_{obs}, m]$, where $[\cdot,\cdot]$ represents the concatenation operation. However, considering that it contains relatively high noise in the initial steps due to the inherent property of the diffusion model, it is not appropriate to directly concatenate them in the beginning. To this end, we adopt a gating mechanism to automatically learn the weights for balancing the contributions between two kinds of guidance information. We first calculate the weights for the future guidance g^m_{fut} and observed trajectory guidance v_{obs} to produce appropriate guidance. Formally,

$$\gamma^m_{fut} = \mathbf{Gate}_{fut}([g^m_{fut}, v_{obs}, m]), \tag{6}$$

where γ^m_{fut} is the learnt weight. $\mathbf{Gate}$ is a two layers MLP with the Sigmoid activation in this paper.

The guidance $\mathbf{c}^m_{\phi,fut}$ can be then obtained by:

$$\mathbf{c}^m_{\phi,fut} = \gamma^m_{fut} g^m_{fut} + (1 - \gamma^m_{fut}) v_{obs}. \tag{7}$$

Similarly, $\mathbf{c}^m_{\phi,unobs}$ can be obtained in the same way. After obtaining $\mathbf{c}^m_{\phi,fut}$ and $\mathbf{c}^m_{\phi,unobs}$, the network $\theta = \{\theta_{fut}, \theta_{unobs}\}$ utilizes them to perform mutual guidance. They fuse the guidance with the predicted trajectories at the m^{th} step. Then, a transformer is leveraged to capture temporal dependencies in the fused features. Finally, the outputs of the transformer are passed through two MLPs, i.e., W_{fut} and W_{unobs}, to obtain denoising intensities ϵ^m_{fut} and ϵ^m_{unobs}, respectively. These denoising intensities are employed to generate trajectories of the next steps, *i.e.*, X^{m-1}_{fut} and X^{m-1}_{unobs}.

In this way, we can bidirectionally and consistently generate unobserved historical trajectories and future trajectories step-by-step, effectively utilizing the complementary information between them.

3.4 The Objective Function

We define the objective as the negative log-likelihood of the model p_Θ under X^0_{fut} and X^0_{unobs} as

$$\mathcal{L} \ = \ \mathbb{E}[-log p_\Theta(X^0)]. \tag{8}$$

Here, we also omit the subscript for convenient writing. By minimizing the objective $\mathcal{L}$, the original trajectories X^0_{fut}, and X^0_{unobs} can be recovered through the denoising process. However, it is difficult to directly compute $\mathcal{L}$. Therefore, we employ the variational methods to derive the Variational Lower Bound (VLB) [20] of the expectation, denoted as:

$$\begin{aligned}\mathcal{L} \ \leq \ -\mathcal{L}_{VLB} = \mathbb{E}_q\left[log\frac{q(X^{1:M}|X^0)}{p_\Theta(X^{0:M})}\right] = \mathbb{E}_q[KL(q(X^M|X^0)||p_\Theta(X^M|\mathbf{c}^M_\phi) \\ - log p_\Theta(X^0|X^1, \mathbf{c}^1_\phi) + \sum_{m=2}^{M} KL(q(X^{m-1}|X^m, X^0)||p_\Theta(X^{m-1}|X^m, \mathbf{c}^m_\phi))].\end{aligned}$$

The first term of $\mathcal{L}_{VLB}$ approximates to 0, as both $q(X^M|X^0)$ and $p_\Theta(X^M|\mathbf{c}^M_\phi)$ are approximate to $\mathcal{N}(0, I)$. The second term can be formulated as a special case of the third term when $m = 1$. The third term computes the KL divergence between the estimated distribution $p_\Theta(X^{m-1}|X^m, \mathbf{c}^m_\phi)$ and the true posterior distribution $q(X^{m-1}|X^m, X^0)$, aiming to lower the error between the estimated distribution and the ground-truth posterior distribution. To determine the $q(X^{m-1}|X^m, X^0)$, we apply the Bayes formula as follows:

$$q(X^{m-1}|X^m, X^0) = q(X^m|X^{m-1}, X^0)\frac{q(X^{m-1}|X^0)}{q(X^m|X^0)} = q(X^m|X^{m-1})\frac{q(X^{m-1}|X^0)}{q(X^m|X^0)}. \tag{9}$$

https://doi.org/10.52202/075280-0633

By applying the Bayes formula, we observe each term can be calculated with Equation 2. We then substitute the results of Equation 2 to Equation 9 to obtain the mean and variance of the posterior distribution $q(X^{m-1}|X^m, X^0)$ as:

$$\widetilde{\sigma}^m = \frac{1-\overline{\alpha}^{m-1}}{1-\overline{\alpha}^m} \cdot \beta^m, \quad \widetilde{\mu}^m(X^m, X^0) = \frac{\sqrt{\alpha^m}(1-\overline{\alpha}^{m-1})}{1-\overline{\alpha^m}} X^m + \frac{\sqrt{\overline{\alpha}^{m-1}}\beta^m}{1-\overline{\alpha}^m} X^0. \tag{10}$$

Note that $\widetilde{\sigma}^m$ is a constant value related to β^m, thus the KL in the third term can be further derived as:

$$KL(q(X^{m-1}|X^m, X^0)||p_\Theta(X^{m-1}|X^m, \mathbf{c}_\phi^m)) \propto ||\widetilde{\mu}^m(X^m, X^0) - \mu_\Theta(X^m, m, \mathbf{c}_\phi^m)||_2. \tag{11}$$

By reparameterizing μ_Θ and substituting it into Equation 11, we can further simplify the expression and finally obtain the diffusion loss.

$$\begin{aligned} \mathcal{L}_{unobs} &= \mathbb{E}_{X^0_{unobs},\epsilon,m}||\epsilon - \epsilon_\Theta(X^m_{unobs}, m, \mathbf{c}^m_{\phi,unobs})||_2 \\ \mathcal{L}_{fut} &= \mathbb{E}_{X^0_{fut},\epsilon,m}||\epsilon - \epsilon_\Theta(X^m_{fut}, m, \mathbf{c}^m_{\phi,fut})||_2 \\ \mathcal{L}_d &= \mathcal{L}_{fut} + \mathcal{L}_{unobs} \end{aligned} \tag{12}$$

where $\epsilon \sim \mathcal{N}(0, I)$, X^m_{fut} and X^m_{unobs} are calculated by Equation 2, and ϵ_Θ represents the aforementioned BCDUnit.

3.5 Mutual Guidance based Optimizing and Inference

When optimizing the model, we take the following four steps: Firstly, we sample future ground-truth trajectories X^0_{fut}, instantaneous observations X_{obs}, and unobserved historical ground-truth trajectories X^0_{unobs} from the training dataset. We also sample a timestep $m \sim uniform(1, M)$ and a Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$. Secondly, we use the encoder to encode X_{obs} as the v_{obs}, and employ the Equation 2 to obtain the X^m_{fut} and X^m_{unobs}. Thirdly, we utilize the Equation 5,6, and 7 to obtain mutual guidance $\mathbf{c}^m_{\phi,unobs}$ and $\mathbf{c}^m_{\phi,fut}$. Finally, we calculate the loss defined in Equation 12 and take the gradient descent to optimize the model until it converges.

For inference, we first sample instantaneous observations X_{obs} from the testing data and two Gaussian noises $\hat{X}^M_{fut}$ and $\hat{X}^M_{unobs}$ from $\mathcal{N}(0, I)$. Following the third step in the optimizing stage, we obtain mutual guidance $\mathbf{c}^M_{\phi,fut}$ and $\mathbf{c}^M_{\phi,unobs}$. Finally, we execute the following two updates from $m = M$ to 1 so that the predicted trajectories can be continuously refined until $\hat{X}^0_{fut}$ and $\hat{X}^0_{unobs}$ are generated,

$$\hat{X}^{m-1}_{fut} := \frac{1}{\sqrt{\alpha^m}}(\hat{X}^m_{fut} - \frac{\beta^m}{\sqrt{1-\overline{\alpha}^m}}\epsilon_\Theta(\hat{X}^m_{fut}, m, \mathbf{c}^m_{\phi,fut})) + \widetilde{\sigma}^m\epsilon, \tag{13}$$

$$\hat{X}^{m-1}_{unobs} := \frac{1}{\sqrt{\alpha^m}}(\hat{X}^m_{unobs} - \frac{\beta^m}{\sqrt{1-\overline{\alpha}^m}}\epsilon_\Theta(\hat{X}^m_{unobs}, m, \mathbf{c}^m_{\phi,unobs})) + \widetilde{\sigma}^m\epsilon, \tag{14}$$

where ϵ is a random noise sampled from the standard Gaussian distribution. The details of training and inference procedures are provided in Appendix.

4 Experiments

4.1 Experiment Settings

Dataset. We verify the effectiveness of our proposed method on the widely used ETH/UCY [37, 26] and Stanford Drone [38] Dataset (SDD). ETH/UCY is a dataset group. It consists of 5 different scenes, among which 2 scenes (ETH, HOTEL) are from the ETH dataset, and the other three scenes (UNIV, ZARA1, and ZARA2) come from the UCY dataset. The whole dataset includes more than 1500 pedestrians. We follow the widely used leave-one-scene-out protocol, i.e., the models are trained on 4 scenes and tested on the remaining one [15, 42]. SDD is a large scale dataset consisting of 20 scenes. It contains various agents such as pedestrians, bicycles, and vehicles.

Evaluation Metrics. Following previous works [43, 19, 14, 53, 48], we employ the Average Displacement Error (ADE) and Final Displacement Error (FDE) as metrics to evaluate the performance of future trajectory predictions. In the instantaneous trajectory prediction setting, the observations are

https://doi.org/10.52202/075280-0633

Table 1: Comparisons of different methods on the ETH/UCY dataset. The metrics are presented as ADE/FDE (m).

Model	Methods	Dataset					
		ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Trajectron++	Instantaneous	0.76/1.43	0.30/0.56	0.36/0.74	0.22/0.42	0.18/0.34	0.36/0.70
	MOE [46]	0.64/1.12	0.20/0.33	0.33/0.62	0.22/0.42	**0.17/0.32**	0.31/0.56
	MOE w/o Image	0.68/1.22	0.25/0.49	0.35/0.68	0.22/0.42	0.18/0.33	0.34/0.63
	DTO [34]	0.70/1.23	0.22/0.45	0.32/0.62	0.22/0.42	**0.17**/0.33	0.33/0.61
	BCDiff	**0.61/1.09**	**0.16/0.28**	**0.28/0.53**	**0.22/0.41**	0.18/0.33	**0.29/0.53**
PCCSNet	Instantaneous	0.34/0.65	0.14/0.25	0.31/0.63	0.23/0.46	0.16/0.37	0.24/0.47
	MOE [46]	0.31/0.57	**0.13**/0.21	**0.25**/0.53	0.20/0.41	**0.14/0.31**	0.20/0.41
	MOE w/o Image	0.32/0.61	0.14/0.24	0.28/0.57	0.21/0.45	0.15/0.34	0.22/0.44
	DTO [34]	0.33/0.64	0.14/0.24	0.31/0.62	0.22/0.46	0.15/0.35	0.23/0.46
	BCDiff	**0.30/0.56**	**0.13/0.20**	**0.25/0.52**	**0.18/0.37**	**0.14/0.31**	**0.19/0.39**
SGCN	Instantaneous	0.88/1.66	0.55/1.16	0.38/0.71	0.30/0.54	0.25/0.46	0.47/0.91
	MOE [46]	0.74/1.41	0.45/0.85	0.38/0.71	**0.29**/0.54	0.25/0.45	0.42/0.79
	MOE w/o Image	0.79/1.52	0.50/1.05	0.38/0.71	0.30/0.54	0.25/0.46	0.44/0.85
	DTO [34]	0.80/1.56	0.49/1.02	0.38/0.71	0.30/0.54	0.25/0.46	0.44/0.86
	BCDiff	**0.66/1.18**	**0.34/0.62**	**0.38/0.70**	0.30/**0.54**	**0.25/0.44**	**0.39/0.72**
SocialVAE	Instantaneous	0.64/1.10	0.21/0.34	0.27/0.51	0.22/0.39	0.18/0.34	0.30/0.54
	MOE [46]	0.57/1.01	0.17/0.29	0.26/0.44	0.22/**0.36**	0.17/0.32	0.28/0.48
	MOE w/o Image	0.59/1.06	0.20/0.33	0.27/0.49	0.22/0.38	0.18/0.33	0.29/0.52
	DTO [34]	0.61/1.06	0.18/0.31	0.25/0.43	0.22/0.38	0.17/0.33	0.29/0.50
	BCDiff	**0.53/0.91**	**0.17/0.27**	**0.24/0.40**	**0.21**/0.37	**0.16/0.26**	**0.26/0.44**

reduced to 2 frames, and the length of future predictions is 12 frames. Following previous works [46, 15, 31], we sample 20 future predicted trajectories, and report the final error by the minimum error over all predicted trajectories.

Backbone and Baselines. To demonstrate the compatible ability of our BCDiff, we apply it to three popular trajectory prediction models, Trajectron++ [42], PCCSNet [47], SGCN [43], and SocialVAE [50] by replacing their decoders with our BCDiff. Moreover, we compare BCDiff with the following baselines: **Instantaneous** means directly predicting the trajectories of the next 12 frames conditions on 2 frames of observations using the above three backbones. Additionally, we take **MOE** [46] and **DTO** [34] as two baselines for instantaneous trajectory prediction. Since the original MOE employs image semantic information, we also implement MOE without using image semantic information, denoted as **MOE w/o Image**, in order to fairly compare with other methods.

4.2 Experiment Results and Analysis

Performance on Instantaneous Trajectory Prediction. The overall performance are listed in Table 1 and 2. The results by applying our BCDiff to three different backbones, consistently outperform the baseline methods on all the datasets. This demonstrates the effectiveness of our proposed method for instantaneous trajectory prediction. Meanwhile, it also illustrates our method can be well compatible with different trajectory prediction models. Note that the performance of MOE declines when image information is not included, which shows that MOE heavily depends on image semantic information. In contrast, our proposed method does not rely on any additional image information.

Ablation Study. We perform the ablation studies, as listed in Table 3. We first utilize two separate diffusion models to predict previous unobserved trajectories and future trajectories, respectively, denoting it as BCDiff-w/o.Guidance. Then, we only utilize unobserved and observed trajectories to guide the generation of future trajectories, denoted as BCDiff-Uni.Guidance. BCDiff-Uni.Guidance is better than BCDiff-w/o.Guidance, meaning that the guidance from previous unobserved historical trajectories is helpful for improving the prediction performance of future trajectories. Moreover, by

https://doi.org/10.52202/075280-0633

Table 2: Comparisons of different methods on the Stanford Drone dataset. The metrics are presented as ADE/FDE (m).

Methods	Trajectron++	PCCSNet	SGCN	SocialVAE
Instantaneous	13.07/22.88	9.19/17.71	15.40/25.69	9.56/16.10
MOE[46]	11.71/19.54	8.40/16.08	14.45/24.88	9.12/14.98
MOE w/o Image	12.41/21.46	8.87/16.80	15.02/25.13	9.39/15.87
DTO[34]	12.32/20.79	8.93/16.92	14.99/25.07	9.28/15.58
BCDiff	**11.56/19.32**	**8.32/15.87**	**13.67/23.92**	**9.05/14.86**

Table 3: Ablation Studies using Trajectron++ on the ETH/UCY and Stanford Drone datasets.

Method	ETH/UCY	SDD
BCDiff-w/o.Guidance	0.36/0.70	13.07/22.88
BCDiff-Uni.Guidance	0.33/0.60	12.57/21.26
BCDiff-Bi.Guidance	0.32/0.57	11.96/19.88
BCDiff-Bi.Guidance & Gate	0.29/0.53	11.56/19.32

incorporating bidirectional guidance, called BCDiff-Bi.Guidance, we observe the performance is further improved. This illustrates the guidance from future trajectories is beneficial for predicting previous unobserved trajectories, thereby improving the final prediction performance. Finally, we integrate the gating mechanism into our model, named as BCDiff-Bi.Guidance & Gate. It achieves the best performance, showing the gating mechanism is effective for trajectory prediction.

Analysis on Length of Predicted Unobserved Points. We investigate the impact of the number of predicted unobserved points on trajectory prediction. We use the Trajectron++ encoder [42], as listed in Table 4. As the number of predicted points increases, the accuracy of future predictions gradually improves, which can be attributed to the additional unobserved points providing more useful information. When $T = 5$, the prediction performance starts to decline. This is because it becomes unprecise when predicting previous unobserved historical points with a larger length, thus introducing noise into our method.

Table 4: Analysis of different unobserved point lengths T_{unobs}

Dataset	Direction	$T_{unobs} = 1$	$T_{unobs} = 2$	$T_{unobs} = 3$	$T_{unobs} = 4$	$T_{unobs} = 5$	$T_{unobs} = 6$
ETH/UCY	Future	0.31/0.55	0.30/0.54	0.30/0.53	**0.29/0.53**	0.32/0.57	0.34/0.60
	Unobserved	0.006/0.006	0.034/0.033	0.055/0.063	0.086/0.089	0.108/0.133	0.149/0.212
SDD	Future	11.86/19.78	11.81/19.71	11.68/19.50	**11.56/19.32**	11.92/20.01	12.33/20.98
	Unobserved	0.311/0.312	1.564/1.492	2.626/2.678	3.552/3.884	4.828/5.423	6.177/7.894

Qualitative Analysis. We visualize the predicted trajectories in four different scenarios: Walking side-by-side, walking along, turning and yielding, as shown in Figure 2. BCDiff can accurately predict future trajectories, compared to MOE and DTO in all four scenarios. This is because BCDiff utilizes temporal information of unobserved trajectories to aid the prediction of future trajectories. Note that when pedestrians are walking side-by-side, as depicted in the first column of Figure 2, BCDiff utilizes the predicted unobserved historical trajectories to capture the walking patterns between two pedestrians, thereby precisely forecasting future trajectories. However, DTO and MOE predict deviated trajectories, due to only using two observed frames. Moreover, we visualize the gating weights γ^m_{fut} and γ^m_{unobs} in the denoising process from $m = 40$ to $m = 1$. As shown in Figure 3, the weights are gradually increased. It indicates our gating mechanism can adaptively learn weights, assigning lower weights to the predicted guidance in the initial steps.

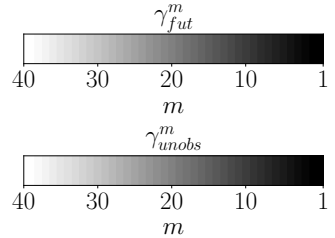

Figure 3: Visualization of weights γ^m_{fut} and γ^m_{unobs}. Darker colors represent larger values.

https://doi.org/10.52202/075280-0633

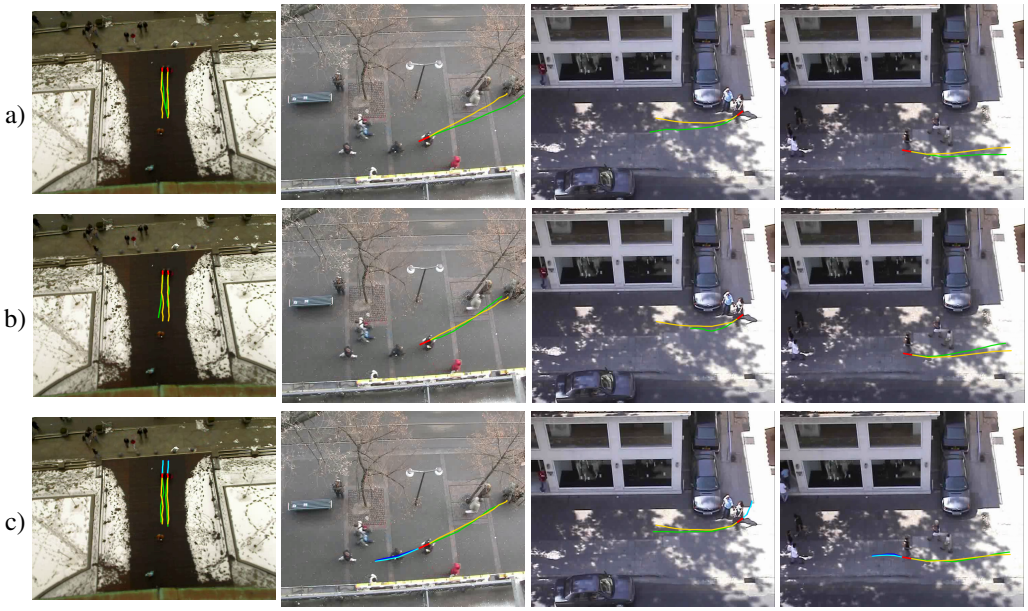

Figure 2: Visualization of predicted trajectories on the ETH/UCY Dataset. Given the instantaneous observed trajectories (red), we predict the future trajectories (green) by (a) MOE, (b) DTO and (c) Our BCDiff. The ground-truth future trajectories are shown in orange color. In addition, we also draw the predicted unobserved trajectories (blue) by our BCDiff and ground-truth (cyan). Our predicted future trajectories are closer to the ground-truth, compared to other methods.

5 Conclusion

We proposed a diffusion model based framework for the task of instantaneous trajectory prediction. The proposed framework simultaneously generated both unobserved historical and future trajectories by designing a mutual guidance mechanism to couple two diffusion models, such that more temporal information can be leveraged for future trajectory prediction. We introduced a gating strategy, automatically balancing the contributions of different guidance information. Experiments demonstrated our proposed framework achieved superior performance to the state-of-the-art methods.

Acknowledgments and Disclosure of Funding

This work was supported by the NSFC under Grants 62122013, U2001211. This work was also supported by the Innovative Development Joint Fund Key Projects of Shandong NSF under Grants ZR2022LZH007.

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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

[3] Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Non-probability sampling network for stochastic human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6477–6487, 2022.

[4] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *2019 international conference on robotics and automation (ICRA)*, pages 6015–6022. IEEE, 2019.

[5] Guangyi Chen, Zhenhao Chen, Shunxing Fan, and Kun Zhang. Unsupervised sampling promoting for stochastic human trajectory prediction. *arXiv preprint arXiv:2304.04298*, 2023.

[6] Guangyi Chen, Junlong Li, Nuoxing Zhou, Liangliang Ren, and Jiwen Lu. Personalized trajectory prediction via distribution discrimination. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15580–15589, 2021.

[7] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*, 2020.

[8] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022.

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[10] William Feller. On the theory of stochastic processes, with particular reference to applications. In *Selected Papers I*, pages 769–798. Springer, 2015.

[11] Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11525–11533, 2020.

[12] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*, 2022.

[13] Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15303–15312, 2021.

[14] Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic trajectory prediction via motion indeterminacy diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17113–17122, 2022.

[15] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2255–2264, 2018.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[17] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022.

[18] Xin Huang, Stephen G McGill, Jonathan A DeCastro, Luke Fletcher, John J Leonard, Brian C Williams, and Guy Rosman. Diversitygan: Diversity-aware vehicle motion prediction via latent semantic sampling. *IEEE Robotics and Automation Letters*, 5(4):5089–5096, 2020.

[19] Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6272–6281, 2019.

[20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[21] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020.

[22] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *Advances in Neural Information Processing Systems*, 32, 2019.

https://doi.org/10.52202/075280-0633

[23] Vladimir Kulikov, Shahar Yadin, Matan Kleiner, and Tomer Michaeli. Sinddm: A single image denoising diffusion model. *arXiv preprint arXiv:2211.16582*, 2022.

[24] Mihee Lee, Samuel S Sohn, Seonghyeon Moon, Sejong Yoon, Mubbasir Kapadia, and Vladimir Pavlovic. Muse-vae: multi-scale vae for environment-aware long term trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2221–2230, 2022.

[25] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 336–345, 2017.

[26] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer graphics forum*, volume 26, pages 655–664. Wiley Online Library, 2007.

[27] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. *Advances in neural information processing systems*, 33:19783–19794, 2020.

[28] Xin Li, Xiaowen Ying, and Mooi Choo Chuah. Grip++: Enhanced graph-based interaction-aware trajectory prediction for autonomous driving. *arXiv preprint arXiv:1907.07792*, 2019.

[29] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021.

[30] Yecheng Jason Ma, Jeevana Priya Inala, Dinesh Jayaraman, and Osbert Bastani. Likelihood-based diverse sampling for trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13279–13288, 2021.

[31] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 759–776. Springer, 2020.

[32] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. *arXiv preprint arXiv:2303.10895*, 2023.

[33] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[34] Alessio Monti, Angelo Porrello, Simone Calderara, Pasquale Coscia, Lamberto Ballan, and Rita Cucchiara. How many observations are enough? knowledge distillation for trajectory forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6553–6562, 2022.

[35] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. *arXiv preprint arXiv:2207.05844*, 2022.

[36] Jiquan Ngiam, Vijay Vasudevan, Benjamin Caine, Zhengdong Zhang, Hao-Tien Lewis Chiang, Jeffrey Ling, Rebecca Roelofs, Alex Bewley, Chenxi Liu, Ashish Venugopal, et al. Scene transformer: A unified architecture for predicting future trajectories of multiple agents. In *International Conference on Learning Representations*, 2022.

[37] Stefano Pellegrini, Andreas Ess, and Luc Van Gool. Improving data association by joint modeling of pedestrian trajectories and groupings. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I 11*, pages 452–465. Springer, 2010.

[38] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pages 549–565. Springer, 2016.

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[40] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1349–1358, 2019.

[41] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[42] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 683–700. Springer, 2020.

[43] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8994–9003, 2021.

[44] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[45] Hao Sun, Zhiqun Zhao, and Zhihai He. Reciprocal learning networks for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7416–7425, 2020.

[46] Jianhua Sun, Yuxuan Li, Liang Chai, Hao-Shu Fang, Yong-Lu Li, and Cewu Lu. Human trajectory prediction with momentary observation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6467–6476, 2022.

[47] Jianhua Sun, Yuxuan Li, Hao-Shu Fang, and Cewu Lu. Three steps to multimodal trajectory prediction: Modality clustering, classification and synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13250–13259, 2021.

[48] Conghao Wong, Beihao Xia, Ziming Hong, Qinmu Peng, Wei Yuan, Qiong Cao, Yibo Yang, and Xinge You. View vertically: A hierarchical network for trajectory prediction via fourier spectrums. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 682–700. Springer, 2022.

[49] Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6507, 2022.

[50] Pei Xu, Jean-Bernard Hayet, and Ioannis Karamouzas. Socialvae: Human trajectory prediction using timewise latents. In *European Conference on Computer Vision*, pages 511–528. Springer, 2022.

[51] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 507–523. Springer, 2020.

https://doi.org/10.52202/075280-0633

[52] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021.

[53] Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pages 376–394. Springer, 2022.

[54] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Ben Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, et al. Tnt: Target-driven trajectory prediction. In *Conference on Robot Learning*, pages 895–904. PMLR, 2021.

[55] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134, 2019.

[56] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5826–5835, 2021.

[57] Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8823–8833, 2022.