

SingularTrajectory: Universal Trajectory Predictor Using Diffusion Model

Inhwan Bae, Young-Jae Park and Hae-Gon Jeon*
 AI Graduate School, GIST, South Korea

{inhwanbae, youngjae.park}@gm.gist.ac.kr, haegonj@gist.ac.kr

Abstract

There are five types of trajectory prediction tasks: deterministic, stochastic, domain adaptation, momentary observation, and few-shot. These associated tasks are defined by various factors, such as the length of input paths, data split and pre-processing methods. Interestingly, even though they commonly take sequential coordinates of observations as input and infer future paths in the same coordinates as output, designing specialized architectures for each task is still necessary. For the other task, generality issues can lead to sub-optimal performances. In this paper, we propose SingularTrajectory, a diffusion-based universal trajectory prediction framework to reduce the performance gap across the five tasks. The core of SingularTrajectory is to unify a variety of human dynamics representations on the associated tasks. To do this, we first build a Singular space to project all types of motion patterns from each task into one embedding space. We next propose an adaptive anchor working in the Singular space. Unlike traditional fixed anchor methods that sometimes yield unacceptable paths, our adaptive anchor enables correct anchors, which are put into a wrong location, based on a traversability map. Finally, we adopt a diffusion-based predictor to further enhance the prototype paths using a cascaded denoising process. Our unified framework ensures the generality across various benchmark settings such as input modality, and trajectory lengths. Extensive experiments on five public benchmarks demonstrate that SingularTrajectory substantially outperforms existing models, highlighting its effectiveness in estimating general dynamics of human movements. Code is publicly available at <https://github.com/inhwanbae/SingularTrajectory>.

1. Introduction

Extensive studies of trajectory prediction methods have been conducted in the computer vision field for several decades [20, 52]. They have demonstrated its importance in various applications, including crowd simulation, social robot navi-

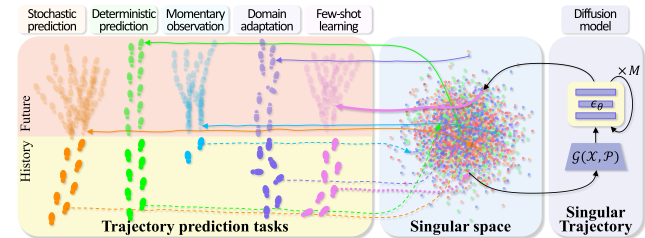


Figure 1. An overview of our SingularTrajectory framework. All relevant human trajectory prediction tasks can be represented in our Singular space, a unified feature embedding space for human dynamics. Using the embedding features, our diffusion-based model for a universal trajectory prediction makes prediction for all the tasks in this same space.

gation, obstacle avoidance and security and surveillance systems, etc. Trajectory prediction takes sequential coordinate values of agents as input and infers their future pathways in common [1, 19, 83]. Such tasks vary depending on the application. The tasks are determined with respect to the number of input/output, data processing, and the use of geological information.

The tasks for trajectory prediction can be categorized into five groups: (1) Stochastic prediction is to predict 20 multi-modal future trajectories from one observation with 8 frames [19]. Each future trajectory consists of 12 frames. (2) Deterministic prediction takes one observation with 8 frames, but infers only one future trajectory with 12 frames [1]. (3) The momentary observation uses only two frames to predict 20 multi-modal future paths with 12 frames [77]. (4) Domain adaptation splits training data with respect to places in datasets, trains a model on one place, and then checks the transferability to other places [93]. (5) The few-shot task only uses partial data to build a dataset-efficient model [53], as illustrated in Fig. 1. Until now, each specialized architecture for a task type has provided performance gains.

However, two questions arise. First, why do state-of-the-art models for one task undergo significant performance drops when applied to other trajectory prediction tasks? Second, is it feasible to design a general predictor that works across the five tasks?

As answers to these questions, we present a universal tra-

*Corresponding author

jectory predictor to achieve generality in predictions, named SingularTrajectory. The main idea is to unify the modalities of human dynamic representations across the five tasks. To do this, we first introduce a Singular space, an embedding space consisting of representative motion patterns for each task. The motion patterns play a role in the basis function for pedestrian movements, and are extracted using Singular Value Decomposition (SVD). They are then projected onto Singular space.

We next propose an environment-adaptive anchor working in the Singular space. Unlike traditional fixed anchor methods [6, 29] that sometimes fail to handle different target data distribution, our adaptive anchor is able to correct prototype paths from the adaptive anchor in Singular space if they are put into the wrong locations, based on an input traversability map. Lastly, we generate socially-acceptable future trajectories for all agents in scenes with a diffusion-based predictor which denoises residuals of perturbed prototype paths. Thanks to the cascaded denoising process of diffusion models, we refine the prototype paths. Here, historical pathways, agent interactions, and environmental information are provided as conditions to guide them in the Markov chain of the denoising diffusion processes.

Experimental results demonstrate that our SingularTrajectory can successfully represent pedestrian motion dynamics, and significantly improve prediction accuracy for the five tasks on challenging public benchmark datasets.

2. Related Works

We review previous studies on trajectory prediction that have attempted to address various benchmark scenarios.

2.1. Pedestrian Trajectory Prediction

Ways of predicting pedestrian future trajectories have been studied for a long time in the computer vision field. Pioneering works [20, 52, 60, 95] model an invisible social norm based on motion dynamics as an energy minimization problem. Introducing recurrent neural networks [1, 19, 64, 97] achieves significant improvements by providing highly sequential representations of high-level shape of paths. These methods determine the most probable path, called deterministic trajectory prediction. The following works model mutual influences among agents using attention mechanisms [17, 23, 64, 83], graph convolutional networks [2, 9, 27, 41, 53, 54, 75], graph attention networks [5, 22, 28, 38, 39, 66, 82], and transformers [4, 18, 55, 68, 86, 87, 97, 98]. Additional visual information allows us to leverage environmental constraints from traversability maps [13–15, 28, 37, 46, 47, 49, 50, 63, 65, 74, 75, 79, 81, 85, 94, 99, 102]. Depending on the constraints, predictors take either recurrent [1, 8, 10, 11, 18, 19, 30, 43, 44, 51, 56, 61, 64, 92, 100, 101] or simultaneous approaches [2, 3, 35, 53, 66, 67, 89] to extrapolate the future pathways.

Meanwhile, with the success of generative models, the importance of multimodality has begun to emerge, called stochastic trajectory prediction. Stochastic prediction enables us to consider all of an agent’s possible future pathways. For example, an agent at a crossroads may either walk straight or turn left/right. Here, the stochastic prediction infers all potential future modes. This approach has become mainstream in this field. Starting from Social-GAN [19], a bivariate Gaussian distribution [1, 2, 34, 53, 66, 69, 70, 93, 96, 97], generative adversarial network [14, 19, 22, 28, 36, 40, 63, 74, 78, 102], and conditional variational autoencoder [7, 10, 23, 30, 31, 33, 45, 64, 76, 84, 88, 91] have been adopted for stochastic trajectory prediction. Anchor-conditioned methods can explicitly represent different modalities by prototyping possible paths [6, 29]. Most recently, diffusion-based models have revealed their tremendous representation capacities in numerous tasks [16, 21, 57, 71–73], proving its potential for stochastic trajectory prediction [18, 25, 48, 62]. In this study, we take full advantage of both the anchor-conditioned approach and the diffusion-based model to achieve the explainability and generalizability of the trajectory prediction tasks.

2.2. Various Trajectory Prediction Tasks

Beyond the standard benchmark protocol of stochastic prediction, there are three other variants of this task: momentary observation, domain adaptation, and few-shot learning. Works in [12, 55, 77] only take two frames as input for the momentary trajectory prediction. Multi-task learning, self-supervised learning, and knowledge distillation techniques have been used to extract rich features from the limited data. Another works [24, 93, 103] focuses on domain adaptation across trajectory domains, captured from different surveillance views. A transferable graph neural network is introduced to adaptively learn domain-invariant knowledge. The others [53, 101] adopt few-shot learning for better training efficiency.

Although these works take and infer the sequential coordinates of agent trajectories in common, there is no unified model for all the tasks. Despite the tremendous efforts to design a specialized architecture for one task, they cannot be applied to the other tasks without suffering significant performance drops. In the next section, we will describe how to design a unified architecture, which consistently produces promising results on the five associated tasks.

3. Methodology

We describe how to learn a general representation of human motions. We first define a general trajectory prediction problem in Sec. 3.1 and provide preliminaries of explicit formulations on the SVD and diffusion process in Sec. 3.2. We then introduce a motion feature extraction to build our Singular space, and a projection of any trajectory from each task onto

it in Sec. 3.3. Next, we propose an environment-adaptive anchor using motion vectors and input image in Sec. 3.4. Finally, we present the SingularTrajectory predictor based on the diffusion model in Sec. 3.5.

3.1. Problem Definition

Trajectory prediction aims to predict the future paths of agents based on their historical path and surrounding contexts. Suppose that at each timestamp t , there are N pedestrians in a scene with the 2D spatial coordinate position $\{\mathbf{p}_t^n \in \mathbb{R}^2 | n \in [1, \dots, N]\}$. A pedestrian historical trajectory \mathbf{X}_n over T_{hist} timesteps can be represented as the cumulative coordinates $\mathbf{X}_n = \{\mathbf{p}_t^n | t \in [1, \dots, T_{hist}]\}$. Similarly, future trajectories \mathbf{Y}_n for the time duration T_{fut} to be predicted can be written as $\mathbf{Y}_n = \{\mathbf{p}_t^n | t \in [T_{hist}+1, \dots, T_{hist}+T_{fut}]\}$. The prediction system takes both the historical trajectories for all N people $\mathbf{X} = \{\mathbf{X}_n | n \in [1, \dots, N]\}$ and the scene image map \mathbf{I} for environmental information as input. The deterministic prediction system infers one sequence of the most reliable future trajectory $\hat{\mathbf{Y}} = \{\hat{\mathbf{Y}}_n | n \in [1, \dots, N]\}$. For the stochastic prediction, because of the indeterminacy of the future movements, S multiple pathways for all the N pedestrians $\hat{\mathbf{Y}} = \{\hat{\mathbf{Y}}_n^s | n \in [1, \dots, N], s \in [1, \dots, S]\}$ are generated so that at least one sample is close to the ground-truth trajectory.

3.2. Preliminaries

Singular Value Decomposition. Singular Value Decomposition (SVD) decomposes a matrix into three resultant matrices. Given a matrix \mathbf{A} , its SVD is represented as:

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^\top, \quad (1)$$

where \mathbf{U} is an orthogonal left singular vector matrix, whose columns are eigenvectors of $\mathbf{A} \times \mathbf{A}^\top$. $\mathbf{\Sigma}$ is a diagonal matrix with the singular values of \mathbf{A} , consisting of K non-negative values in descending order. \mathbf{V} is a right singular vector matrix, which is also orthogonal and its columns are the eigenvectors of $\mathbf{A}^\top \times \mathbf{A}$.

To remove the redundant part of the raw data, the truncation technique is often applied to the results after the decomposition. The idea behind the truncated SVD is to approximate the original matrix \mathbf{A} with the lower rank. With the K to determine the number of singular values to retain, $\mathbf{\Sigma}$ can be simplified to $\mathbf{\Sigma}_K$ which contains only the K largest singular values. Similarly, \mathbf{U} and \mathbf{V}^\top are reduced to \mathbf{U}_K and \mathbf{V}_K^\top by keeping the first K columns and rows, respectively. This process eliminates the smallest singular values, which are not needed to express the original data and often correspond to noise or redundant information. This is useful for practical scenarios dealing with large and potentially sparse matrices because we can reconstruct a close approximation of the original data with significantly less storage space.

Diffusion models. The diffusion model operates by transforming a noisy distribution, represented by the noise vector

\mathbf{y}_M , into the desired data \mathbf{y}_0 through a series of M diffusion steps. These steps involve intermediate latent variables $\{\mathbf{y}_m | m \in [1, \dots, M]\}$, and encompass both the diffusion and denoising processes. The diffusion process adds a small amount of noise to data in order to obtain the standard normal distribution $q(\mathbf{y}_M)$ from the distribution $q(\mathbf{y}_0)$ using the Markov chain as:

$$q(\mathbf{y}_{1:M} | \mathbf{y}_0) := \prod_{m=1}^M q(\mathbf{y}_m | \mathbf{y}_{m-1}) \quad (2)$$

$$q(\mathbf{y}_m | \mathbf{y}_{m-1}) := \mathcal{N}(\mathbf{y}_m; \sqrt{1 - \beta_m} \mathbf{y}_{m-1}, \beta_m \mathbf{I})$$

where β_t is a small positive constant and a variance schedule for adding noise. The denoising process uses the \mathbf{y}_m to recover \mathbf{y}_0 with a learnable network as follows:

$$p_\theta(\mathbf{y}_{0:M}) := p(\mathbf{y}_M) \prod_{m=1}^M p_\theta(\mathbf{y}_{m-1} | \mathbf{y}_m), \quad (3)$$

$$p_\theta(\mathbf{y}_{m-1} | \mathbf{y}_m) := \mathcal{N}(\mathbf{y}_{m-1}; \epsilon_\theta(\mathbf{y}_m, m), \beta_m \mathbf{I}).$$

where $\mathbf{y}_M \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is an initial noise sampled from the Gaussian distribution $p(\mathbf{y}_M)$, and θ denotes the learnable parameter of the diffusion model. ϵ_θ is a learnable denoising model of a clean data \mathbf{y}_0 , and a corrupted data \mathbf{y}_m at a step m . The objective is to train the neural network so that the denoising process predicts the true data-generating distribution well. This is often done by maximizing the evidence lower bound, ensuring that the samples generated by the diffusion model are indistinguishable from the real data.

3.3. Unifying the Motion Space

The trajectory prediction model uses a learnable network to capture the relationship in consideration of the input coordinates of pedestrians, input images and output coordinates for each pedestrian. Since expectations for input and output trajectories are different for each associated task (e.g., length and multimodality), they should be viewed as different data spaces even if they use the same coordinate systems. We introduce a method to merge raw data in each space into a Singular space for human motion dynamics.

Singular space construction. To discover motion dynamics from the raw data, we first extract primitive motion features. Inspired by the successful low-rank approximation of raw trajectory data using eigenvectors in [6], we also employ a similar strategy using singular vectors from the truncated SVD to extract principal motion components from the entire training dataset.

First, we cut-off paths of all pedestrians in the dataset into T_{win} lengths through a sliding window to create a total of L gist of trajectory set $\mathbf{A} \in \mathbb{R}^{L \times (2 \times T_{win})}$. Here, \mathbf{A} is a temporary matrix to extract a motion vector from a set of trajectory in the initialization phase. Next, we decompose \mathbf{A} to obtain truncated matrices $\mathbf{U}_K \in \mathbb{R}^{L \times K}$, $\mathbf{\Sigma}_K \in \mathbb{R}^{K \times K}$

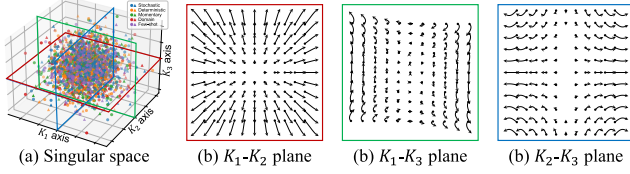


Figure 2. Visualization of the trajectories in Singular space. (a) The circle and triangle markers indicate the history and future trajectory coordinates, respectively. Each color also refers to each associated task. (b-d) Each marker in the Singular space corresponds to each trajectory in raw data. And, the slicing planes, representing a set of arrows, mean human dynamics of straight forward motions and turning motions.

and $\mathbf{V}_K^\top \in \mathbb{R}^{K \times (2 \times T_{win})}$. Here, \mathbf{V}_K^\top is a set of K spatio-temporal motion vectors $\mathbf{v}_k \in \mathbb{R}^{2 \times T_{win}}$ representing the most dominant motion dynamics of pedestrians. Since \mathbf{v}_k are orthogonal to each other, we define a Singular space coordinate system using the K singular vectors as basis vectors for each axis. In Singular space, trajectories \mathbf{A} can be a coordinate of $\mathcal{A} = \mathbf{U}_K \cdot \boldsymbol{\Sigma}_K \in \mathbb{R}^{L \times K}$, indicating that each motion vector has an influence on reconstructing the L trajectories. In other words, we can project the \mathbf{A} into the coordinate in Singular space \mathcal{A} as follows:

$$\mathcal{A} = \mathbf{A} \times (\mathbf{V}_K^\top)^{-1} = \mathbf{A} \times \mathbf{V}_K, \quad (4)$$

where $(\mathbf{V}_K^\top)^{-1}$ can be simplified into \mathbf{V}_K with the property of an orthogonal matrix¹.

In the Singular space, we can now concentrate on the motion flow by using the coefficients of motion patterns, instead of considering every position over time. Note that the results from the SVD depend highly on various factors such as the window size for inputs/outputs, data processing, and the differences in the raw data caused by geological variations. Due to the issue, works using SVD only convert the output space [26, 59, 80] or conduct separate decompositions for each of input and output data spaces [6]. However, in this case, they cause inconsistent representations of motion dynamics, even for the same pedestrian, which leads to a lack of generality. In the next step, we introduce a new method to address this issue.

Projection of any trajectory into Singular space. We aim to express the input/output trajectories \mathbf{X} and \mathbf{Y} of the five associated tasks in Singular space all at once. Our core idea stems from the notion that all pedestrians share the same human motion dynamics and thus will likely show a similar pattern. Starting with the projection function Eq. (4), which is a projection matrix for the fixed trajectory length T_{win} , we extend it to any length T_{hist} and T_{fut} , which varies from task to task. To handle the motion patterns regardless of their lengths, we interpolate $\mathbf{v}_k \in \mathbb{R}^{2 \times T_{win}}$ to $\mathbf{v}_{x,k} \in \mathbb{R}^{2 \times T_{hist}}$.

¹For better understanding, we display the coordinate variable in the Singular space using the calligraphic font

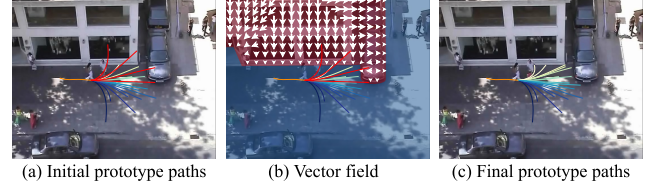


Figure 3. An example of the adaptive anchor generation. (a) The initial prototype anchor \mathcal{P} is placed on the last observation coordinate of a person. In this instance, four prototype paths (highlighted in red) are incorrectly placed at the non-traversable locations. (b) Vector field $\vec{F}_I(x, y)$ is computed to guide toward in the nearest traversable areas. (c) The initial prototype paths are then tailored to the environment using the vector field.

Since the motion pattern can be seen as a 2-dimensional curve, we use Cardinal B-splines to make a transformation matrix $\mathbf{C}_{T_{hist}} \in \mathbb{R}^{(2 \times T_{hist}) \times (2 \times T_{win})}$ using the Irwin-Hall distribution. The constant value C depends only on the length, so $\mathbf{v}_{x,k}$ can be approximated. The trajectory $\mathbf{X} \in \mathbb{R}^{N \times (2 \times T_{hist})}$ is projected to the coordinate $\mathcal{X} \in \mathbb{R}^{N \times (2 \times T_{hist})}$ in Singular space as follows:

$$\mathbf{v}_{x,k} = \mathbf{C}_{T_{hist}} \times \mathbf{v}_k. \quad (5)$$

$$\mathcal{X} = \mathbf{X} \times \mathbf{C}_{T_{hist}} \times \mathbf{V}_K. \quad (6)$$

In the same way as Eq. (6), the trajectory $\mathbf{Y} \in \mathbb{R}^{N \times (2 \times T_{fut})}$ also can be projected to $\mathcal{Y} \in \mathbb{R}^{N \times K}$ using \mathbf{V}_K and $\mathbf{C}_{T_{fut}}$.

Through this process, Singular space can represent a variety of trajectories even with different lengths, including motion vectors for the input/output as well as the task-specific data, as visualized in Fig. 2. Establishing a common ground for the trajectory representation is crucial for our model’s adaptability and robust performances across different trajectory prediction benchmarks. Moreover, the strategy of focusing on the overall motion flow, rather than frame-by-frame coordinates, further improves the model’s capacity to understand and predict socially compliant trajectories in diverse real-world scenarios.

3.4. Adaptive Anchor

Beyond the trajectory data integration, we also introduce how to incorporate environmental contexts into trajectory predictors. To do this, we propose an adaptive anchor which is expressed as a set of motion vectors from input images and consists of prototype paths in Singular space.

Prototype anchor formation. Previous anchor-based human trajectory prediction approaches [6, 29] use a fixed anchor for any pedestrian, and the anchor is refined for output trajectories. Although these methods can explicitly model the multimodality, they sometimes fail to handle the case where the prototype path is put into wrong locations, or blocked by static obstacles. This is because the wrong prototypes are treated as a hard constraint. To avoid this problem,

we use the input image as a traversability map to correct the wrong prototype paths.

We start with the converted future trajectory \mathcal{Y} in the whole dataset \mathbf{Y} using Sec. 3.3. Following [6, 59], the coordinates \mathcal{Y} are then clustered into S centroids, which can be viewed as groups of components representing different multimodal futures. In addition, each centroid in Singular space is used as an initial prototype path \mathcal{P}_s for constructing an anchor $\mathcal{P} \in \mathbb{R}^{S \times K}$. But, these initial prototype paths are still fixed, so they still have a limitation which is unlikely to consider environmental information.

Adaptive anchor generation. To make use of environmental information, we introduce a module to deform the anchor. Using an off-the-shelf semantic segmentation model from [46], we can obtain binary traversable maps \mathbf{I}_{map} from an input image. We then derive a vector field $\vec{F}_I(x, y)$ to fix the wrongly located prototype path by directing it to a nearest traversable regions. If the initial prototype path \mathcal{P}_s is in the wrong place, we deform it by adding the vector fields into the initial anchor until they reach equilibrium states to obtain the final prototype paths \mathcal{P}'_s as follows:

$$\mathcal{P}'_s = \mathcal{P}_s + \vec{F}_I(\mathcal{P}_s \mathbf{V}_K^\top \mathbf{C}_{T_{fut}}^{-1}). \quad (7)$$

Through this process, the prototype path is re-located toward the nearest walkway, as demonstrated in Fig. 3. In other words, the anchor plays a role in the environment-adaptive prototype paths, unlike the existing fixed anchor methods.

Because the prototype paths are in Singular space, the scene image can be implicitly represented as the adaptive anchor. In addition, the scene image can be approximated with the set of motions. By projecting the scene images into the Singular space, our model understands the surrounding environment. This holistic approach, integrating both coordinate and environmental cues, sets the stage for more realistic and reliable trajectory predictions.

3.5. Diffusion-Based SingularTrajectory Model

As the final step, we develop a framework called SingularTrajectory, a unified model that works well across the five tasks. Leveraging the Singular space and the adaptive anchor, our diffusion-based SingularTrajectory model can precisely forecast potential future paths.

Denosing a perturbed trajectory anchor. Unlike previous diffusion-based predictors, which directly forecast the future path from Gaussian noise [18, 25], we devise a stepwise refinement from the adaptive anchor for a realistic trajectory. Here, historical pathways \mathcal{X} , environmental information \mathcal{P} , and agent interactions $\mathcal{G}(\mathcal{X}, \mathcal{P})$ are encoded and used as conditions to guide the denoising processes. To better capture agents interactions \mathcal{G} in the denoising process, we adopt the transformer model. Similar to other transformer-based trajectory predictors [18, 48], our SingularTrajectory

encodes spatio-temporal information to account for agent-agent and agent-environment interactions using \mathcal{X} and \mathcal{P} , respectively. These conditions are then concatenated into one feature vector representation and fed into the diffusion model $\epsilon_\theta(\mathbf{y}_m, m, \mathcal{X}, \mathcal{P}, \mathcal{G})$ to learn, by contrasting the motion patterns from previous diffusion steps to distinguish the added noise, and to close the reality gap by generating socially-acceptable future paths $\hat{\mathbf{Y}}$.

This refinement process works in a cascading manner by $\{\mathcal{P}, \dots, \hat{\mathcal{Y}}\}_{m=1}^M$ as follows:

$$\hat{\mathbf{Y}} = \mathcal{P} + p(\mathbf{y}_M) \prod_{m=1}^M p_\theta(\mathbf{y}_{m-1} | \mathbf{y}_m). \quad (8)$$

By predicting only the residuals $\mathbf{y}_m \in \mathbb{R}^{S \times K}$ to adjust the anchors, the problem is simplified, in that the model is able to use a prior knowledge on the initial state. With this process, we thus ensure more precise and reliable trajectory generations, achieving generality in various environments and applications.

Implementation details. To construct Singular space, we empirically set K to 4. We set $T_{win} = T_{fut} = 12$ in order to prevent information loss due to the approximation of the motion vector during prediction. For an anchor diffusion model, we devise a one-layer transformer for encoding motion and context information, where the dimension is set to 256 with 4-head attention. We set $M = 10$ and schedule the diffusion timesteps following DDIM [72]. To train the SingularTrajectory in an end-to-end manner, we use a mean square error (MSE) as a loss function between the output and a random Gaussian noise for the current iteration. The training is performed with AdamW optimizer [42], with a learning rate of 0.001 and batch size of 512 for 256 epochs. All the experiments are conducted on a single NVIDIA A6000 GPU, which usually takes about an hour to train each scene.

4. Experiments

In this section, we conduct comprehensive experiments to verify the generality of our SingularTrajectory model for the trajectory prediction tasks. We first describe our experimental setup in Sec. 4.1. We then provide comparison results with state-of-the-art models on public benchmark datasets in Sec. 4.2. Finally, we perform an extensive ablation study demonstrating the effect of each component of our method in Sec. 4.3.

4.1. Experimental Setup

Datasets. To compare our SingularTrajectory with state-of-the-art baselines, we conduct quantitative evaluations on two common datasets, ETH [60] and UCY [32], for all the associated tasks. The ETH and UCY datasets consist of various motions of 1,536 pedestrians across five unique scenes: ETH, Hotel, Univ, Zara1 and Zara2. They are recorded in

Domain Adaptation (ADE)	A2*					B2*					C2*					D2*					E2*					*2*
	A2B	A2C	A2D	A2E	AVG	B2A	B2C	B2D	B2E	AVG	C2A	C2B	C2D	C2E	AVG	D2A	D2B	D2C	D2E	AVG	E2A	E2B	E2C	E2D	AVG	AVG
STGCNN [53]	1.83	1.58	1.30	1.31	1.51	3.02	1.38	2.63	1.58	2.15	1.16	0.70	0.82	0.54	0.81	1.04	1.05	0.73	0.47	0.82	0.98	1.09	0.74	0.50	0.83	1.22
PECNet [45]	1.97	1.68	1.24	1.35	1.56	3.11	1.35	2.69	1.62	2.19	1.39	0.82	0.93	0.57	0.93	1.10	1.17	0.92	0.52	0.93	1.01	1.25	0.83	0.61	0.93	1.31
AgentFormer [98]	1.73	1.64	1.87	1.57	1.70	1.49	1.76	2.13	1.73	1.78	1.73	0.19	2.01	1.79	1.43	1.71	2.05	1.92	1.32	1.75	1.96	1.91	1.89	1.91	1.92	1.72
MID [18]	1.26	1.43	1.65	1.43	1.44	1.23	0.87	0.98	0.77	0.96	1.61	0.63	0.81	0.58	0.91	1.42	0.75	0.81	0.74	0.93	1.47	0.68	0.88	0.91	0.98	1.04
EigenTrajectory [6]	0.39	0.71	0.73	<u>0.45</u>	0.57	0.95	0.68	0.51	0.42	0.64	<u>0.94</u>	<u>0.32</u>	0.49	0.37	<u>0.53</u>	1.10	1.03	1.36	0.43	0.98	0.92	0.51	0.62	0.52	0.64	0.67
T-GNN [93]	1.13	1.25	0.94	1.03	1.09	2.54	1.08	2.25	1.41	1.82	0.97	0.54	0.61	0.23	0.59	<u>0.88</u>	0.78	0.59	0.32	<u>0.64</u>	<u>0.87</u>	0.72	0.65	0.34	0.65	0.96
K0 [24]	0.45	0.78	0.68	0.59	0.63	<u>0.87</u>	0.65	<u>0.49</u>	<u>0.36</u>	<u>0.59</u>	1.08	0.46	0.44	<u>0.36</u>	0.59	1.13	0.56	0.59	0.53	0.70	1.13	0.39	0.59	0.48	0.65	0.63
Adaptive [24]	<u>0.36</u>	<u>0.64</u>	<u>0.53</u>	<u>0.45</u>	<u>0.50</u>	0.90	<u>0.62</u>	0.50	<u>0.36</u>	0.60	1.11	0.46	<u>0.48</u>	<u>0.42</u>	0.62	1.11	<u>0.52</u>	0.54	<u>0.47</u>	0.66	1.05	<u>0.34</u>	<u>0.56</u>	<u>0.44</u>	<u>0.60</u>	<u>0.59</u>
SingularTrajectory	0.29	0.59	0.51	0.42	0.45	0.66	0.55	0.45	0.35	0.50	0.79	0.30	<u>0.48</u>	<u>0.36</u>	0.48	0.73	0.29	<u>0.55</u>	<u>0.36</u>	0.49	0.64	0.23	0.55	0.45	0.47	0.48

Domain Adaptation (FDE)	A2*					B2*					C2*					D2*					E2*					*2*
	A2B	A2C	A2D	A2E	AVG	B2A	B2C	B2D	B2E	AVG	C2A	C2B	C2D	C2E	AVG	D2A	D2B	D2C	D2E	AVG	E2A	E2B	E2C	E2D	AVG	AVG
STGCNN [53]	3.24	2.86	2.53	2.43	2.77	5.16	2.51	4.86	2.88	3.85	2.30	1.34	1.74	1.10	1.62	2.21	1.99	1.41	0.88	1.62	2.10	2.05	1.47	1.01	1.66	2.30
PECNet [45]	3.33	2.83	2.53	2.45	2.79	5.23	2.48	4.90	2.86	3.87	2.22	1.32	1.68	1.12	1.59	2.20	2.05	1.52	0.88	1.66	2.10	1.84	1.45	0.98	1.59	2.29
AgentFormer [98]	3.60	3.54	4.52	3.95	3.90	3.76	3.67	5.17	3.42	4.01	3.77	3.49	4.12	3.76	3.79	3.88	3.75	4.27	3.61	3.88	3.82	3.55	3.74	4.32	3.86	3.89
MID [18]	2.42	2.77	3.22	2.77	2.80	2.34	1.66	1.87	1.44	1.83	3.13	1.28	1.74	1.24	1.85	2.73	1.49	1.67	1.54	1.86	2.87	1.41	1.88	1.93	2.02	2.07
EigenTrajectory [6]	0.76	1.43	1.56	0.96	1.18	2.10	1.38	1.07	0.86	1.35	2.01	<u>0.64</u>	1.10	0.82	<u>1.14</u>	2.42	1.03	1.36	0.97	1.44	2.04	<u>0.51</u>	1.29	1.11	1.24	1.27
T-GNN [93]	2.18	2.25	1.78	1.84	2.01	4.15	1.82	4.04	2.53	3.14	<u>1.91</u>	1.12	1.30	0.87	1.30	<u>1.92</u>	1.46	1.25	0.65	<u>1.32</u>	<u>1.86</u>	1.45	1.28	0.72	1.33	1.82
K0 [24]	0.94	1.72	1.51	1.29	1.37	<u>1.81</u>	1.33	<u>1.02</u>	<u>0.75</u>	<u>1.23</u>	2.17	0.91	0.94	0.81	1.21	2.18	0.98	1.20	1.03	1.35	2.12	0.68	1.22	1.03	1.26	1.28
Adaptive [24]	<u>0.69</u>	<u>1.34</u>	<u>1.12</u>	<u>0.91</u>	<u>1.02</u>	1.85	<u>1.27</u>	1.05	0.76	<u>1.23</u>	2.26	0.92	<u>1.03</u>	0.94	1.29	2.23	1.02	1.17	1.00	1.36	2.07	0.63	<u>1.19</u>	<u>0.94</u>	<u>1.21</u>	<u>1.22</u>
SingularTrajectory	0.57	1.19	1.08	0.81	0.91	1.16	1.17	0.97	0.73	1.01	1.36	0.62	1.11	0.79	0.97	1.22	0.52	<u>1.18</u>	<u>0.73</u>	0.92	1.07	0.43	1.15	<u>0.94</u>	0.90	0.94

Table 1. Comparison of our SingularTrajectory with other state-of-the-art methods in domain adaptation task. Following [24, 93], we deterministically predict one future path for all dataset pairs (ADE/FDE). **Bold**: Best, Underline: Second best.

Stochastic	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Social-GAN [19]	0.87/1.62	0.67/1.37	0.76/1.52	0.35/0.68	0.42/0.84	0.61/1.21
STGCNN [53]	0.64/1.11	0.49/0.85	0.44/0.79	0.34/0.53	0.30/0.48	0.44/0.75
STAR [97]	0.57/1.11	0.19/0.37	0.35/0.75	0.26/0.57	0.25/0.58	0.33/0.68
PECNet [45]	0.61/1.07	0.22/0.39	0.34/0.56	0.25/0.45	0.19/0.33	0.32/0.56
MID [18]	0.57/0.93	0.21/0.33	0.29/0.55	0.28/0.50	0.20/0.37	0.31/0.54
LBEBM [58]	0.60/1.06	0.21/0.38	0.28/0.54	0.21/0.39	0.15/0.30	0.29/0.53
Trajectron++ [64]	0.61/1.03	0.20/0.28	0.30/0.55	0.24/0.41	0.18/0.32	0.31/0.52
GroupNet [88]	0.46/0.73	0.15/0.25	0.26/0.49	0.21/0.39	0.17/0.33	0.25/0.44
AgentFormer [98]	0.46/0.80	0.14/0.22	0.25/0.45	<u>0.18/0.30</u>	<u>0.14/0.24</u>	<u>0.23/0.40</u>
GP-Graph [4]	0.43/0.63	0.18/0.30	0.24/0.42	0.17/0.31	0.15/0.29	<u>0.23/0.39</u>
NPSN [5]	<u>0.36/0.59</u>	0.16/0.25	<u>0.23/0.39</u>	<u>0.18/0.32</u>	<u>0.14/0.25</u>	0.21/0.36
EqMotion [90]	0.40/0.61	<u>0.12/0.18</u>	<u>0.23/0.43</u>	<u>0.18/0.32</u>	0.13/0.23	0.21/0.35
EigenTrajectory [6]	<u>0.36/0.53</u>	<u>0.12/0.19</u>	0.24/0.43	0.19/0.33	<u>0.14/0.24</u>	0.21/0.34
SocialVAE [91]	0.41/0.58	0.13/0.19	0.21/0.36	0.17/0.29	0.13/0.22	0.21/0.33
LED [48]	0.39/0.58	0.11/0.17	0.26/0.43	<u>0.18/0.26</u>	0.13/0.22	0.21/0.33
SingularTrajectory	0.35/0.42	0.13/0.19	0.25/0.44	0.19/0.32	0.15/0.25	0.21/0.32

Table 2. Evaluation on the stochastic trajectory prediction task.

surveillance views, and the trajectories are labeled in world coordinates.

Benchmarks. We evaluate our model for the five trajectory prediction tasks: (1) stochastic prediction, (2) deterministic prediction, (3) momentary observation, (4) domain adaptation, and (5) few-shot learning. The stochastic trajectory prediction employs a leave-one-out strategy [5, 19, 22, 53, 66] for training and inference across the five ETH-UCY scenes. In the scenes, the initial 3.2 seconds (equivalent to $T_{hist} = 8$ frames) of all paths are used as input, and the subsequent 4.8 seconds (corresponding to $T_{fut} = 12$ frames) are predicted. The model generates $S = 20$ paths, and one with the lowest error is chosen as the final prediction. In the deterministic prediction, the trajectory prediction methods generate only $S = 1$ path. For the momentary observation, the observation length is $T_{hist} = 2$ whereas still $T_{fut} = 12$. For the domain adaptation, models are trained on each of the five scenes individually, and then evaluated on the other scenes, instead

Deterministic	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
STGCNN [53]	1.04/1.93	0.88/1.63	0.71/1.34	0.66/1.22	0.57/1.06	0.77/1.44
PECNet [45]	1.20/2.73	0.68/1.51	0.78/1.71	0.82/1.85	0.62/1.46	0.82/1.85
AgentFormer [98]	1.67/3.65	1.61/3.59	1.62/3.53	1.85/4.13	1.68/3.74	1.69/3.73
MID [18]	1.42/2.94	0.64/1.30	0.76/1.62	0.74/1.59	0.60/1.31	0.83/1.75
EigenTrajectory [6]	<u>0.93/2.05</u>	<u>0.33/0.64</u>	<u>0.58/1.23</u>	<u>0.45/0.99</u>	<u>0.34/0.75</u>	<u>0.53/1.13</u>
Social-LSTM [11]	1.09/2.35	0.79/1.76	0.67/1.40	0.47/1.00	0.56/1.17	0.72/1.54
Social-GAN [19]	1.13/2.21	1.01/2.18	0.60/1.28	0.42/0.91	0.52/1.11	0.67/1.41
SR-LSTM [100]	1.01/1.93	0.35/0.72	0.66/1.38	0.56/1.23	0.44/0.90	0.60/1.23
STAR-D [97]	0.97/2.00	<u>0.32/0.73</u>	<u>0.56/1.25</u>	<u>0.44/0.96</u>	0.35/0.77	0.53/1.14
Trajectron++ [64]	1.02/2.09	0.33/0.63	0.52/1.16	0.42/0.94	0.32/0.71	<u>0.52/1.11</u>
SingularTrajectory	0.72/1.23	0.27/0.50	0.57/1.12	0.44/0.93	0.35/0.73	0.47/0.90

Table 3. Evaluation on the deterministic trajectory prediction task.

of using the leave-one-out strategy. In the few-shot learning, they are trained using only 10% of the training set, and then assessed on the entire test set. To measure the prediction performances, two common metrics are used for all the tasks, Average Displacement Error (ADE) and Final Displacement Error (FDE), measuring the average and destination error between prediction and its ground-truth path, respectively.

4.2. Evaluation Results

Stochastic prediction task. In Tab. 2, since the ETH/UCY dataset release, many state-of-the-art models have made new records. Among them, our model also achieves performance competitive with the comparison methods. Our model regards historical, social, and environmental conditions of observations and images as the most important factors for successfully generating multimodal trajectories.

Deterministic prediction task. Next, we evaluate our model in the deterministic trajectory prediction task in Tab. 3. Our SingularTrajectory model has the best performance over both these latest state-of-the-art models, and the models optimized for deterministic prediction. In particular, our model

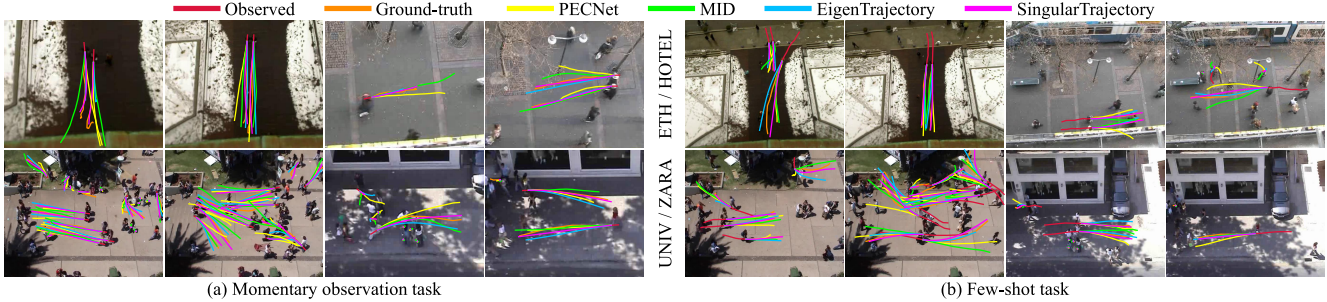


Figure 4. Visualization of prediction results on (a) momentary observation task and (b) few-shot task. To aid visualization, the best trajectory among $S = 20$ samples are reported.

Momentary	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
STGCNN [53]	1.24/2.23	0.77/1.44	0.45/0.81	0.38/0.57	0.35/0.58	0.64/1.13
PECNet [45]	0.63/1.04	0.28/0.53	0.28/0.49	0.25/0.44	0.19/0.34	0.33/0.57
AgentFormer [98]	1.10/2.11	0.50/1.02	0.52/1.10	0.56/1.18	0.43/0.89	0.62/1.26
MID [18]	0.63/1.05	0.29/0.49	0.30/0.56	0.30/0.56	0.22/0.40	0.35/0.61
EigenTrajectory [6]	<u>0.46/0.76</u>	<u>0.17/0.28</u>	<u>0.25/0.44</u>	<u>0.19/0.35</u>	<u>0.15/0.27</u>	<u>0.25/0.42</u>
STT [55]	0.72/1.45	0.48/0.48	0.53/1.09	0.64/1.21	0.44/0.88	0.57/0.93
STT+DTO [55]	0.62/1.22	0.29/0.56	0.58/1.14	0.45/0.98	0.34/0.74	0.46/0.93
MOE-Next [77]	0.71/1.57	0.30/0.58	0.52/1.12	0.38/0.81	0.33/0.73	0.45/0.96
MOE-Traj++ [77]	0.64/1.12	0.20/0.33	0.33/0.62	<u>0.22/0.42</u>	<u>0.17/0.32</u>	<u>0.31/0.56</u>
SingularTrajectory	0.45/0.67	0.18/0.29	0.24/0.43	0.19/0.33	0.17/0.28	0.25/0.40

Table 4. Evaluation on the momentary observation task.

exhibits significant performance improvements in the ETH and HOTEL scenes. These scenes have noisy observation paths, which have a negative impact on the prediction. We are able to achieve accurate predictions by leveraging the overall motion flow, which acts as a low-pass filter over the noisy sequence, in Singular space.

Momentary observation task. In Tab. 4, the state-of-the-art predictors for momentary observation mainly focus on the coordinates themselves, leading to a performance drop, given only two frames. Some models for this task try to bridge these gaps, but have not been sufficient. Fortunately, with the benefit of our Singular Space, even with only a two-frame input, our model can successfully represent the overall long-term flow. Note that our SingularTrajectory with two-frame observation demonstrates the closest performance to state-of-the-art stochastic prediction models with an entire history frame; even this is achieved without any masked trajectory complement or knowledge distillation. Consequently, this allows it to accurately pinpoint the future locations of pedestrians, whose examples are displayed in Fig. 4.

Domain adaptation task. We next evaluate the performance of SingularTrajectory in a domain adaptation task. For simplicity, the ETH, HOTEL, UNIV, ZARA1, and ZARA2 scenes are denoted as A, B, C, D, and E, respectively. For example, ‘A2B’ means that a model is trained on the ETH scene and tested on the HOTEL scene. As demonstrated in Tab. 1, our SingularTrajectory model shows the performance nearly equivalent to those of models specifically designed for deterministic prediction. Particularly,

Few-Shot	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
STGCNN [53]	0.89/1.63	1.22/2.48	0.90/1.61	0.68/1.25	1.36/2.12	1.01/1.82
PECNet [45]	0.72/1.46	0.29/0.53	0.58/0.93	<u>0.27/0.44</u>	0.22/0.38	0.41/0.75
AgentFormer [98]	1.60/2.65	1.02/1.64	1.13/1.90	1.19/2.01	1.08/1.59	1.20/1.96
MID [18]	0.57/0.92	0.21/0.33	0.32/0.60	<u>0.27/0.49</u>	0.24/0.42	0.32/0.55
EigenTrajectory [6]	0.39/0.64	0.13/0.21	0.25/0.43	0.21/0.39	0.15/0.27	<u>0.23/0.39</u>
SingularTrajectory	0.35/0.46	0.14/0.21	0.26/0.44	0.21/0.36	<u>0.18/0.31</u>	0.23/0.35

Table 5. Evaluation on the few-shot trajectory prediction task.

our model produces impressive results in the challenging B2A, C2A, D2A and E2A scenarios where even models specialized in domain adaptation fail. Our SingularTrajectory framework specializes in learning general human motions, and is not limited to a specific domain, enabling accurate predictions even in extreme cases.

Few-shot task. Finally, the results for the few-shot task are reported in Tab. 5, whose examples are in Fig. 4. As expected, our SingularTrajectory significantly outperforms the existing models. With limited data, the existing works, except the diffusion-based model, tend to easily overfit. In contrast, our methods make it possible to explicitly designate future prototype paths, while taking full advantage of the expression ability of the diffusion model. In particular, even with only 10% of the data, our model achieves substantial improvements with respect to both data efficiency and performance. Figure 5 illustrates several cases where there are differences between the predictions of SingularTrajectory and other comparison methods. Previous works often show significant performance drops in other associated tasks compared to stochastic prediction tasks. In contrast, our model consistently predicts the best trajectories across multiple tasks.

4.3. Ablation Studies

The number of motion vectors K . First, we conduct a component study by varying the dimension K of Singular space in Tab. 6. To find the best number of singular vectors in general, we carry out experiments across all five tasks using ZARA1 scene where there are both human-environmental and human-human interactions, following [93]. As the number of motion vectors K increases, more detailed movements

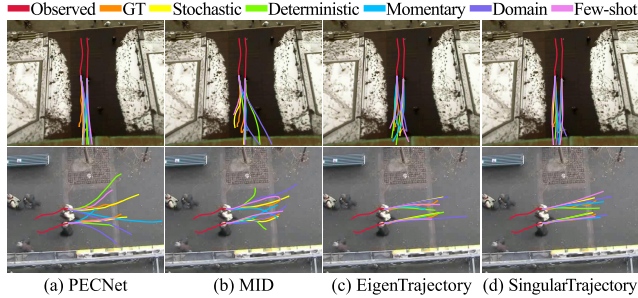


Figure 5. Visualization of prediction consistency across five tasks. The more consistent the prediction is the better.

K	Deterministic	Stochastic	Momentary	Domain	Few-shot	Average
1	0.52 / 1.03	0.30 / 0.55	0.31 / 0.58	0.51 / 1.07	0.34 / 0.61	0.40 / 0.77
2	0.51 / 1.01	<u>0.20 / 0.32</u>	<u>0.20 / 0.33</u>	0.50 / 1.06	0.23 / <u>0.37</u>	0.33 / 0.62
3	0.45 / 0.93	0.19 / 0.33	<u>0.20 / 0.33</u>	<u>0.47 / 1.03</u>	<u>0.22 / 0.38</u>	<u>0.31 / 0.60</u>
4	<u>0.44 / 0.93</u>	0.19 / 0.32	0.19 / 0.33	<u>0.47 / 1.03</u>	0.21 / 0.36	0.30 / 0.59
5	0.43 / 0.94	0.19 / 0.33	<u>0.20 / 0.34</u>	0.46 / 1.04	<u>0.22 / 0.37</u>	0.30 / 0.60
6	0.43 / 0.94	0.19 / 0.32	<u>0.20 / 0.34</u>	0.46 / 1.04	<u>0.22 / 0.37</u>	0.30 / 0.60

Table 6. Ablation study on the Singular space dimension K .

Adoption	Deterministic	Stochastic	Momentary	Domain	Few-shot	Average
Direct	0.75 / 1.47	0.27 / 0.48	0.31 / 0.54	0.78 / 1.60	0.27 / 0.48	0.48 / 0.91
Initial	<u>0.47 / 0.95</u>	<u>0.22 / 0.40</u>	<u>0.22 / 0.39</u>	<u>0.50 / 1.17</u>	<u>0.23 / 0.40</u>	<u>0.33 / 0.66</u>
Residual	0.44 / 0.93	0.19 / 0.32	0.19 / 0.33	0.47 / 1.03	0.21 / 0.36	0.30 / 0.59
Independent	0.46 / 0.94	0.21 / 0.39	0.21 / 0.39	0.49 / 1.16	0.21 / 0.39	0.32 / 0.65
Jointly	0.44 / 0.93	0.19 / 0.32	0.19 / 0.33	0.47 / 1.03	0.21 / 0.36	0.30 / 0.59

Table 7. Ablation study on the adoption of diffusion model.

are captured. In contrast, when using the smaller K , it compresses the space, mainly to represent the overall motion flow. The performance tends to plateau when K is larger than 3, as long as K is not too small to cover most movements. We set $K = 4$ as the dimension for the Singular space because it shows the most effective prediction results across all tasks.

Trajectory denoising methods. Next, we evaluate three types of diffusion models for the trajectory prediction tasks, as shown in Tab. 7. First, we use a basic model similar to MID [18], which directly denoises from a Gaussian noise to a trajectory. This method fails to achieve good performances. The use of an anchor as an intermediate state, similar to LED [48], which reduces the denoising steps by skipping the initial denoising steps, seems to validate its generality. This demonstrates that our adaptive anchor can function as a good initializer for the diffusion model, particularly showing nearly identical outcomes in the stochastic, momentary observation, and few-shot tasks when predicting a multimodal path. However, we confirm that our scheme, which denoises only a residual by adding perturbation to a prototype path in Fig. 6, showcases the best performance in all the tasks. Additionally, compared to refining the prototype paths independently, regarding them as a batch dimension, our model can accurately predict the future when prototype paths are jointly refined.

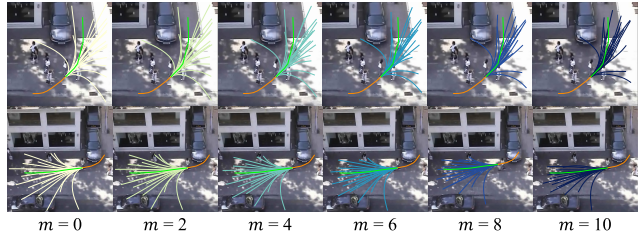


Figure 6. Visualization of anchor refinement. The denoising process progressively refines the prototype paths at each diffusion step m .

M	Deterministic	Stochastic	Momentary	Domain	Few-shot	Average
1	0.45 / 0.95	0.20 / 0.34	<u>0.20 / 0.35</u>	0.48 / 1.05	0.21 / 0.37	0.31 / 0.61
2	0.44 / 0.93	0.19 / <u>0.33</u>	<u>0.20 / 0.34</u>	0.47 / <u>1.04</u>	0.20 / 0.36	0.30 / <u>0.60</u>
5	0.44 / 0.93	0.19 / <u>0.33</u>	<u>0.20 / 0.34</u>	0.47 / <u>1.04</u>	0.20 / 0.36	0.30 / <u>0.60</u>
10	0.44 / 0.93	0.19 / 0.32	0.19 / 0.33	0.47 / 1.03	0.21 / 0.36	0.30 / 0.59
25	0.44 / 0.93	0.19 / 0.32	<u>0.20 / 0.34</u>	0.47 / 1.03	0.20 / 0.36	0.30 / <u>0.60</u>

Table 8. Ablation study on the diffusion steps M .

Diffusion steps M . Lastly, we check how many steps in the diffusion model are needed for the cascaded refinement of the adaptive anchor. In Tab. 8, we confirm that the best performance comes from $M = 10$. This is because the DDIM scheduler accelerates its convergence and the prototype path provides a rough initial trajectory, and so fewer denoising steps are sufficient. However, as the number of denoising steps increases, the information in the initial prototype path becomes attenuated due to the noise. As a result, we observe a slight decrease in performance.

5. Conclusion

In this study, we introduce SingularTrajectory, a universal trajectory predictor model for all related trajectory prediction tasks. By unifying trajectory modalities into one Singular space, our approach standardizes trajectory data with shared motion dynamics, which eliminates the need for task-specific adjustments. The incorporation of an adaptive anchor system further personalizes the prototype paths, allowing them to interpret and adapt to environmental contexts and enhancing the reliability of the trajectory prediction. By successfully incorporating Singular space into the diffusion model, our SingularTrajectory framework successfully achieves the state-of-the-art results across five different benchmarks. This establishes SingularTrajectory itself as a general solution that covers a multitude of scenarios.

Acknowledgement This research was supported by 'Project for Science and Technology Opens the Future of the Region' program through the INNOPOLIS FOUNDATION funded by Ministry of Science and ICT (Project Number: 2022-DD-UP-0312), Vehicles AI Convergence Research & Development Program through the National IT Industry Promotion Agency of Korea (NIPA) funded by the Ministry of Science and ICT (No.S1602-20-1001), and the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST)).

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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2, 6
- [2] Inhwon Bae and Hae-Gon Jeon. Disentangled multi-relational graph convolutional network for pedestrian trajectory prediction. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021. 2
- [3] Inhwon Bae and Hae-Gon Jeon. A set of control points conditioned pedestrian trajectory prediction. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2023. 2
- [4] Inhwon Bae, Jin-Hwi Park, and Hae-Gon Jeon. Learning pedestrian group representations for multi-modal trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2, 6
- [5] Inhwon 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 (CVPR)*, 2022. 2, 6
- [6] Inhwon Bae, Jean Oh, and Hae-Gon Jeon. EigenTrajectory: Low-rank descriptors for multi-modal trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2, 3, 4, 5, 6, 7
- [7] Apratim Bhattacharyya, Michael Hanselmann, Mario Fritz, Bernt Schiele, and Christoph-Nikolas Straehle. Conditional flow variational autoencoders for structured sequence prediction. *arXiv preprint arXiv:1908.09008*, 2020. 2
- [8] Niccoló Bisagno, Bo Zhang, and Nicola Conci. Group lstm: Group trajectory prediction in crowded scenarios. In *Proceedings of the European Conference on Computer Vision Workshop (ECCVW)*, 2018. 2
- [9] Guangyi Chen, Junlong Li, Jiwen Lu, and Jie Zhou. Human trajectory prediction via counterfactual analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2
- [10] 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 (ICCV)*, 2021. 2
- [11] Guangyi Chen, Zhenhao Chen, Shunxing Fan, and Kun Zhang. Unsupervised sampling promoting for stochastic human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2
- [12] Sourav Das, Guglielmo Camporese, and Lamberto Ballan. Distilling knowledge for short-to-long term trajectory prediction. *arXiv preprint arXiv:2305.08553*, 2023. 2
- [13] Patrick Dendorfer, Aljosa Osep, and Laura Leal-Taixe. Goalgan: Multimodal trajectory prediction based on goal position estimation. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2020. 2
- [14] Patrick Dendorfer, Sven Elflein, and Laura Leal-Taixé. Mrgan: A multi-generator model preventing out-of-distribution samples in pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2
- [15] Nachiket Deo and Mohan M. Trivedi. Trajectory forecasts in unknown environments conditioned on grid-based plans. *arXiv preprint arXiv:2001.00735*, 2020. 2
- [16] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2021. 2
- [17] Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural Networks*, 108:466–478, 2018. 2
- [18] 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 (CVPR)*, 2022. 2, 5, 6, 7, 8
- [19] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 6
- [20] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995. 1, 2
- [21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2020. 2
- [22] 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 (ICCV)*, 2019. 2, 6
- [23] Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2
- [24] Boris Ivanovic, James Harrison, and Marco Pavone. Expanding the deployment envelope of behavior prediction via adaptive meta-learning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 2, 6
- [25] Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiodiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 5
- [26] Dongkwon Jin, Wonhui Park, Seong-Gyun Jeong, Heeyeon Kwon, and Chang-Su Kim. Eigenlanes: Data-driven lane descriptors for structurally diverse lanes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 4
- [27] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of*

- the International Conference on Learning Representations (ICLR)*, 2017. 2
- [28] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. Social-bi-gat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2019. 2
- [29] Parth Kothari, Brian Siffringer, and Alexandre Alahi. Interpretable social anchors for human trajectory forecasting in crowds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 4
- [30] 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 (CVPR)*, 2022. 2
- [31] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B. Choy, Philip H. S. Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
- [32] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3): 655–664, 2007. 5
- [33] Jiachen Li, Hengbo Ma, and Masayoshi Tomizuka. Conditional generative neural system for probabilistic trajectory prediction. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. 2
- [34] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2020. 2
- [35] Shijie Li, Yanying Zhou, Jinhui Yi, and Juergen Gall. Spatial-temporal consistency network for low-latency trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2
- [36] Yuke Li. Which way are you going? imitative decision learning for path forecasting in dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2
- [37] Junwei Liang, Lu Jiang, Juan Carlos Nieves, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2
- [38] Junwei Liang, Lu Jiang, and Alexander Hauptmann. Simaug: Learning robust representations from simulation for trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2
- [39] Junwei Liang, Lu Jiang, Kevin Murphy, Ting Yu, and Alexander Hauptmann. The garden of forking paths: Towards multi-future trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2
- [40] Rongqin Liang, Yuanman Li, Xia Li, Yi Tang, Jiantao Zhou, and Wenbin Zou. Temporal pyramid network for pedestrian trajectory prediction with multi-supervision. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021. 2
- [41] Yuejiang Liu, Qi Yan, and Alexandre Alahi. Social nce: Contrastive learning of socially-aware motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2
- [42] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 5
- [43] Yuexin Ma, Xinge Zhu, Xinjing Cheng, Ruigang Yang, Jiming Liu, and Dinesh Manocha. Autotrajectory: Label-free trajectory extraction and prediction from videos using dynamic points. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2
- [44] Takahiro Maeda and Norimichi Ukita. Fast inference and update of probabilistic density estimation on trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2
- [45] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 6, 7
- [46] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From goals, waypoints & paths to long term human trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 5
- [47] Huynh Manh and Gita Alaghband. Scene-1stm: A model for human trajectory prediction. *arXiv preprint arXiv:1808.04018*, 2018. 2
- [48] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 5, 6, 8
- [49] Francesco Marchetti, Federico Becattini, Lorenzo Seidenari, and Alberto Del Bimbo. Mantra: Memory augmented networks for multiple trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2
- [50] Francesco Marchetti, Federico Becattini, Lorenzo Seidenari, and Alberto Del Bimbo. Multiple trajectory prediction of moving agents with memory augmented networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 2
- [51] Francesco Marchetti, Federico Becattini, Lorenzo Seidenari, and Alberto Del Bimbo. Smemo: Social memory for trajectory forecasting. *arXiv preprint arXiv:2203.12446*, 2022. 2
- [52] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1, 2
- [53] Abdullh Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal

- graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 6, 7
- [54] Abdullah Mohamed, Deyao Zhu, Warren Vu, Mohamed Elhoseiny, and Christian Claudel. Social-implicit: Rethinking trajectory prediction evaluation and the effectiveness of implicit maximum likelihood estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2
- [55] 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 (CVPR)*, 2022. 2, 7
- [56] Ingrid Navarro and Jean Oh. Social-patternn: Socially-aware trajectory prediction guided by motion patterns. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022. 2
- [57] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021. 2
- [58] Bo Pang, Tianyang Zhao, Xu Xie, and Ying Nian Wu. Trajectory prediction with latent belief energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 6
- [59] Wonhui Park, Dongkwon Jin, and Chang-Su Kim. Eigencontours: Novel contour descriptors based on low-rank approximation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 4, 5
- [60] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2009. 2, 5
- [61] Mark Pfeiffer, Giuseppe Paolo, Hannes Sommer, Juan I. Nieto, Roland Y. Siegwart, and César Cadena. A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018. 2
- [62] Davis Rempe, Zhengyi Luo, Xue Bin Peng, Ye Yuan, Kris Kitani, Karsten Kreis, Sanja Fidler, and Or Litany. Trace and pace: Controllable pedestrian animation via guided trajectory diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2
- [63] 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 (CVPR)*, 2019. 2
- [64] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 6
- [65] Nasim Shafiee, Taskin Padir, and Ehsan Elhamifar. Introvert: Human trajectory prediction via conditional 3d attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2
- [66] 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 (CVPR)*, 2021. 2, 6
- [67] Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Fang Zheng, Nanning Zheng, and Gang Hua. Social interpretable tree for pedestrian trajectory prediction. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2022. 2
- [68] Liushuai Shi, Le Wang, Sanping Zhou, and Gang Hua. Trajectory unified transformer for pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2
- [69] Xiaodan Shi, Xiaowei Shao, Zipei Fan, Renhe Jiang, Haoran Zhang, Zhiling Guo, Guangming Wu, Wei Yuan, and Ryosuke Shibasaki. Multimodal interaction-aware trajectory prediction in crowded space. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 2
- [70] Xiaodan Shi, Xiaowei Shao, Guangming Wu, Haoran Zhang, Zhiling Guo, Renhe Jiang, and Ryosuke Shibasaki. Social-dpf: Socially acceptable distribution prediction of futures. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021. 2
- [71] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2015. 2
- [72] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5
- [73] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2019. 2
- [74] 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 (CVPR)*, 2020. 2
- [75] Jianhua Sun, Qinhong Jiang, and Cewu Lu. Recursive social behavior graph for trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2
- [76] 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 (ICCV)*, 2021. 2
- [77] 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 (CVPR)*, 2022. 1, 2, 7
- [78] Jianhua Sun, Yuxuan Li, Liang Chai, and Cewu Lu. Stimulus verification is a universal and effective sampler in multi-modal human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2
- [79] Chaofan Tao, Qinong Jiang, and Lixin Duan. Dynamic and static context-aware lstm for multi-agent motion prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2
- [80] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1991. 4
- [81] Daksh Varshneya and G. Srinivasaraghavan. Human trajectory prediction using spatially aware deep attention models. *arXiv preprint arXiv:1705.09436*, 2017. 2
- [82] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 2
- [83] Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018. 1, 2
- [84] Chuhua Wang, Yuchen Wang, Mingze Xu, and David J Crandall. Stepwise goal-driven networks for trajectory prediction. *IEEE Robotics and Automation Letters (RA-L)*, 2022. 2
- [85] Yuning Wang, Pu Zhang, Lei Bai, and Jianru Xue. Fend: A future enhanced distribution-aware contrastive learning framework for long-tail trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2
- [86] Song Wen, Hao Wang, and Dimitris Metaxas. Social ode: Multi-agent trajectory forecasting with neural ordinary differential equations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2
- [87] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2
- [88] 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 (CVPR)*, 2022. 2, 6
- [89] Chenxin Xu, Weibo Mao, Wenjun Zhang, and Siheng Chen. Remember intentions: Retrospective-memory-based trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
- [90] Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 6
- [91] Pei Xu, Jean-Bernard Hayet, and Ioannis Karamouzas. Socialvae: Human trajectory prediction using timewise latents. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2, 6
- [92] Yi Xu, Jing Yang, and Shaoyi Du. Cf-lstm: Cascaded feature-based long short-term networks for predicting pedestrian trajectory. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 2
- [93] Yi Xu, Lichen Wang, Yizhou Wang, and Yun Fu. Adaptive trajectory prediction via transferable gnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 6, 7
- [94] Hao Xue, Du Q Huynh, and Mark Reynolds. Ss-lstm: A hierarchical lstm model for pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2018. 2
- [95] Kota Yamaguchi, Alexander C Berg, Luis E Ortiz, and Tamara L Berg. Who are you with and where are you going? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2
- [96] Yu Yao, Ella Atkins, Matthew Johnson-Roberson, Ram Vasudevan, and Xiaoxiao Du. Bitrap: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation. *IEEE Robotics and Automation Letters (RA-L)*, 2021. 2
- [97] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 6
- [98] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 6, 7
- [99] Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2
- [100] Pu Zhang, Wanli Ouyang, Pengfei Zhang, Jianru Xue, and Nanning Zheng. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 6
- [101] He Zhao and Richard P. Wildes. Where are you heading? dynamic trajectory prediction with expert goal examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2
- [102] 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 (CVPR)*, 2019. 2
- [103] Chunyu Zhi, Huaijiang Sun, and Tian Xu. Adaptive trajectory prediction without catastrophic forgetting. *The Journal of Supercomputing*, 2023. 2