NEURALIZED MARKOV RANDOM FIELD FOR INTERACTION-AWARE STOCHASTIC HUMAN TRAJECTORY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Interactive human motions and the continuously changing nature of intentions pose significant challenges for human trajectory prediction. In this paper, we present a neuralized Markov random field (MRF)-based motion evolution method for probabilistic interaction-aware human trajectory prediction. We use MRF to model each agent's motion and the resulting crowd interactions over time, hence is robust against noisy observations and enables group reasoning. We approximate the modeled distribution using two conditional variational autoencoders (CVAEs) for efficient learning and inference. Our proposed method achieves state-of-the-art performance on ADE/FDE metrics across two dataset categories: overhead datasets ETH/UCY, SDD, and NBA, and ego-centric JRDB. Furthermore, our approach allows for real-time stochastic inference in bustling environments, making it well-suited for a 30FPS video setting. We will open-source our codes upon paper acceptance.

1 INTRODUCTION

Trajectory prediction aims to estimate the most possible future states of one or multiple interacting agents conditioned on past observations. It also serves as the underlying foundation for various intelligent systems, including autonomous driving (Jiang et al., 2023; Girase et al., 2021), human-robot interaction systems such as robot guidance (Aghili, 2012; Rudenko et al., 2020), service robots (Lee et al., 2018; Sun et al., 2018), *etc*. One of the key challenges in this task is to capture the inherent multimodal behaviors of agents, which are influenced by environmental obstacles, social rules, self-intention changes, interactive dynamics, *etc*. Compared to vehicle trajectories, human motions exhibit more short-term uncertainties due to fewer lane constraints in open spaces, and smaller momentum makes the movements more agile and flexible, rendering them difficult to predict.

Current prediction works mainly address such multi-agents entangling problems from two aspects: i) stochastic algorithms that generate multiple future trajectories simultaneously (Gupta et al., 2018; Salzmann et al., 2020; Gu et al., 2022) to cover possible intention changes; ii) interaction modeling through graphs at various scales (Li et al., 2020; Xu et al., 2022a), socially-aware operations (Alahi et al., 2016; Xu et al., 2018), attention mechanism (Yu et al., 2020; Girgis et al., 2021), *et al*. Furthermore, these two directions are usually combined for further performance improvements (Mangalam et al., 2020; Sadeghian et al., 2019; Yuan et al., 2021). Most of these methods assume that the interaction pattern can be captured and applied to future time, and thus extract human interaction features only from history sequences. Some other works such as latent self-motion (Choi & Hebert, 2006), intended actions with heuristics (Han et al., 2023), crowds as dissipative systems modeling (Bhaskara et al., 2024), *etc.* apply the Markov property in their framework designs to construct temporal and spatial dependencies. Our method differs with an explicit structured modeling of each agent's motion and crowd interactions throughout the entire sequences, employing MRF to iteratively infer the stochastic distribution of future joint configurations.

In this work, we introduce an MRF realized by neural networks for interaction-aware human trajectory prediction, as sketched in Figure 1. We first derive the crowd motion evolution as a probabilistic distribution, which consists of: i) a **Bayesian update term** to predict the next states based on the given observations; and ii) a **transition term** that is further factorized into a **self-evolution term** to

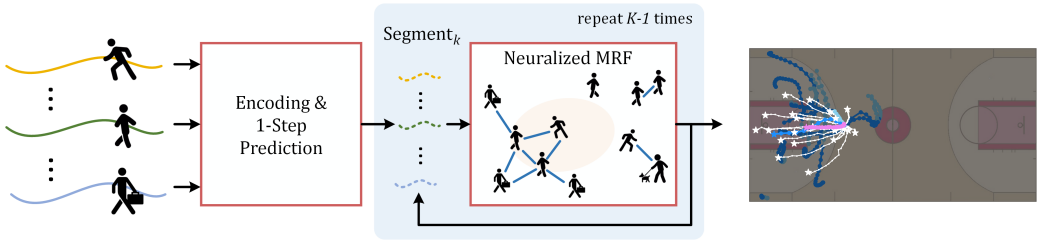


Figure 1: Neuralized MRF for motion and interaction aware trajectory prediction. Red boxes represent computation modules. Our proposed method leverages Markovian motion assumption and explicitly models future transitions and interactions. Through iterative prediction, it produces a concentrated distribution around the ground truth with more accurate best-of-20 predictions.

predict the next states based on their current states, and an **interaction term** to model the spatial relations between agents. Figure 2 sketches the proposed MRF-based evolution. It is built on the assumption that human movements are Markovian up to certain frequencies. By dividing the future joint space configuration (*i.e.*, the collection of individual trajectories) sequences into multiple periods, the resulting trajectory segments form a Markov chain conditioned on past observations. Within each segment, trajectory clips are treated as nodes in a dynamically constructed graph, with edges linked based on a distance threshold to model interactions between nearby agents. Consequently, a joint configuration state can be represented by either one undirected graph or a set of multiple undirected sub-graphs. The internal motion dynamics of the state are explicitly modeled by the MRF, which inherently captures the evolving pair-wise and group-wise relationships for the future.

We approximate the distribution using a neural network framework with two CVAEs, employing variational inference as exact inference is intractable. Our proposed framework is composed of three sub-networks: i) a CVAE for the Bayesian update term; ii) a recurrent CVAE for the transition term comprising the self-evolution and interaction terms; and iii) a neural sampler to generate multi-modal samples from the learned latent distributions. We conduct experiments across four datasets named ETH/UCY, SDD, NBA SportVU, and JRDB to validate the effectiveness of our approach. Through quantitative and qualitative results, we show that our proposed neural interaction-aware MRF can capture dynamic and hierarchical interactions among agents via neuralized potentials for future behaviors. Our approach achieves state-of-the-art performance under ADE/FDE metrics, while yielding real-time inference speed across various environments at the same time. Furthermore, our method is robust across the simulated observation noise, and the learned potentials in MRF can be utilized for additional human-centered scene understanding tasks such as group inference.

Our contributions can be summarized as follows: i) A new MRF-based framework for the human trajectory prediction task. Our MRF explicitly models the agent's motion dynamics and the resulting crowd interactions. ii) Tractable learning and inference of the MRF by introducing a novel and lightweight neural network. iii) The state-of-the-art and time-efficient prediction performance of our approach, along with its robustness under noise disturbance, is demonstrated through evaluations on interaction-rich datasets.

2 Related Works

Interaction Modeling. Previous works have shown various approaches to model the interactions within crowds, utilizing RNNs (Vemula et al., 2018; Zhang et al., 2019), graph networks (Mohamed et al., 2020; Salzmann et al., 2020; Xu et al., 2022a), attention mechanisms (Kosaraju et al., 2019; Mangalam et al., 2020; Kamra et al., 2020), and transformers to capture spatial and temporal dependencies together (Yu et al., 2020; Giuliari et al., 2021; Yuan et al., 2021; Saadatnejad et al., 2023; Shi et al., 2023). Specifically, graph-based networks usually aggregate edge information from neighboring agents and conduct message passing to obtain the final ego-node representations (Salzmann et al., 2020; Hu et al., 2020); while attention and transformer structures attribute varying importance levels for spatially proximal agents and focus on the most relevant ones to assist social-influence counting. Our approach explicitly models the full dynamics of joint space state transitions and interactions with an MRF, and attentions are employed to extract features from the past.

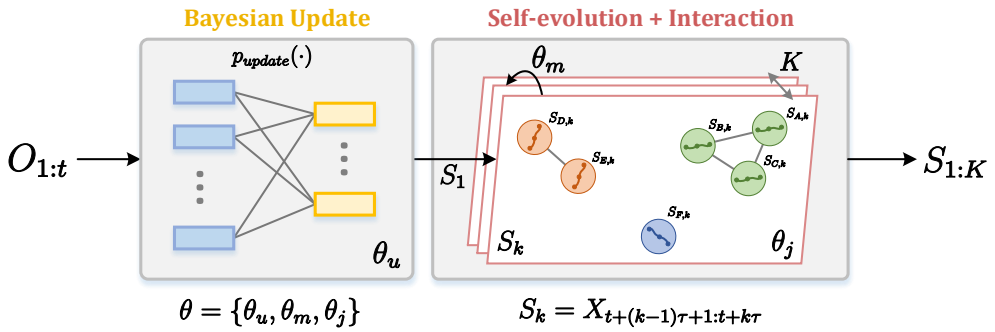


Figure 2: Overview of our MRF-based human motion evolution. Given the history sequence, $O_{1:t}$, the first future segment S_1 is estimated via the Bayesian update module, and future segments $S_{2:K}$ are generated from the self-evolution and interaction module. The Bayesian update, self-evolution and interaction modules are parameterized by $\theta = \{\theta_u, \theta_m, \theta_j\}$.

Stochastic Trajectory Prediction. Aside from deterministic methods (Alahi et al., 2016; Vemula et al., 2018; Zhang et al., 2019), more stochastic trajectory prediction works with deep generative models have been proposed to manage the multi-modalities of human motions. Generative Adversarial Network (GAN) (Gupta et al., 2018; Sadeghian et al., 2019; Dendorfer et al., 2021), Variational Auto-Encoder (VAE) (Mangalam et al., 2020; Lee et al., 2022; Xu et al., 2022c), and Conditional Variational Autoencoder (CVAE) (Lee et al., 2017; Salzmann et al., 2020; Cheng et al., 2021; Xu et al., 2022a; Yue et al., 2022; Mohamed et al., 2020) are several lines, with some variants integrating normalizing flow to implement non-Gaussian priors (Bhattacharyya et al., 2019) or heatmap to utilize rasterized images (Mangalam et al., 2021). Recently, denoising diffusion probabilistic models have also been applied into time series forecasting (Rasul et al., 2021), thereby expanding to trajectory prediction. MID (Gu et al., 2022) formulates the trajectories generation process as a reverse diffusion process by Markov chain with Gaussian transitions. To tackle the sample generation efficiency, LED (Mao et al., 2023) utilizes a learned initializer to produce correlated samples in the reverse process, replacing the Gaussian prior and skipping numerous denoising steps. SingularTrajectory (Bae et al., 2024b) embeds all types of motion patterns into a Singular space for the generality across benchmarks and uses cascaded denoising in its diffusion-based predictor, significantly surpassing MID.

Markov Random Field Applications. As a popular tool for modeling dependent distributions with an undirected graph, the Markov random field (MRF) has been widely applied in tasks such as multi-target tracking (Khan et al., 2004) and location inferring (Mattyus et al., 2015), among others. Recently, NMRF (Guan et al., 2024) has also utilized MRFs to capture complicated pixel relationships for stereo matching. In trajectory prediction tasks, the Markov property-*i.e.*, a pedestrian's future state primarily depends only on its current state-has been implemented for agent-wise self-motion modeling, from early piecewise segment approach (Choi & Hebert, 2006) to more recent FlowMNO (Bhaskara et al., 2024). By using a heuristic that relates coordinates and actions, such as turning right or left, S-T CRF (Han et al., 2023) employs a CRF to model intended actions for future timesteps. Rather than applying the Markov property only to sub-components, we use the MRF to model the full dynamics of state transitions and crowd interactions, providing a neural network-based method for learning and inference.

3 Crowds Motion Evolution with Markov Random Field

3.1 Problem Formulation

Given t frames of observations $O_{1:t} \triangleq \{O_1, O_2, ..., O_t\}$ up to the current time t, the goal of human trajectory prediction is to determine the future joint configuration space of the humans $X_{t+1:t+T} \triangleq \{X_{t+1}, X_{t+2}, ..., X_{t+T}\}$ in the next T time steps. We further define the joint space configuration $X_t \triangleq \{X_{i,t}\}_{i=1}^{M}$ to consist of M individual states.

Although it is possible to solve the human trajectory prediction problem with a deterministic function map $f : O_{1:t} \mapsto X_{t+1:t+T}$, we model a probabilistic distribution $p(X_{t+1:t+T} \mid O_{1:t})$ to enhance the learning of human interactions under noisy observations for future trajectory prediction. Formally, we define the evolution of joint configuration segments as $S = \{S_1, \cdots, S_K \mid S_k \triangleq X_{t+(k-1)\tau+1:t+k\tau}\}$. S is a Markov chain parameterized by θ to indicate that the future prediction segments with an assumed period τ are conditionally independent of the history segments given the current state and transition dynamics hyper-parameter. As a result, we get:

$$\begin{aligned} p(X_{t+1:t+T} \mid O_{1:t}, \theta) &= p(X_{t+1}, X_{t+2}, ..., X_{t+T} \mid O_{1:t}, \theta) \\ &= p(S_1, S_2, ..., S_K \mid O_{1:t}, \theta) \\ &= p(S_K \mid S_{1:K-1}, O_{1:t}, \theta) p(S_{1:K-1} \mid O_{1:t}, \theta) \\ &= p(S_K \mid S_{1:K-1}, O_{1:t}, \theta) \dots p(S_2 \mid S_1, O_{1:t}, \theta) p(S_1 \mid O_{1:t}, \theta) \\ &= p(S_K \mid S_{K-1}, O_{1:t}, \theta) \dots p(S_2 \mid S_1, O_{1:t}, \theta) p(S_1 \mid O_{1:t}, \theta) \\ &= \underbrace{p(S_1 \mid O_{1:t}, \theta)}_{\text{Bayesian update}} \underbrace{\prod_{k=1}^{K-1} p(S_{k+1} \mid S_k, \theta)}_{\text{self-evolution + interaction}}, \end{aligned} \tag{1}$$

where $K = |S| = T//\tau$ is the set size of configuration segments determined by period τ (also referred to as stride in network implementation). θ is typically estimated from observations which encodes Bayesian update θ_u based on past measurements, and system dynamics that include individual predictive motion models θ_m and pairwise interactions θ_j, with visualizations in Figure 2.

3.2 Realization with Conditional Variational Autoencoder

We propose to realize the posterior distribution $p(X_{t+1:t+T} \mid O_{1:t}, \theta)$ with conditional variational autoencoders (CVAEs) since exact inference is computationally intractable. Let Z represent the latent embedding of the CVAE. We define the log posterior as: $\log p(S_1, \cdots, S_K \mid O_{1:t}, \theta) \triangleq \log p_\theta(S \mid O)$ and maximize the log-posterior with the evidential lower bound:

$$\begin{aligned} \log p_\theta(S \mid O) &\geq \mathcal{L}_{ELBO} \\ &= \mathbb{E}_{q_\psi(Z|O,S)}\big[\log p_\phi(S \mid Z, O)\big] - D_{\mathrm{KL}}(q_\psi(Z \mid O, S) \parallel p(Z)), \end{aligned} \tag{2}$$

where $p_\phi(\cdot)$ is the decoder and $q_\psi(\cdot)$ is the encoder parameterized by ϕ and ψ, respectively. The first term is the reconstruction loss and the second term enforces the latent space to be close to the normal distribution. S is the reconstructed output of the decoder and O is the condition of the CVAE. We further assume that the latent embedding Z is conditionally independent of O given S, *i.e.*, $q_\psi(Z \mid O, S) = q_\psi(Z \mid S)$. The CVAE can be trained by minimizing the evidential lower bound on an n-item dataset $\{\mathbb{O}, \mathbb{X}\}$, where $|\mathbb{O}| = |\mathbb{X}| = n$, $\mathbb{O} = \{O_{1:t}\}$ is the set of observations and $\mathbb{X} = \{X_{t+1:t+T}\}$ is the set of ground truth predictions (*cf.* Section 3.4 more the training details).

3.3 Network Architecture

Figure 3 shows an illustration of our network architecture that consists of two CVAEs: 1) *Bayesian Update* and 2) *MRF-based Evolution* to approximate the two factors of the posterior distribution derived in Equation 1.

Bayesian Update CVAE. We design the Bayesian Update CVAE to approximate the Bayesian update factor $p(S_1 \mid O_{1:t}, \theta) = p(S_1 \mid O_{1:t}, \theta_u)$ in the posterior distribution $p_\theta(S \mid O)$. The given observations $O_{1:t}$ are first encoded into time-dependent features using the History Encoder constructed from convolutional layers and the gated recurrent units (GRU), and then fed into a single self-attention layer to output socially-dependent features. We use the socially-dependent features as the condition for our Bayesian Update CVAE. We introduce a Future Encoder which acts as the CVAE encoder to encode the given predicted trajectory $X_{t+1:t+T}$ into the latent embedding $Z \sim \mathcal{N}(\mu, \sigma^2)$ during CVAE training. Finally, we concatenate the socially-dependent features with the latent embedding as the input to a multilayer perceptron-based Update Decoder that reconstructs the first segment $\hat{S}_1$. Note that the Future Encoder and the Update Decoder play the role of the CVAE encoder $q_\psi(Z \mid S)$ and decoder $p_\phi(S \mid Z, O)$ in Equation 2, respectively.

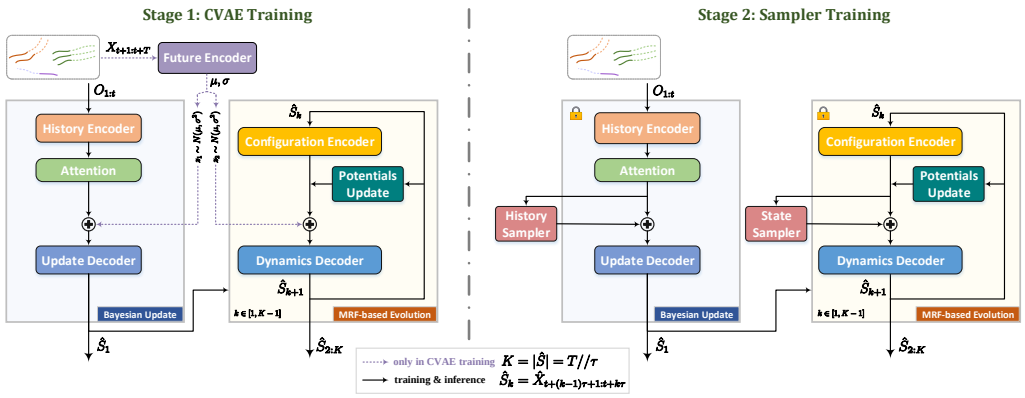


Figure 3: Our network architecture. The Bayesian Update module generates predictions $\hat{S}_1$ for the first defined sequence segment, which then functions as the input into the MRF-based Evolution module to produce succeeding samples $\hat{S}_2 : \hat{S}_K$ iteratively.

MRF-based Evolution CVAE. We propose the MRF-based Evolution CVAE to approximate the self-evolution and interaction factor $\prod_{k=1}^{K-1} p(S_{k+1} \mid S_k, \theta)$ in the posterior distribution $p_\theta(S \mid O)$. We further factorize $p(S_{k+1} \mid S_k, \theta)$ into two parts:

i) The ***space-level distribution*** $p(S_k \mid \theta_j)$ which models the k-th joint space configuration. We are inspired by (Khan et al., 2004; 2005) to construct a Markov random field (MRF) with an undirected graph $G = (V, E)$, where the segments S are the nodes V and the edges E represent the interaction between the segments, *i.e.*:

$$p(S_k \mid \theta) = p(S_k \mid \theta_j) \propto \prod_{ij \in \mathrm{E}} \gamma(S_{i,k}, S_{j,k} \mid \theta_j). \tag{3}$$

$\gamma(\cdot)$ is the pairwise potential of the MRF. We link the edges E dynamically according to pairwise distance in each computation step.

ii) The ***agent-level distribution*** $p(S_{i,k+1} \mid S_{i,k}, \theta_m)$ which models the agent-wise self-transition. Putting the space-level and agent-level distributions together, the self-evolution and interaction factor in Equation 1 becomes:

$$p(S_{k+1} \mid S_k, \theta) \propto \prod_i p(S_{i,k+1} \mid S_{i,k}, \theta_m) \prod_{ij \in \mathrm{E}} \gamma(S_{i,k}, S_{j,k} \mid \theta_j), \tag{4}$$

which we approximate with the MRF-based Evolution CVAE. To this end, we propose a Configuration Encoder which predicts $S_{i,k+1}$ from $S_{i,k}$ to approximate $p(S_{i,k+1} \mid S_{i,k}, \theta_m)$. We further introduce a Potential Update module which calculates pairwise spatial distances by using the last frame of the configuration segment, links edges according to a defined distance threshold d, and accumulates edge features into the connected nodes to approximate the MRF $\prod_{ij \in \mathrm{E}} \gamma(S_{i,k}, S_{j,k} \mid \theta_j)$.

The two features from the Configuration Encoder and Potential Update module are the condition of the CVAE, which are then concatenated with the latent embedding Z and fed into the Dynamics Decoder to output the joint space configuration $\hat{S}_{k+1}$ for the next timestep. These procedures are looped until the final segment prediction is obtained. All sub-modules are implemented using multilayer perceptions. Note that the Future Encoder and the Dynamics Decoder play the role of the CVAE encoder $q_\psi(Z \mid S)$ and decoder $p_\phi(S \mid Z, O)$ in Equation 2, respectively.

3.4 Network Training

We train our network in two stages as illustrated in Figure 3. In Stage 1, we train the CVAEs using the evidential lower bound loss derived in Equation 2. The reconstruction loss is defined as the

minimum error across N samples and the KL-divergence is summed over all segments as follows:

$$\mathcal{L}_{stage1} = \alpha \cdot \min_{N} ||\hat{X}_{t+1:t+T} - X_{t+1:t+T}||_2 + \beta \cdot \sum_{\tau} D_{\mathrm{KL}}(q_\phi \parallel \mathcal{N}(0, I)). \tag{5}$$

Instead of directly sampling from the standard normal distribution $\mathcal{N}(0, I)$, we further train two samplers: a History Sampler and a State Sampler with other modules frozen in Stage 2 for purposive sampling as illustrated in the right-half of Figure 3, where we replace the KL-divergence with discrepancy loss (Bae et al., 2022b) of samples over the encoded joint configuration state:

$$\mathcal{L}_{stage2} = \alpha \cdot \min_{N} ||\hat{X}_{t+1:t+T} - X_{t+1:t+T}||_2 + \lambda \cdot \frac{1}{N} \sum_{i} -\log \min_{\substack{j \in [1,\ldots,N] \\ j \neq i}} ||\hat{Y}_i - \hat{Y}_j||, \tag{6}$$

where $\hat{Y}_i$ is the i-th sampler output among N samples before feature fusion in Stage 2. α, β and λ are hyperparameters to balance the loss terms during training. After training, we only apply the components in Stage 2 for inference.

4 EXPERIMENTS

Datasets. We evaluate our methods on four trajectory prediction datasets: ETH/UCY (Pellegrini et al., 2009; Lerner et al., 2007), Stanford Drone Dataset (SDD) (Robicquet et al., 2016), NBA SportVU Dataset (NBA), and the JackRabbot Dataset and Benchmark (JRDB) (Martin-Martin et al., 2021), covering interaction-rich indoor and outdoor scenarios. **ETH/UCY** dataset contains five subsets: ETH, HOTEL, UNIV, ZARA1, and ZARA2. We follow the leave-one-out approach from (Gupta et al., 2018) with four subsets for training-validation and the remaining subset for testing, predicting the future 12 frames (4.8s) using 8 frames observations (3.2s). **SDD** is a bird's eye view pedestrian dataset collected on a university campus. It is originally annotated in pixels without providing precise projection matrices. We predict the future 12 frames (4.8s) on 8 frames (3.2s) history and report the metrics on both pixel and meter units. **NBA** contains the trajectories of 10 players and the ball in real basketball games collected by the SportVU tracking system. We predict the future 20 frames (4.0s) based on 10 frames (2.0s) history, similar to most other works. The agent's intention changes suddenly and frequently in this dataset, resulting in zigzags, sharp turnarounds, *etc* which increases difficulties compared to the pedestrian datasets. **JRDB** is a large-scale egocentric dataset recorded by a social robot in indoor and outdoor environments with stationary and moving behaviors. We follow the train-validation-test split applied in Social-Transmotion (Saadatnejad et al., 2023) for deterministic prediction. For stochastic situations, splits are in accordance with the official JRDB detection and tracking challenge. Since all labels contained are in camera view and the robot is moving for several scenarios, we utilize the provided IMU information to transform all trajectories into global world coordinates. As officially suggested, the trajectory of future 12 frames is predicted over the past 9 frames with 2.5 frames per second frequency.

Implementation details. We calculate displacement for each configuration segment instead of absolute quantities for robustness: $\hat{S}_{k+1} = \hat{S}_k + \Delta\hat{S}_k$, where $\Delta\hat{S}_k$ is network output. As mentioned in Section 3.3, the History Encoder comprises a 1D convolution layer with a kernel size of three and an output channel of 32, alongside a GRU layer with a hidden embedding dimension of 256, similar to LED (Mao et al., 2023). All the other modules are built using multi-layer perceptions, and the distance threshold d for the potential update module is set to 3m. We set sample number N to 20 in both training and testing stages for stochastic prediction, and train the entire model with AdamW optimizer and StepLR scheduler on one Quadro RTX 8000 GPU. The latent vectors Z_1 and Z_2 have 16 dimensions and the embedding dimension of the configuration encoder is 32. Other network details and dataset-specific hyperparameters are attached in the appendix.

Evaluation metrics. We evaluate our proposed method with ADE and FDE for a single sample in the deterministic case, and with minADE_{20} and minFDE_{20} in the stochastic cases following the previous works (Mangalam et al., 2020; Mao et al., 2023). ADE quantifies the average displacement error between predictions and ground truth across all time steps, while FDE measures the displacement error from the final timestep to the current time. The minADE_{20} and minFDE_{20} metrics calculate the minimum error among 20 predictions, and it is commonly used in evaluating generative models' ability to generate high-quality samples.

Table 1: minADE$_{20}$/minFDE$_{20}$ in **meters** on ETH/UCY datasets. **Bold**/underline indicate the best/second-best results. * denotes results reproduced from official codes. † marks results where the best model was chosen directly based on the test set. We address this issue by following the standard train-val-test split in Social-GAN (Gupta et al., 2018), selecting the best model via the validation set. ‡ is for the future information leakage issue fixed. Inference speed is measured for generating 20 samples under a scene with 57 pedestrians.

Subset	Social-GAN* (Gupta et al., 2018)	Social-STGCNN (Mohamed et al., 2020)	PECNet*† (Mangalam et al., 2020)	STAR (Yu et al., 2020)	Trajectron++*‡ (Salzmann et al., 2020)	SGCN (Shi et al., 2021)	AgentFormer* (Yuan et al., 2021)	NPSN*(best) (Bae et al., 2022b)	GP-Graph (Bae et al., 2022a)	MID*†(DDIM) (Gu et al., 2022)	TUTR*† (Shi et al., 2023)	SingularTrajectory* (Bae et al., 2024b)	**Ours**
ETH	0.77/1.40	0.65/1.10	0.64/1.13	0.36/0.65	0.61/1.03	0.63/1.03	0.46/0.73	0.37/0.60	0.43/0.63	0.46/0.74	0.45/0.67	0.35/0.46	**0.26/0.37**
HOTEL	0.43/0.88	0.50/0.86	0.22/0.38	0.17/0.36	0.20/0.28	0.32/0.55	0.14/0.23	0.16/0.25	0.18/0.30	0.18/0.30	0.14/0.20	0.13/0.20	**0.11/0.17**
UNIV	0.74/1.50	0.44/0.80	0.35/0.57	0.31/0.62	0.30/0.54	0.37/0.70	0.25/0.44	**0.23/0.39**	0.24/0.42	0.25/0.48	0.24/0.44	0.27/0.47	0.28/0.49
ZARA1	0.35/0.70	0.34/0.53	0.25/0.45	0.26/0.55	0.24/0.41	0.29/0.53	0.18/**0.30**	0.18/0.32	**0.17**/0.31	0.23/0.45	0.19/0.36	0.19/0.33	**0.17/0.30**
ZARA2	0.36/0.72	0.31/0.48	0.18/0.31	0.22/0.46	0.17/0.32	0.25/0.45	**0.14/0.24**	**0.14**/0.25	0.15/0.29	0.18/0.35	0.15/0.28	0.15/0.27	**0.14**/0.25
AVG (meters)	0.53/1.04	0.45/0.75	0.33/0.60	0.26/0.53	0.30/0.52	0.37/0.65	0.23/0.39	0.22/0.36	0.23/0.39	0.26/0.46	0.23/0.39	0.22/0.34	**0.19/0.32**
Speed (57 per.)	$\sim 63.2ms$	$\sim 1.8ms$	$\sim 47.6ms$	$\sim 2376ms$	$\sim 254ms$	$\sim 6.5ms$	$\sim 108ms$	$\sim 8.6ms$	$\sim 30ms$	$\sim 400ms$	$\sim 591ms$	$\sim 15.3ms$	$\sim 9.8ms$

Table 2: minADE$_{20}$/minFDE$_{20}$ in **pixels** and **meters** (if available) on SDD dataset. **Bold**/underline indicate the best/second-best results. * denotes results reproduced from official codes.

	Social-GAN (Gupta et al., 2018)	Social-STGCNN (Mohamed et al., 2020)	PECNet (Mangalam et al., 2020)	Trajectron++ (Salzmann et al., 2020)	BiTraP (Yao et al., 2021)	MID* (Gu et al., 2022)	MemoNet (Xu et al., 2022b)	SGNet-ED (Wang et al., 2022)	GP-Graph (Bae et al., 2022a)	Social-VAE (Xu et al., 2022c)	Social-VAE+FPC (Xu et al., 2022c)	TUTR* (Shi et al., 2023)	**Ours**
pixels (meters if avl.)	27.23/41.44	20.8/33.2	9.29/15.93	10.00/17.15 (0.34/0.58)	9.09/16.31 (0.32/0.57)	9.08/17.04	8.56/12.66	9.69/17.01 (0.33/0.58)	9.1/13.8	8.88/14.81 (0.30/0.50)	8.10/11.72 (0.27/0.39)	7.90/12.96	**7.20/11.29** (0.25/0.39)

4.1 Trajectory Prediction Results

ETH/UCY. Table 1 shows the comparison between our model and recent state-of-the-art benchmarks on the ETH/UCY dataset. Since several baselines report their models' test performances by confounding the validation set with the test split, we correct these numbers by re-training with official codes and selecting the best model based on the validation set to ensure fairness. Additionally, recent work such as LMTraj (Bae et al., 2024a) utilizes large language models (LLMs) to capture the multi-modality of human trajectories. However, we omit LMTraj from the baseline since its inference speed is heavily constrained by the response times of models such as GPT-3.5 and GPT-4, and its prediction performance does not surpass SingularTrajectory (Bae et al., 2024b). As shown in the table, our method balances the prediction accuracy and computational efficiency with the best average minADE$_{20}$/minFDE$_{20}$ and an inference speed faster than 100Hz under a 57-pedestrian scenario in the ETH/UCY dataset. It outperforms diffusion-based methods including MID (Gu et al., 2022) and SingularTrajectory (Bae et al., 2024b).

SDD. Table 2 shows the comparisons on the SDD dataset with our proposed model maintaining the best performance. We transform the pixel-unit annotations into meter-unit following the baseline SocialVAE (Xu et al., 2022c) since our method uses a meter-based distance threshold d. It is important to note that due to the lack of precise projection matrices in the SDD dataset, the pixel-to-meter scale is estimated with large variation and unreliability with some values derived from Google Maps and others based on rough guesses (Amirian et al., 2020). We include comparisons on this dataset because it is widely used in the community. However, we must emphasize that pixel errors would only be meaningful if a pixel corresponds to a fixed physical distance across all scenes, which unfortunately is not guaranteed by SDD.

NBA. As presented in Table 3, the proposed method is then evaluated against 10 baselines on the NBA dataset. Our method maintains state-of-the-art accuracy to the best-of-20 samples. We compare the inference time with several baselines where official implementations for this dataset are provided, showing that our model infers trajectories less than one-third of the time while preserving the best prediction performance. Figure 4 visualizes the estimation results from the current best approach and our model. The prediction samples from LED (Mao et al., 2023) are noisy with sev-

Table 3: minADE$_{20}$ and minFDE$_{20}$ in **meters** on NBA dataset. **Bold**/underline indicate the best/second-best results. * denotes results reproduced from official codes.

Time	Social-GAN (Gupta et al., 2018)	Social-STGCNN (Mohamed et al., 2020)	PECNet (Mangalam et al., 2020)	STAR (Yu et al., 2020)	Trajectron++ (Salzmann et al., 2020)	MemoNet (Xu et al., 2022b)	NPSN (Bae et al., 2022b)	GroupNet (Xu et al., 2022a)	MID (Gu et al., 2022)	LED* (Mao et al., 2023)	**Ours**
1.0s	0.41/0.62	0.34/0.48	0.40/0.71	0.43/0.66	0.30/0.38	0.38/0.56	0.35/0.58	0.26/0.34	0.28/0.37	0.18/0.27	**0.16/0.24**
2.0s	0.81/1.32	0.71/0.94	0.83/1.61	0.75/1.24	0.59/0.82	0.71/1.14	0.68/1.23	0.49/0.70	0.51/0.72	0.37/0.56	**0.34/0.50**
3.0s	1.19/1.94	1.09/1.77	1.27/2.44	1.03/1.51	0.85/1.24	1.00/1.57	1.01/1.76	0.73/1.02	0.71/0.98	0.58/0.85	**0.53/0.75**
Total(4.0s)	1.59/2.41	1.53/2.26	1.69/2.95	1.13/2.01	1.15/1.57	1.25/1.47	1.31/1.79	0.96/1.30	0.96/1.27	0.82/1.15	**0.75/0.97**
Speed (11 per.)	NA	NA	NA	NA	NA	NA	NA	$\sim 21.4ms$	$> 900ms$	$\sim 65.7ms$	$\sim 19.3ms$

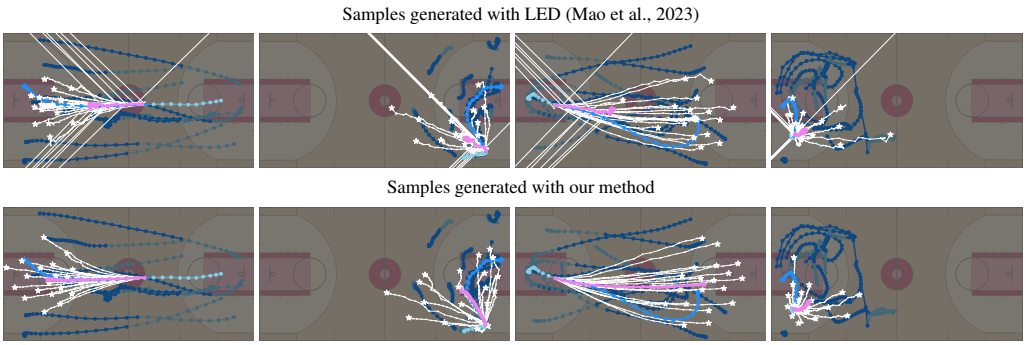


Figure 4: Generated prediction samples with $N = 20$ on NBA dataset. The light blue and dark blue represent the history and future ground truth respectively, and white curves are samples. Mean estimation is highlighted by violet.

Table 4: Comparisons for ADE and FDE on JRDB dataset with deterministic trajectory prediction, in **meter** unit. In input modality, 'T' represents trajectory only, and '2d BB' means the bounding box of the person in 2D images. ⋆ highlights previously reported best performance.

Models	Input Modality	ADE (meters)	FDE (meters)
Social-LSTM (Alahi et al., 2016)	T	0.47	0.95
Social-GAN (Gupta et al., 2018)	T	0.50	0.99
Directional-LSTM (Kothari et al., 2021)	T	0.45	0.87
Trajectron++ (Salzmann et al., 2020)	T	0.40	0.78
Autobots (Girgis et al., 2021)	T	0.39	0.80
EqMotion (Xu et al., 2023)	T	0.42	0.78
Social-Transmotion⋆ (Saadatnejad et al., 2023)	T	0.40	0.77
Social-Transmotion⋆ (Saadatnejad et al., 2023)	T + 2d BB	0.37	0.73
LED (Mao et al., 2023) (N=1)	T	0.32	0.54
Ours (N=1, Stage 1 Only)	T	**0.26**	**0.48**

eral sampled positions deviating significantly from the real potential distribution of human motions. This noise can negatively impact downstream tasks requiring estimated distributions with confidence since it is impractical to measure the ground truth future and report the minimum error in such cases.

JRDB. We finally explore the performance of our method on JRDB, which contains diverse captured indoor and outdoor environments close to humans by a robot. To ensure a fair comparison with other state-of-the-art methods, many of which provide deterministic results, we degenerate our method to output only one sample at a time. Throughout training, validation, and testing, we set N to 1 for the sampler, and the resulting ADE/FDE metrics are detailed in Table 4. Leveraging trajectory-only inputs, our approach outperforms the previous best model Social-Transmotion (Saadatnejad et al., 2023) that combines 2D bounding boxes extracted from images by approximately 29% and 34% in terms of ADE and FDE, respectively. We also conduct experiments under stochastic conditions to further investigate the effectiveness of our proposed approach with results presented in Table 5. Our model significantly outperforms the second-best model LED in deterministic cases in terms of minADE_{20} and minFDE_{20} metrics. Moreover, the inference time of our method is over 17 times faster in a bustling environment involving 80 persons, which indicates it can applied for the standard video setting, and enables real-time capabilities on physical robotic systems or desktop GPUs. Figure 5 illustrates eight examples of our prediction results, demonstrating the ability of our learned distribution to capture human intention changes like sharp turns, sudden stops, *etc*.

Table 5: Stochastic trajectory prediction comparisons on JRDB dataset. minADE_{20} and minFDE_{20} in **meters** are reported.

Method	1.2s	2.4s	3.6s	Total(4.8s)	Speed (80 per.)
LED (Mao et al., 2023)	0.05/0.07	0.09/0.14	0.14/0.21	0.18/0.28	$\sim 118ms$
Ours (Stage 1 Only)	0.05/0.06	0.09/0.13	0.13/0.20	0.17/0.27	$\sim 6.7ms$
Ours	**0.04/0.05**	**0.08/0.11**	**0.11/0.17**	**0.15/0.23**	$\sim 6.8ms$

Table 6: Robustness test for imperfect past trajectories on JRDB dataset. History trajectory includes 9 frames with a scale equal to one, and the percentage stands for the ratio of test data with noise.

Noise Type	ADE	FDE
Gaussian Noise (scale = 0.1, 4 frames)	0.27	0.49
Gaussian Noise (scale = 0.2, 4 frames)	0.32	0.55
Gaussian Noise (scale = 0.5, 4 frames)	0.71	1.18
Dropped History (20%, 5 frames)	0.26	0.48
Dropped History (50%, 5 frames)	0.26	0.49
Dropped History (50%, 7 frames)	0.29	0.52
Dropped History (80%, 7 frames)	0.30	0.54
Ours (full performance)	**0.26**	**0.48**

Table 7: Ablations on sampler types (Left) and different strides (Right). minADE_{20} and minFDE_{20} for the entire trajectory are reported. Influences from stride τ are shown with ZARA1 subset, and they differ across datasets. 'AVG.' is for the average of distance errors on 20 samples and 'SD.' represents the standard deviation. Speed is measured for an 11-person scenario.

Dataset	Stage 1 Only	Stage 2 Only	Two Stages
ETH	0.28/0.46	0.24/0.40	0.26/0.37
HOTEL	0.13/0.21	0.11/0.18	0.11/0.17
UNIV	0.32/0.58	0.28/0.48	0.28/0.49
ZARA1	0.20/0.36	0.20/0.33	0.17/0.30
ZARA2	0.16/0.30	0.15/0.26	0.14/0.25
SDD (pixels)	8.16/13.37	7.83/12.59	7.20/11.29
NBA	0.83/1.16	0.77/0.98	0.75/0.97

Stride τ	minADE/ FDE_{20}	AVG.	SD.	Speed ($\sim$)
1	0.28/0.35	0.84/1.66	0.44/0.92	15.18ms
2	0.18/0.31	0.74/1.52	0.40/0.83	8.23ms
3	**0.17/0.30**	0.73/1.50	**0.38/0.80**	5.93ms
4	0.18/0.30	**0.71/1.46**	0.39/0.81	4.73ms
6	0.18/0.30	0.73/1.50	0.39/0.81	3.56ms

4.2 Robustness against Observation Noise

In real-world scenarios, historical trajectories are often imperfect due to tracking errors and incomplete detection. It may contain truncated and broken trajectories, mismatched detection candidates, *etc*. To validate the robustness, we introduce two types of noise to simulate detection and tracking errors, as shown in Table 6. The test is conducted on the JRDB dataset as it is egocentric and naturally prone to issues when obtaining history sequences from noisy tracklets. To deal with noise type Dropped History, we left pad by repeating the first available frames to buffer the dropped observations, which is akin to a stationary person. Since there is no design bottleneck in our framework that precludes stationary states, we consistently produce high-quality predictions across different noise levels. Empirically, we note that a minimum of 2 moving history frames are needed for our method to capture the motion effectively.

4.3 Ablations

We further explore the effects of sampler types and different strides with results reported in Table 7. When replacing the sampling from a standard normal distribution with purposive sampling, the performances improve from around 7% to 20% on different datasets. Even with Stage 1 solely, we can achieve comparable prediction performances with other baselines. Experiment results also validate that the Markovian assumption of human motion usually holds up to defined observation frequency intervals with a stride τ to reach the best performance.

4.4 Group Reasoning via Potentials

Potentials learned in the trajectory prediction network can also be utilized in group inference. Leveraging group labels from the JRDB-Act (Ehsanpour et al., 2022) dataset, we perform group clustering without prior assumptions or maximum group number constraints. Concatenated with history image features extracted via CLIP (Radford et al., 2021), we build a binary edge classifier supervised by ground truth groups. The resulting group inference, portrayed in Figure 6, counts individuals as part of a group if there is an edge between them and at least one other agent within the current group. Despite some errors, our method effectively captures scenario relationships, beneficial for pedestrian-friendly navigation tasks adhering to social norms.

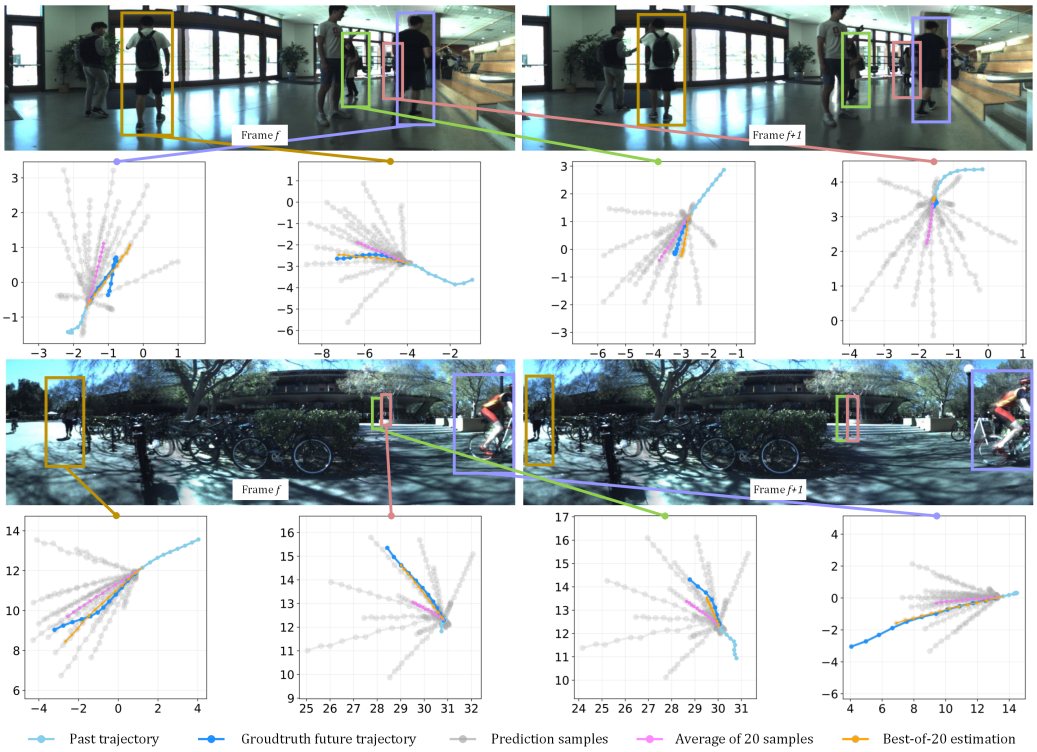


Figure 5: Generated prediction samples with $N = 20$ on JRDB dataset. Best-of-20 samples are highlighted with orange curves.

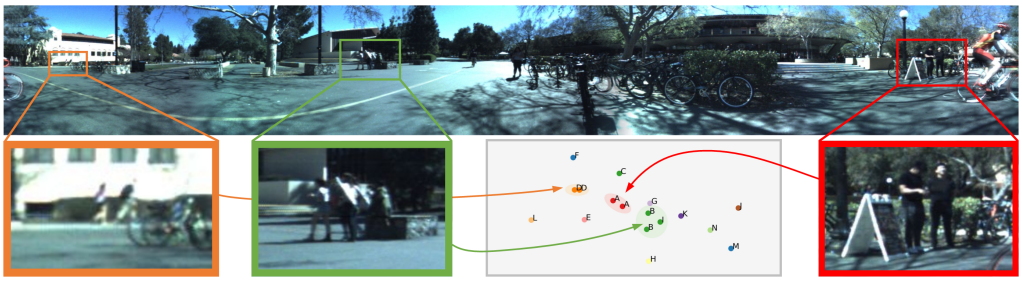


Figure 6: Group inference with learned potentials for JRDB dataset. Matching capital letters represent identical group labels, while ellipses of the same color denote estimated groups. Errors may arise due to low-quality visual cues (*cf.* Person I is misassigned to group B in the green ellipse).

5 Conclusion

In this paper, we propose a neuralized Markov random field (MRF)-based method for human trajectory prediction. Specifically, we introduce an interaction-aware MRF that models agent motion and crowd interactions over time. We then design a neural network framework consisting of two CVAEs to approximate the posterior distribution for efficient learning and inference. Our method achieves state-of-the-art performance across four benchmark datasets and also enables group reasoning. Furthermore, it is robust against noisy observations and allows for real-time stochastic inference, demonstrating its feasibility on real-world downstream systems.

Limitations. We learn the potential function applied to the MRF graph from data; if the training data only exhibits low graph complexity, the model's ability to capture complex interactions will be limited. This complexity generalization issue can be further explored in the future. Moreover, the current prediction pipeline uses only coordinates as input; incorporating environmental contexts, such as obstacles and traversable paths, could also be considered.

REFERENCES

Farhad Aghili. A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics. *IEEE Transactions on Robotics*, 28(3): 634–649, 2012.

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*, pp. 961–971, 2016.

Javad Amirian, Bingqing Zhang, Francisco Valente Castro, Juan Jose Baldelomar, Jean-Bernard Hayet, and Julien Pettré. Opentraj: Assessing prediction complexity in human trajectories datasets. In *Proceedings of the asian conference on computer vision*, 2020.

Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Learning pedestrian group representations for multi-modal trajectory prediction. In *European Conference on Computer Vision*, pp. 270–289. Springer, 2022a.

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*, pp. 6477–6487, 2022b.

Inhwan Bae, Junoh Lee, and Hae-Gon Jeon. Can language beat numerical regression? language-based multimodal trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 753–766, 2024a.

Inhwan Bae, Young-Jae Park, and Hae-Gon Jeon. Singulartrajectory: Universal trajectory predictor using diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Rashmi Bhaskara, Hrishikesh Viswanath, and Aniket Bera. Trajectory prediction for robot navigation using flow-guided markov neural operator. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15209–15216. IEEE, 2024.

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*, 2019.

Hao Cheng, Wentong Liao, Xuejiao Tang, Michael Ying Yang, Monika Sester, and Bodo Rosenhahn. Exploring dynamic context for multi-path trajectory prediction. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12795–12801. IEEE, 2021.

Patrick Pakyan Choi and Martial Hebert. Learning and predicting moving object trajectory: a piecewise trajectory segment approach. *Robotics Institute*, 337:1–17, 2006.

Patrick Dendorfer, Sven Elflein, and Laura Leal-Taixé. Mg-gan: A multi-generator model preventing out-of-distribution samples in pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13158–13167, 2021.

Mahsa Ehsanpour, Fatemeh Saleh, Silvio Savarese, Ian Reid, and Hamid Rezatofighi. Jrdb-act: A large-scale dataset for spatio-temporal action, social group and activity detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20983–20992, 2022.

Harshayu Girase, Haiming Gang, Srikanth Malla, Jiachen Li, Akira Kanehara, Karttikeya Mangalam, and Chiho Choi. Loki: Long term and key intentions for trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9803–9812, 2021.

Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira Ebrahimi Kahou, Felix Heide, and Christopher Pal. Latent variable sequential set transformers for joint multi-agent motion prediction. In *International Conference on Learning Representations*, 2021.

Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *2020 25th international conference on pattern recognition (ICPR)*, pp. 10335–10342. IEEE, 2021.

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*, pp. 17113–17122, 2022.

Tongfan Guan, Chen Wang, and Yun-Hui Liu. Neural markov random field for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5459–5469, 2024.

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*, pp. 2255–2264, 2018.

Pengqian Han, Jiamou Liu, Jialing He, Zeyu Zhang, Song Yang, Yanni Tang, and Partha Roop. St crf: Spatial-temporal conditional random field for human trajectory prediction. *arXiv preprint arXiv:2311.18198*, 2023.

Yue Hu, Siheng Chen, Ya Zhang, and Xiao Gu. Collaborative motion prediction via neural motion message passing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6319–6328, 2020.

Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9644–9653, 2023.

Nitin Kamra, Hao Zhu, Dweep Kumarbhai Trivedi, Ming Zhang, and Yan Liu. Multi-agent trajectory prediction with fuzzy query attention. *Advances in Neural Information Processing Systems*, 33: 22530–22541, 2020.

Zia Khan, Tucker Balch, and Frank Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part IV 8*, pp. 279–290. Springer, 2004.

Zia Khan, Tucker Balch, and Frank Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE transactions on pattern analysis and machine intelligence*, 27(11):1805–1819, 2005.

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.

Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):7386–7400, 2021.

Beom-Jin Lee, Jinyoung Choi, Christina Baek, and Byoung-Tak Zhang. Robust human following by deep bayesian trajectory prediction for home service robots. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 7189–7195. IEEE, 2018.

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*, pp. 2221–2230, 2022.

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*, pp. 336–345, 2017.

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

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.

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*, pp. 759–776. Springer, 2020.

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*, pp. 15233–15242, 2021.

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*, pp. 5517–5526, 2023.

Roberto Martin-Martin, Mihir Patel, Hamid Rezatofighi, Abhijeet Shenoi, JunYoung Gwak, Eric Frankel, Amir Sadeghian, and Silvio Savarese. Jrdb: A dataset and benchmark of egocentric robot visual perception of humans in built environments. *IEEE transactions on pattern analysis and machine intelligence*, 2021.

Gellert Mattyus, Shenlong Wang, Sanja Fidler, and Raquel Urtasun. Enhancing road maps by parsing aerial images around the world. In *Proceedings of the IEEE international conference on computer vision*, pp. 1689–1697, 2015.

Abduallah 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*, pp. 14424–14432, 2020.

Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th international conference on computer vision*, pp. 261–268. IEEE, 2009.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pp. 8857–8868. PMLR, 2021.

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*, pp. 549–565. Springer, 2016.

Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrila, and Kai O Arras. Human motion trajectory prediction: A survey. *The International Journal of Robotics Research*, 39(8):895–935, 2020.

Saeed Saadatnejad, Yang Gao, Kaouther Messaoud, and Alexandre Alahi. Social-transmotion: Promptable human trajectory prediction. In *The Twelfth International Conference on Learning Representations*, 2023.

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*, pp. 1349–1358, 2019.

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*, pp. 683–700. Springer, 2020.

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*, pp. 8994–9003, 2021.

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*, pp. 9675–9684, 2023.

Li Sun, Zhi Yan, Sergi Molina Mellado, Marc Hanheide, and Tom Duckett. 3dof pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5942–5948. IEEE, 2018.

Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *2018 IEEE international Conference on Robotics and Automation (ICRA)*, pp. 4601–4607. IEEE, 2018.

Chuhua Wang, Yuchen Wang, Mingze Xu, and David J Crandall. Stepwise goal-driven networks for trajectory prediction. *IEEE Robotics and Automation Letters*, 7(2):2716–2723, 2022.

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*, pp. 6498–6507, 2022a.

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*, pp. 6488–6497, 2022b.

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*, pp. 1410–1420, 2023.

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

Yanyu Xu, Zhixin Piao, and Shenghua Gao. Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5275–5284, 2018.

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*, 6(2):1463–1470, 2021.

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*, pp. 507–523. Springer, 2020.

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*, pp. 9813–9823, 2021.

Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. In *European Conference on Computer Vision*, pp. 376–394. Springer, 2022.

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*, pp. 12085–12094, 2019.

Table 8: Network design details.

Module Name	Components
History Encoder	Conv-1d (6 → 32) + GRU (32 → 256)
Update Decoder	MLP (256+16 → 512 → 512 → 256 → 2×stride)
Configuration Encoder	MLP (2×stride → 32 → 32 → 32)
Dynamics Decoder	MLP (32+16 → 128 → 128 → 64 → 2×stride)
Potentials Update	MLP (3 → 16 → 32 → 32)
History Sampler	MLP (256 → 128 → 512 → 16×N)
State Sampler	MLP (32 → 32 → 64 → 16)
Future Encoder	MLP (2×stride → 64 → 256 → 256 → 2×16)

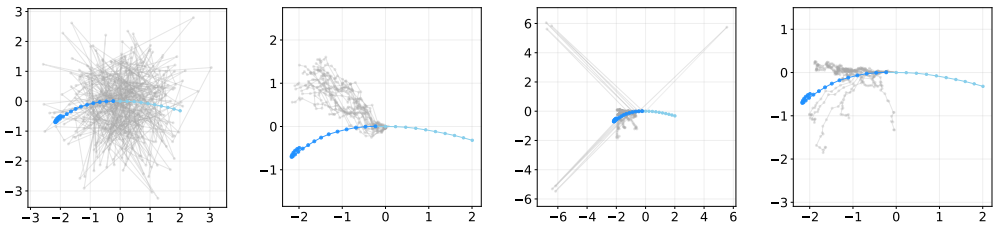
Table 9: Hyperparameters for different datasets.

Dataset	Stride	Batch Size	Learning Rate	Gamma	Step Size	Epoch (VAE)	Epoch (sampler)
ETH	3	64	2×10^{-4}	16	0.5	200	60
HOTEL	3	50	2×10^{-4}	16	0.5	200	60
UNIV	3	32	2×10^{-4}	32	0.9	200	60
ZARA1	3	32	2×10^{-4}	32	0.9	200	60
ZARA2	3	32	2×10^{-4}	32	0.9	200	60
SDD	3	16	8×10^{-4}	32	0.9	500	200
NBA	10	32	1×10^{-4}	32	0.6	150	50
JRDB	3	16	1×10^{-4}	32	0.6	200	50

A APPENDIX

We attach network design details for each sub-component in CVAEs, and data-specific hyperparameters for training and inference in Table 8 and Table 9, respectively. In the training phase, we set hyperparameters α, β and λ in loss terms as 1, 1×10^{-3} and 1×10^{-4}.

In Figure 7, we also compare the initialization and final prediction results for our proposed model and the previous state-of-the-art model LED (Mao et al., 2023) on the NBA dataset. LED starts from a noisy initialization, and it can ignore other candidates when optimizing via the best sample, which results in a distribution with multiple outliers as shown in Figure 7 (c). Conversely, our method produces trajectory-like samples initially due to the modeling of crowd motion and interaction, and converges to a more plausible distribution around the ground truth after optimization.



(a) LED initialization (b) Ours initialization (c) LED prediction (d) Ours prediction

Figure 7: Initialization and prediction results of LED and our method. Trajectories are normalized so that the coordinates of the last frames are located at the origin.