JUNE 18-22, 2023

CVPR

VANCOUVER, CANADA

Leapfrog Diffusion Model for Stochastic Trajectory Prediction

Weibo Mao[1], Chenxin Xu[1] , Qi Zhu[1], Siheng Chen[1,2], Yanfeng Wang[2,1]

[1]Shanghai Jiao Tong University, [2]Shanghai AI Laboratory

Poster: TUE-PM-132

Paper: https://arxiv.org/abs/2303.10895

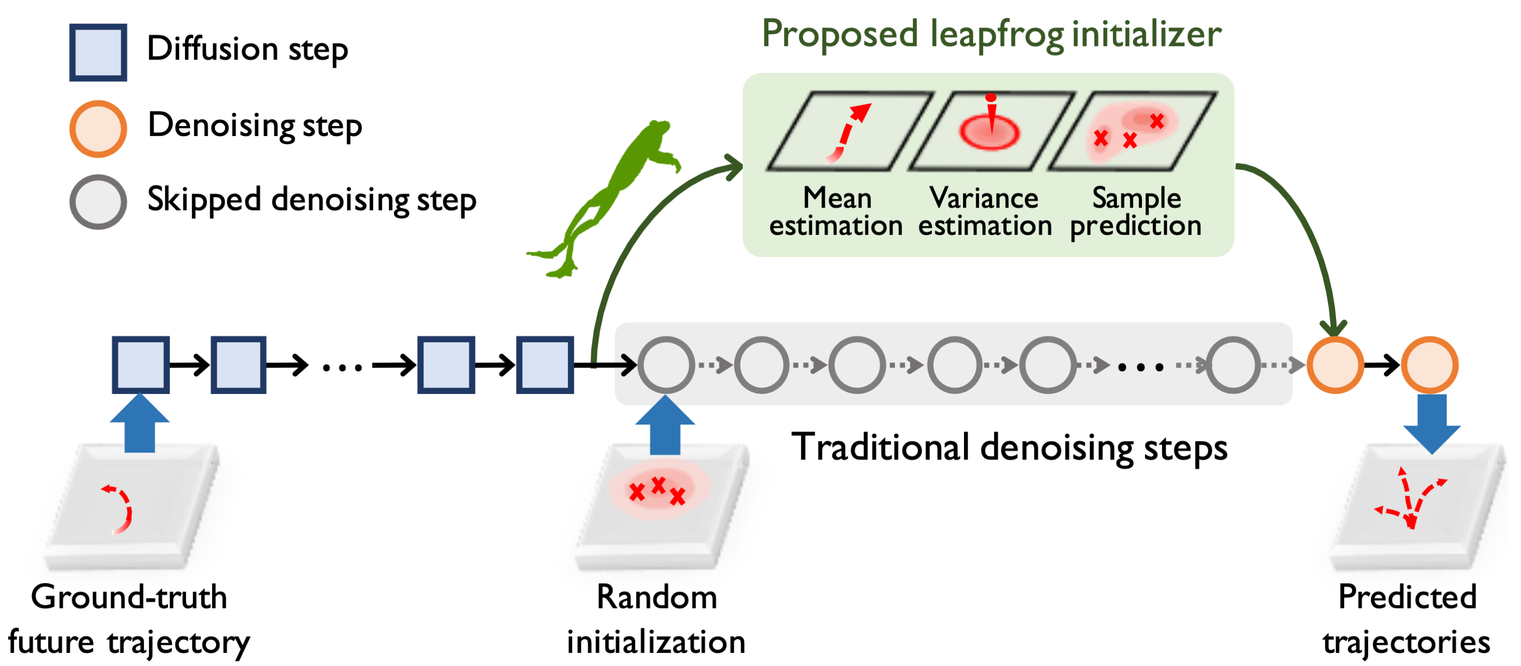
上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

上海人工智能实验室 Shanghai Artificial Intelligence Laboratory

Overview

This work focuses on **accelerating** diffusion models for stochastic trajectory prediction.

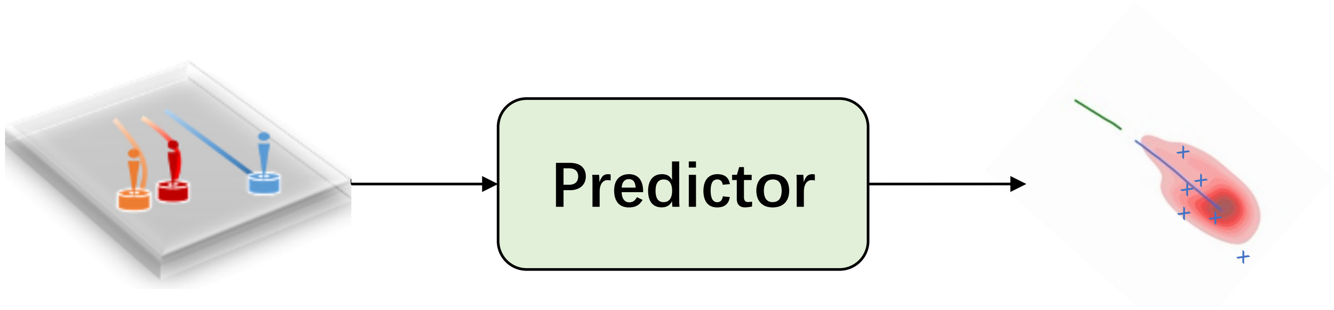
- We propose a novel LEapfrog Diffusion model (LED), which is a denoising-diffusion-based model.
- We design a trainable leapfrog initializer to directly model complex denoised distributions, accelerating inference speed.
- Our method achieves SOTA performance on four datasets while speeds up the inference by around 20 times compared to the standard diffusion model, satisfying real-time prediction needs.



Introduction

- Stochastic Trajectory Prediction

Given the past trajectories, predict the possible future trajectories.



$\mathbb{X}_{\mathcal{N}} = [\mathbf{X}_{\mathcal{N}_1}, \mathbf{X}_{\mathcal{N}_2}, \cdots, \mathbf{X}_{\mathcal{N}_N}]$

Previous/Observed Trajectories

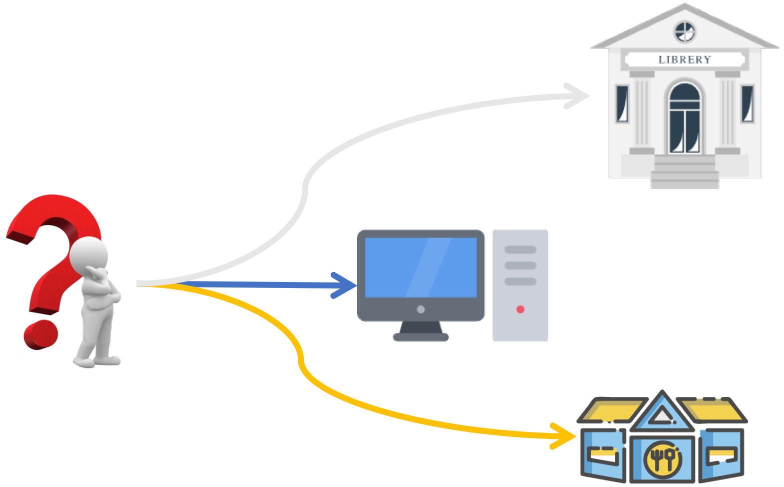
$\mathbf{Y} = [\mathbf{y}^1, \mathbf{y}^2, \cdots, \mathbf{y}^{T_\mathrm{f}}]$

Future/Predicted Trajectories

Introduction

- Stochastic Trajectory Prediction

Given the past trajectories, predict the possible future trajectories.



indeterminacy of human behaviors

multi-modal distribution

Introduction

- Deep Generative Models for Stochastic Trajectory Prediction
  - VAE, GAN, Normalizing Flow, Diffusion Model

	Quality	Diversity	Fast	Related Works
VAE		✅	✅	PECNet, Trajectron++, GroupNet
Normalizing Flow		✅	✅	CF-VAE
GAN	✅		✅	Social-GAN, NMMP
Diffusion Model	✅	✅		MID

Introduction

- Deep Generative Models for Stochastic Trajectory Prediction
  - VAE, GAN, Normalizing Flow, Diffusion Model

	Quality	Diversity	Fast	Related Works
VAE		✅	✅	PECNet, Trajectron++, GroupNet
Normalizing Flow		✅	✅	CF-VAE
GAN	✅		✅	Social-GAN, NMMP
Diffusion Model	✅	✅	✅	MID
			Ours	

Diffusion Models

- Diffusion Process

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$	Past trajectory of the ego/neighboring agent.
$\mathbf{Y}$	Future trajectory of the ego agent.

$$\boxed{\begin{aligned} \mathbf{Y}^0 &= \mathbf{Y}, && \text{(2a)} \\ \mathbf{Y}^\gamma &= f_{\text{diffuse}}(\mathbf{Y}^{\gamma-1}),\ \gamma = 1, \cdots, \Gamma, && \text{(2b)} \end{aligned}}$$

Basic idea: intentionally add a series of noises to a ground-truth future trajectory.

Diffusion Models

- Diffusion Process

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$	Past trajectory of the ego/neighboring agent.
$\mathbf{Y}$	Future trajectory of the ego agent.

$$\boxed{\mathbf{Y}^0 = \mathbf{Y}, \qquad (2a)}$$

$$\mathbf{Y}^\gamma = f_{\text{diffuse}}(\mathbf{Y}^{\gamma-1}),\ \gamma = 1, \cdots, \Gamma, \qquad (2b)$$

(2a) initializes the diffused trajectory using the GT future trajectory

Basic idea: intentionally add a series of noises to a ground-truth future trajectory.

Diffusion Models

- Diffusion Process

Note:

1) No trainable parameters yet!
2) Fixed noise schedule.

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$ Past trajectory of the ego/neighboring agent.

$\mathbf{Y}$ Future trajectory of the ego agent.

$$\mathbf{Y}^0 = \mathbf{Y}, \tag{2a}$$

$$\boxed{\mathbf{Y}^\gamma = f_{\text{diffuse}}(\mathbf{Y}^{\gamma-1}),\ \gamma = 1, \cdots, \Gamma, \qquad (2b)}$$

(2b) uses a forward diffusion operation $f_{\text{diffuse}}(\cdot)$ to successively add noises to $\mathbf{Y}^{\gamma-1}$ and obtain the diffused trajectory $\mathbf{Y}^{\gamma}$

Basic idea: intentionally add a series of noises to a ground-truth future trajectory.

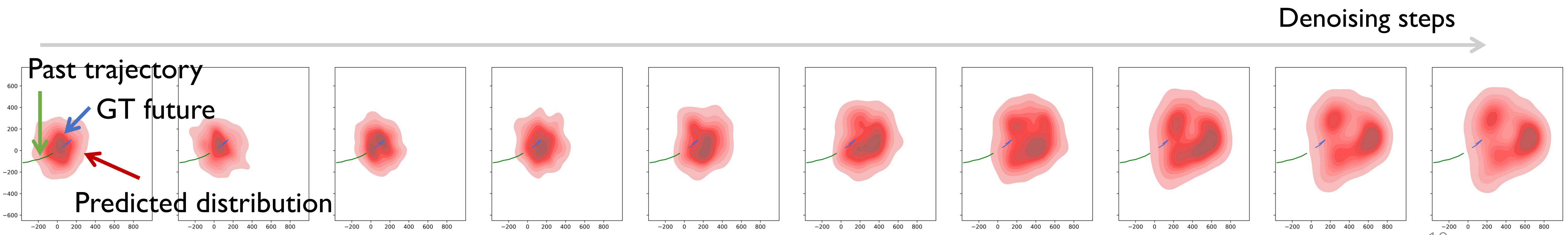
Diffusion Models

- Denoising Process

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$	Past trajectory of the ego/neighboring agent.
$\mathbf{Y}$	Future trajectory of the ego agent.

$$\boxed{\begin{aligned} &\widehat{\mathbf{Y}}_k^{\Gamma} \overset{i.i.d}{\sim} \mathcal{P}(\widehat{\mathbf{Y}}^{\Gamma}) = \mathcal{N}(\widehat{\mathbf{Y}}^{\Gamma}; \mathbf{0}, \mathbf{I}), \text{sample } K \text{ times}, && (2c)\\ &\widehat{\mathbf{Y}}_k^{\gamma} = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}}),\ \gamma = \Gamma-1, \cdots, 0, && (2d)\end{aligned}}$$

Basic idea: recover the future trajectory from noise inputs conditioned on past trajectories.



Diffusion Models

- Denoising Process

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$ Past trajectory of the ego/neighboring agent.

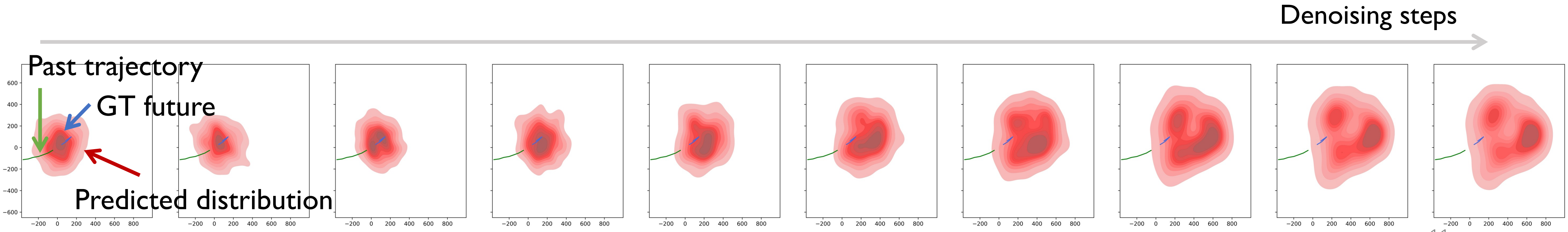
$\mathbf{Y}$ Future trajectory of the ego agent.

$$\boxed{\widehat{\mathbf{Y}}_k^{\Gamma} \overset{i.i.d}{\sim} \mathcal{P}(\widehat{\mathbf{Y}}^{\Gamma}) = \mathcal{N}(\widehat{\mathbf{Y}}^{\Gamma}; \mathbf{0}, \mathbf{I}), \text{sample } K \text{ times}, \quad (2c)}$$

$$\widehat{\mathbf{Y}}_k^{\gamma} = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}}), \ \gamma = \Gamma - 1, \cdots, 0, \quad (2d)$$

(2c) draws K independent and identically distributed samples to initialize denoised trajectories from a normal distribution.

Basic idea: recover the future trajectory from noise inputs conditioned on past trajectories.



Diffusion Models

- Denoising Process

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$ Past trajectory of the ego/neighboring agent.

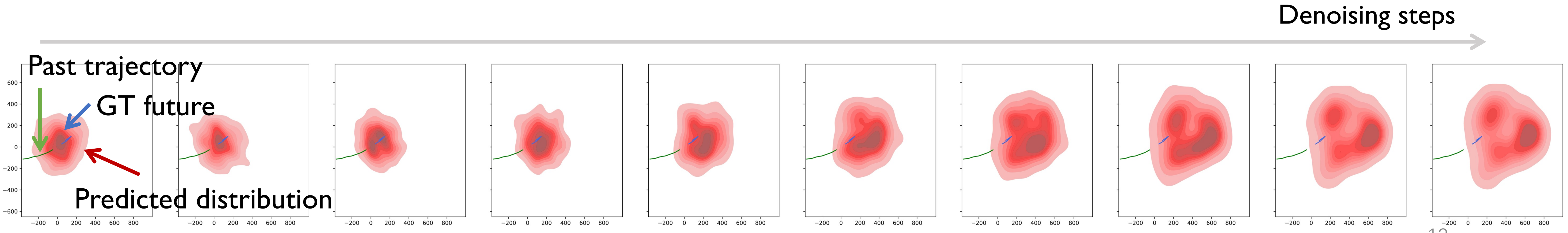
$\mathbf{Y}$ Future trajectory of the ego agent.

$$\widehat{\mathbf{Y}}_k^{\Gamma} \overset{i.i.d}{\sim} \mathcal{P}(\widehat{\mathbf{Y}}^{\Gamma}) = \mathcal{N}(\widehat{\mathbf{Y}}^{\Gamma}; \mathbf{0}, \mathbf{I}), \text{sample } K \text{ times}, \quad (2c)$$

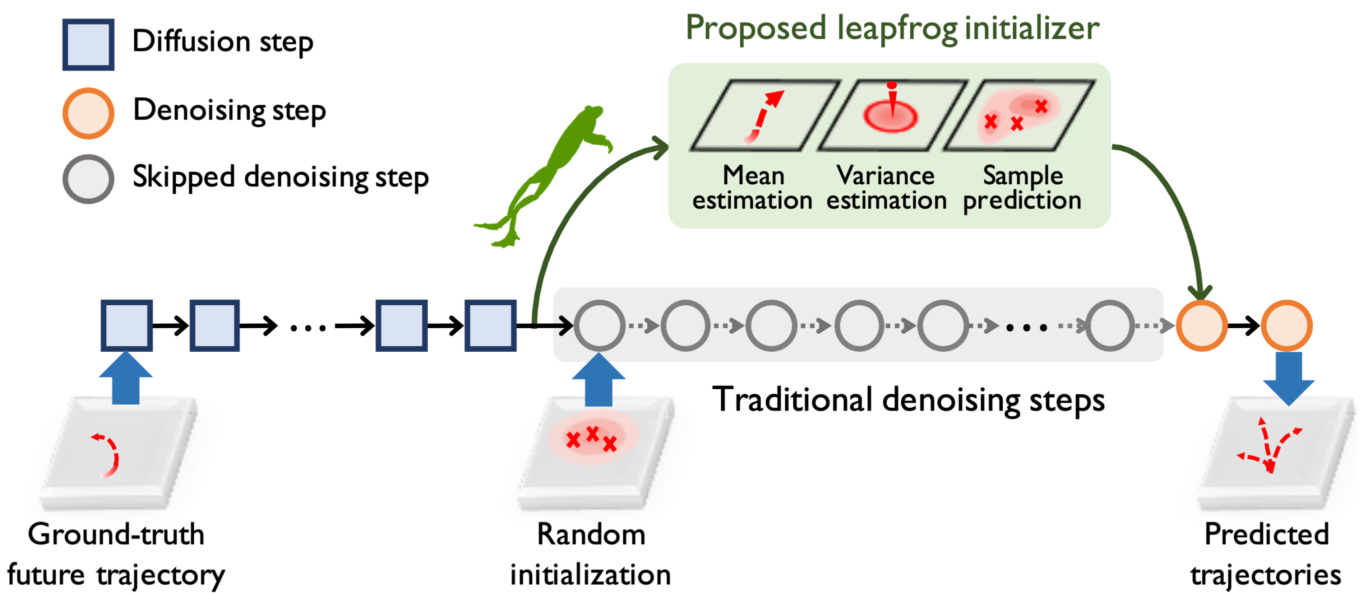
$$\boxed{\widehat{\mathbf{Y}}_k^{\gamma} = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}}), \ \gamma = \Gamma - 1, \cdots, 0, \quad (2d)}$$

(2d) iteratively applies a denoising operation $f_{\text{denoise}}(\cdot)$ to obtain the denoised trajectory conditioned on past trajectories.

Basic idea: recover the future trajectory from noise inputs conditioned on past trajectories.



Methodology – Leapfrog Diffusion Model (LED)



Motivation: LED uses the leapfrog initializer to directly estimate the denoised distribution and substitute a long sequence of traditional denoising steps.

Methodology – LED

- Mathematically,

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$	Past trajectory of the ego/neighboring agent.
$\mathbf{Y}$	Future trajectory of the ego agent.

$$\boxed{\begin{aligned} &\mathbf{Y}^0 = \mathbf{Y}, && \text{(3a)} \\ &\mathbf{Y}^\gamma = f_{\text{diffuse}}(\mathbf{Y}^{\gamma-1}),\ \gamma = 1, \cdots, \Gamma, && \text{(3b)} \end{aligned}}$$

Share the same diffusion process as diffusion models to preserve a promising representation ability;

$$\widehat{\mathcal{Y}}^\tau \overset{K}{\sim} \mathcal{P}(\widehat{\mathbf{Y}}^\tau) = f_{\text{LSG}}(\mathbf{X}, \mathbb{X}_{\mathcal{N}}), \quad \text{(3c)}$$

$$\widehat{\mathbf{Y}}_k^\gamma = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}}),\ \gamma = \tau - 1, \cdots, 0. \quad \text{(3d)}$$

Methodology – LED

- Mathematically,

$\mathbf{X}, \mathbb{X}_{\mathcal{N}}$	Past trajectory of the ego/neighboring agent.
$\mathbf{Y}$	Future trajectory of the ego agent.

$$\mathbf{Y}^0 = \mathbf{Y}, \tag{3a}$$

$$\mathbf{Y}^\gamma = f_{\text{diffuse}}(\mathbf{Y}^{\gamma-1}),\ \gamma = 1, \cdots, \Gamma, \tag{3b}$$

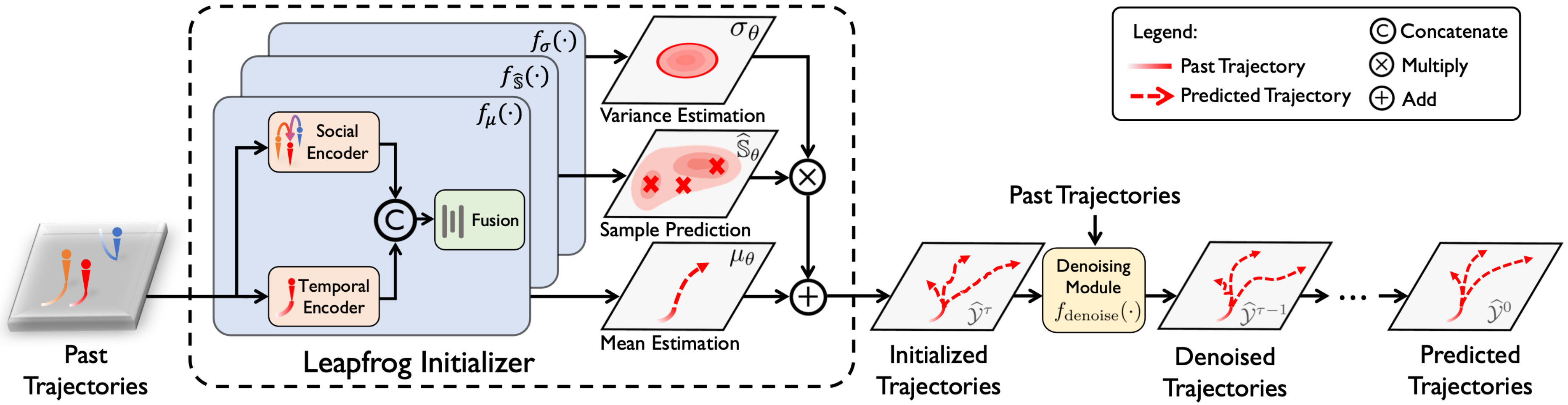
$$\boxed{\widehat{\mathcal{Y}}^\tau \overset{K}{\sim} \mathcal{P}(\widehat{\mathbf{Y}}^\tau) = f_{\text{LSG}}(\mathbf{X}, \mathbb{X}_{\mathcal{N}}), \quad (3c)}$$

$$\widehat{\mathbf{Y}}_k^\gamma = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}}),\ \gamma = \tau-1, \cdots, 0. \tag{3d}$$

(3c) proposes a novel leapfrog initializer $f_{\text{LSG}}(\cdot)$ to directly model the τ-th denoised distribution $\mathcal{P}(\widehat{\mathbf{Y}}^\tau)$;

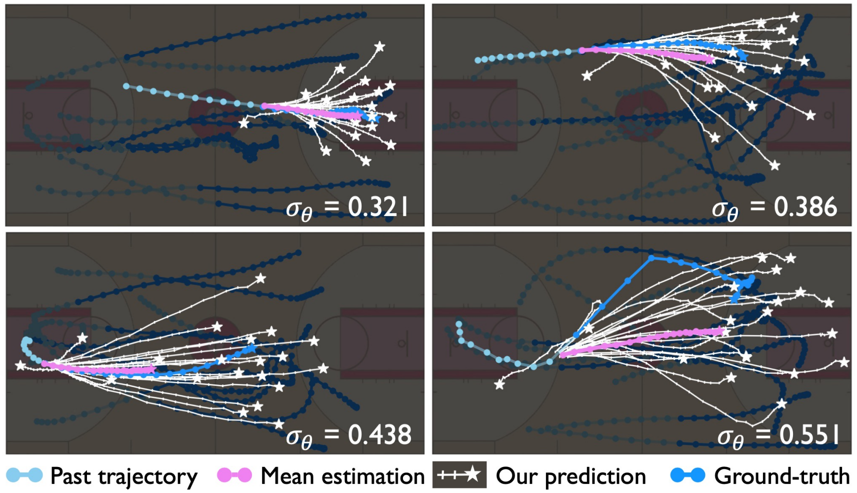
Denoising process with τ-steps!

Core Module – Leapfrog Initializer

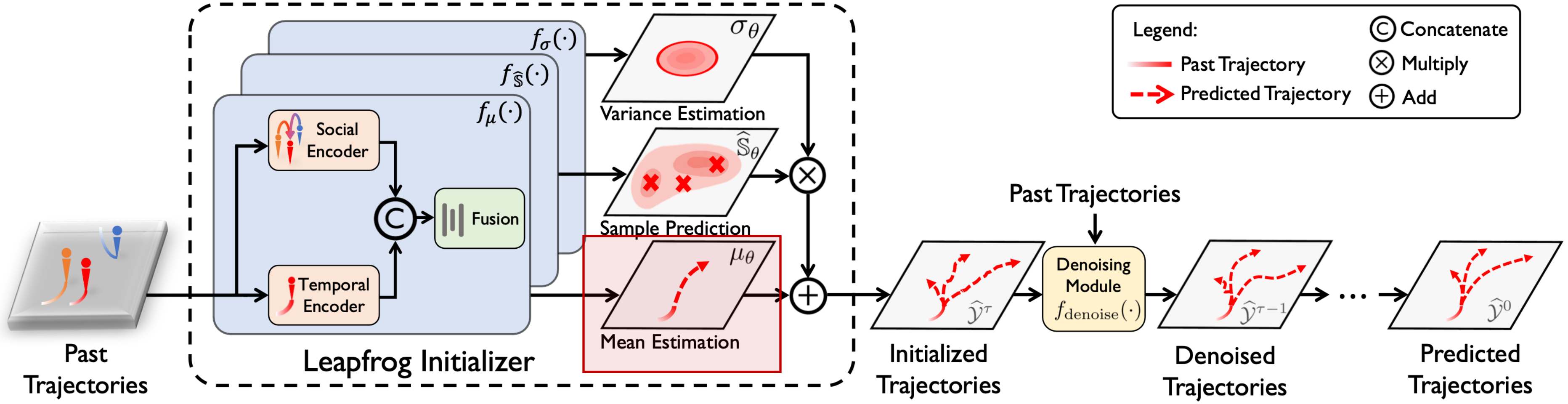


To ease the learning burden of the model, we disassemble the distribution $\mathcal{P}(\widehat{\mathbf{Y}}^\tau)$ into three representative parts: the mean, global variance and sample prediction.

$$
\begin{aligned}
\mu_\theta &= f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_f \times 2}, \\
\sigma_\theta &= f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}, \\
\widehat{\mathbb{S}}_\theta &= [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_f \times 2 \times K}, \\
\widehat{\mathbf{Y}}_k^\tau &= \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_f \times 2}, \qquad (4)
\end{aligned}
$$



Core Module – Leapfrog Initializer

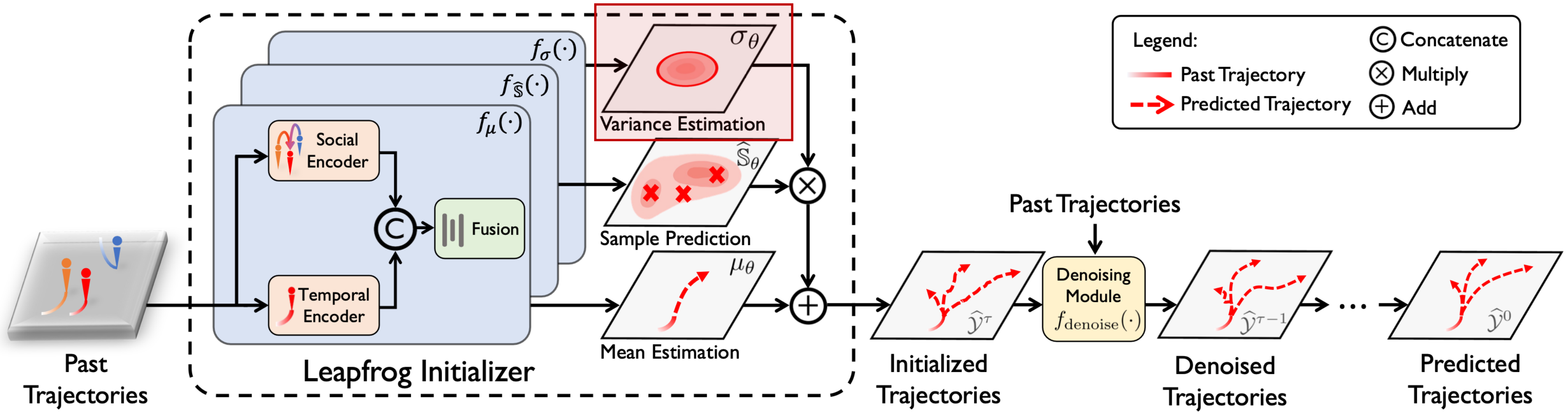


$$\boxed{\mu_\theta = f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_\mathrm{f} \times 2},}$$

Mean estimation: infer the mean trajectory as a backbone of prediction.

$$\begin{aligned}
\sigma_\theta &= f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}, \\
\widehat{\mathbb{S}}_\theta &= [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_\mathrm{f} \times 2 \times K}, \\
\widehat{\mathbf{Y}}_k^\tau &= \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_\mathrm{f} \times 2},
\end{aligned} \tag{4}$$

Core Module – Leapfrog Initializer



$$\mu_\theta = f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_\mathrm{f} \times 2},$$

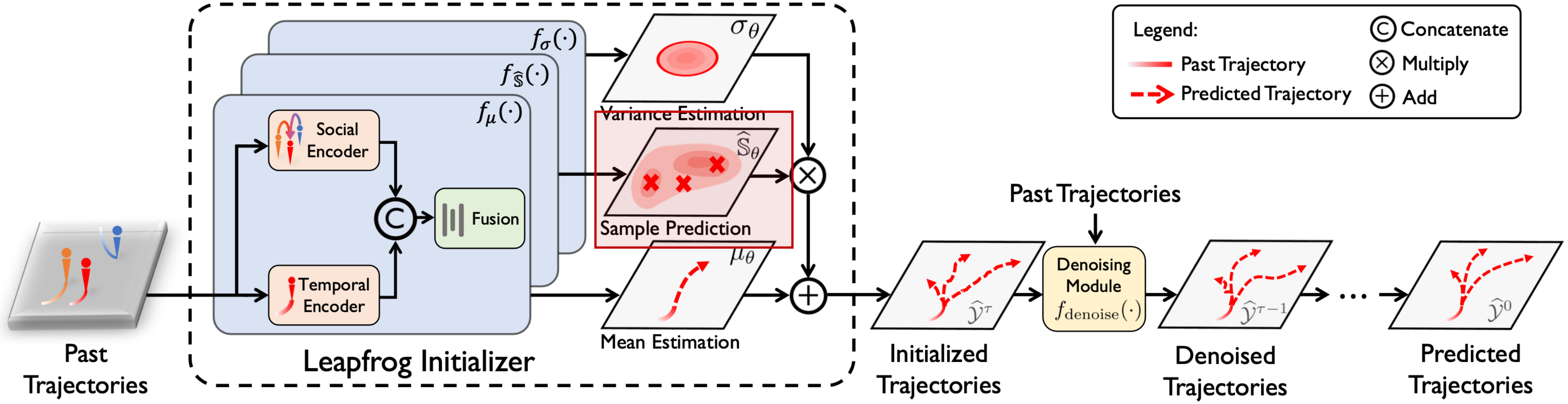
$$\boxed{\sigma_\theta = f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R},}$$

$$\widehat{\mathbb{S}}_\theta = [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_\mathrm{f} \times 2 \times K},$$

$$\widehat{\mathbf{Y}}_k^\tau = \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_\mathrm{f} \times 2}, \qquad (4)$$

Variance estimation: infer the standard deviation and control the prediction diversity

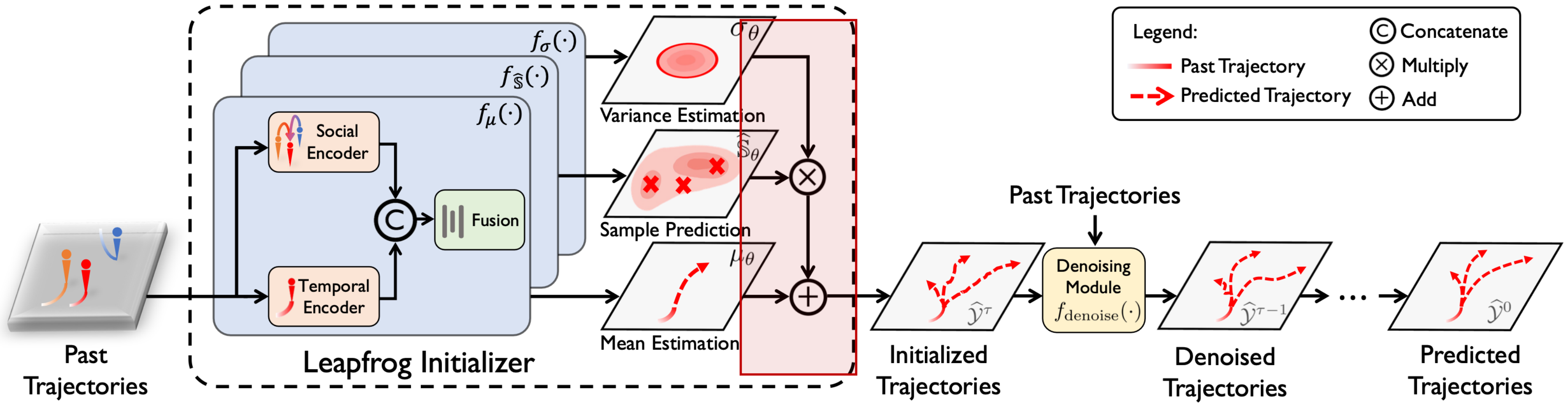
Core Module – Leapfrog Initializer



$$
\begin{aligned}
\mu_\theta &= f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_\mathrm{f} \times 2}, \\
\sigma_\theta &= f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}, \\
\widehat{\mathbb{S}}_\theta &= [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_\mathrm{f} \times 2 \times K}, \\
\widehat{\mathbf{Y}}_k^\tau &= \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_\mathrm{f} \times 2},
\end{aligned} \tag{4}
$$

Sample prediction: predict K samples simultaneously to better allocate sample position.

Core Module – Leapfrog Initializer



$$\mu_\theta = f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_\mathrm{f} \times 2},$$

$$\sigma_\theta = f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R},$$

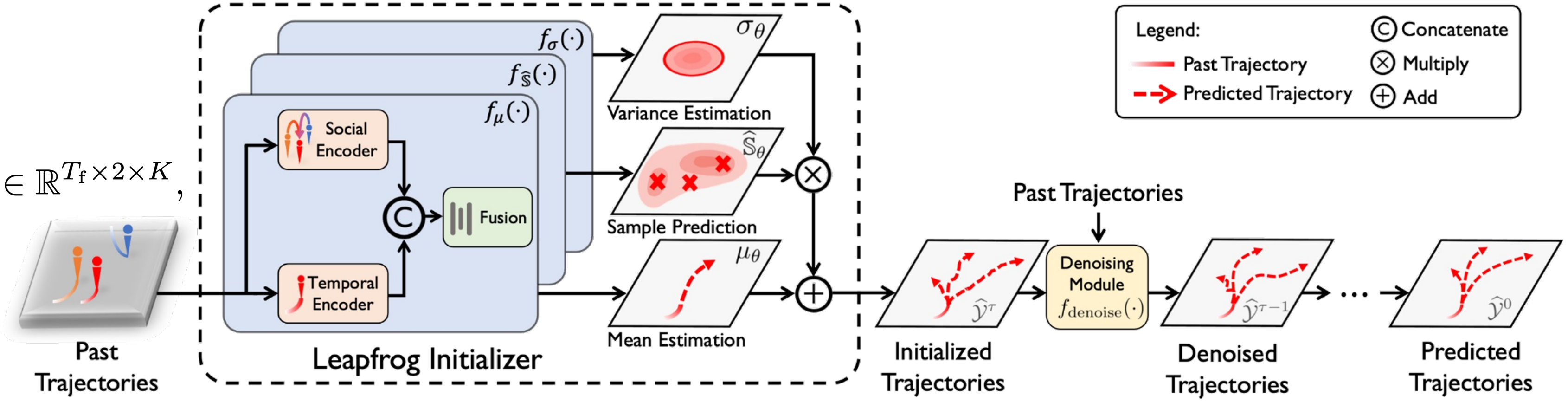
$$\widehat{\mathbb{S}}_\theta = [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_\mathrm{f} \times 2 \times K},$$

$$\boxed{\widehat{\mathbf{Y}}_k^\tau = \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_\mathrm{f} \times 2}, \qquad (4)}$$

reparameterization

Core Module – Leapfrog Initializer

$$\mu_\theta = f_\mu(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R}^{T_\mathrm{f} \times 2},$$
$$\sigma_\theta = f_\sigma(\mathbf{X}, \mathbb{X}_\mathcal{N}) \in \mathbb{R},$$
$$\widehat{\mathbb{S}}_\theta = [\widehat{\mathbf{S}}_{\theta,1}, \cdots, \widehat{\mathbf{S}}_{\theta,K}] = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_\mathcal{N}, \sigma_\theta) \in \mathbb{R}^{T_\mathrm{f} \times 2 \times K},$$
$$\widehat{\mathbf{Y}}_k^\tau = \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k} \in \mathbb{R}^{T_\mathrm{f} \times 2},$$



Loss function:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_\text{distance} + \mathcal{L}_\text{uncertainty} \\ &= \boxed{w \cdot \min_k \|\mathbf{Y} - \widehat{\mathbf{Y}}_k\|_2} + \boxed{\left(\frac{\sum_k \|\mathbf{Y} - \widehat{\mathbf{Y}}_k\|_2}{\sigma_\theta^2 K} + \log \sigma_\theta^2\right),}\end{aligned}$$

best prediction loss uncertainty loss

Inference

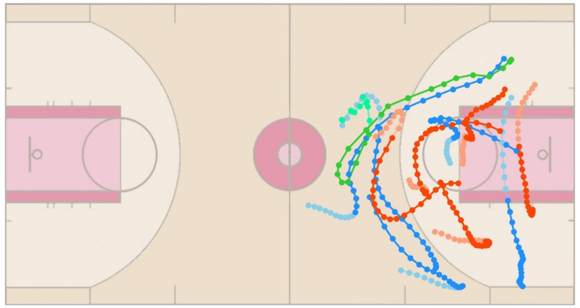
Algorithm 1 Leapfrog Diffusion Model in Inference

Input: Observed trajectories $\mathbf{X}, \mathbb{X}_{\mathcal{N}}$, Leapfrog step τ
Output: Predicted trajectories $\widehat{\mathcal{Y}}$

1: $\mu_\theta = f_\mu(\mathbf{X}, \mathbb{X}_{\mathcal{N}})$ ▷ Mean estimation
2: $\sigma_\theta = f_\sigma(\mathbf{X}, \mathbb{X}_{\mathcal{N}})$ ▷ Variance estimation
3: $\widehat{\mathbb{S}}_\theta = f_{\widehat{\mathbb{S}}}(\mathbf{X}, \mathbb{X}_{\mathcal{N}}, \sigma_\theta)$ ▷ Sample prediction
4: $\widehat{\mathbf{Y}}_k^\tau = \mu_\theta + \sigma_\theta \cdot \widehat{\mathbf{S}}_{\theta,k}, k = 1, \cdots, K$ ▷ Reparameterization
5: **for** $\gamma = \tau - 1, ..., 0$ **do**
6: $\quad \widehat{\mathbf{Y}}_k^\gamma = f_{\text{denoise}}(\widehat{\mathbf{Y}}_k^{\gamma+1}, \mathbf{X}, \mathbb{X}_{\mathcal{N}})$ ▷ Denoising step
7: **end for**
8: $\widehat{\mathcal{Y}} = \widehat{\mathcal{Y}}^0 = \{\widehat{\mathbf{Y}}_1^0, \cdots, \widehat{\mathbf{Y}}_K^0\}$
9: **return** $\widehat{\mathcal{Y}}$

Experiment

- Sport datasets



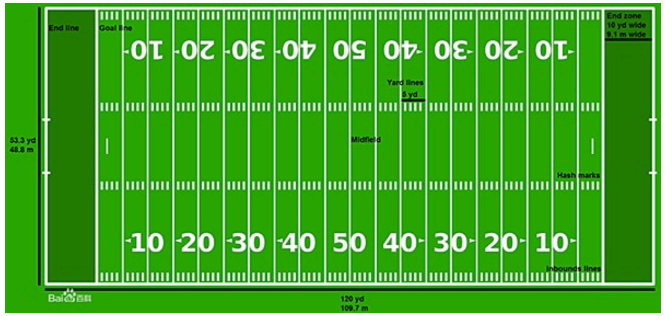


Table 1. Comparison with baseline models on NBA dataset. minADE$_{20}$ /minFDE$_{20}$ (meters) are reported. **Bold**/underlined fonts represent the best/second-best result. Compared to the previous SOTA method, MID, our method achieves a 15.6%/13.4% ADE/FDE improvement.

Time	Social-GAN [15] CVPR'18	STGAT [19] ICCV'19	Social-STGCNN [31] CVPR'20	PECNet [27] ECCV'20	STAR [52] ECCV'20	Trajectron++ [38] ECCV'20	MemoNet [50] CVPR'22	NPSN [2] CVPR'22	GroupNet [49] CVPR'22	MID [14] CVPR'22	**Ours**
1.0s	0.41/0.62	0.35/0.51	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**
2.0s	0.81/1.32	0.73/1.10	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**
3.0s	1.19/1.94	1.04/1.75	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	**0.58/0.84**
Total(4.0s)	1.59/2.41	1.40/2.18	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	**0.81/1.10**

Table 2. Comparison with baseline models on NFL dataset. minADE$_{20}$/minFDE$_{20}$ (meters) are reported. **Bold**/underlined fonts represent the best/second-best result. Compared to the previous SOTA method, MID, our method achieves a 23.7%/21.9% improvement.

Time	Social-GAN [15] CVPR'18	STGAT [19] ICCV'19	Social-STGCNN [31] CVPR'20	PECNet [27] ECCV'20	STAR [52] ECCV'20	Trajectron++ [38] ECCV'20	LB-EBM [34] CVPR'21	NPSN [2] CVPR'22	GroupNet [49] CVPR'22	MID [14] CVPR'22	**Ours**
1.0s	0.37/0.68	0.35/0.64	0.45/0.64	0.52/0.97	0.49/0.84	0.41/0.65	0.75/1.05	0.43/0.64	0.32/0.57	0.30/0.58	**0.21/0.34**
2.0s	0.83/1.53	0.82/1.60	1.06/1.87	1.19/2.47	1.02/1.84	0.93/1.65	1.26/2.28	0.83/1.52	0.73/1.39	0.71/1.31	**0.49/0.91**
Total(3.2s)	1.44/2.51	1.39/2.48	1.82/3.18	1.99/3.84	1.51/2.97	1.54/2.58	1.90/3.25	1.32/2.27	1.21/2.15	1.14/1.92	**0.87/1.50**

SOTA performance!

Experiment

- Pedestrian dataset

Table 3. Comparison with baseline models on SDD dataset. $minADE_{20}/minFDE_{20}$ (meters) are reported. **Bold**/underlined fonts represent the best/second-best result. Our method achieves the best performance in ADE/FDE. * represents the reproduced results from open source.

Time	Social-GAN [15] CVPR'18	SOPHIE [36] CVPR'19	Trajectron++ [38] ECCV'20	NMMP [18] CVPR'20	Evolve-Graph [25] NIPS'20	PECNet [27] ECCV'20	MemoNet [50] CVPR'22	NPSN [2] CVPR'22	GroupNet [49] CVPR'22	MID* [14] CVPR'22	**Ours**
4.8s	27.23/41.44	16.27/29.38	19.30/32.70	14.67/26.72	13.90/22.90	9.96/15.88	8.56/12.66	8.56/11.85	9.31/16.11	9.73/15.32	**8.48/11.66**

Table 4. Comparison with baseline models on ETH-UCY dataset. $minADE_{20}/minFDE_{20}$ (meters) are reported. **Bold**/underlined fonts represent the best/second-best result. In most subsets, our method achieves the best or second-best performance in ADE/FDE.

Subset	Social-GAN [15] CVPR'18	NMMP [18] CVPR'20	STAR [52] ECCV'20	PECNet [27] ECCV'20	Trajectron++ [38] ECCV'20	Agentformer [53] ICCV'21	MemoNet [50] CVPR'22	NPSN [2] CVPR'22	GroupNet [49] CVPR'22	MID [14] CVPR'22	**Ours**
ETH	0.87/1.62	0.61/1.08	0.36/0.65	0.54/0.87	0.61/1.02	0.45/0.75	0.40/0.61	0.40/0.76	0.46/0.73	**0.39**/0.66	**0.39/0.58**
Hotel	0.67/1.37	0.33/0.63	0.17/0.36	0.18/0.24	0.19/0.28	0.14/0.22	**0.11/0.17**	0.12/0.18	0.15/0.25	0.13/0.22	**0.11/0.17**
Univ	0.76/1.52	0.52/1.11	0.31/0.62	0.35/0.60	0.30/0.54	0.25/0.45	0.24/0.43	**0.22/0.41**	0.26/0.49	**0.22**/0.45	0.26/0.43
Zara1	0.35/0.68	0.32/0.66	0.29/0.52	0.22/0.39	0.24/0.42	0.18/0.30	0.18/0.32	**0.17**/0.31	0.21/0.39	**0.17**/0.30	0.18/**0.26**
Zara2	0.42/0.84	0.43/0.85	0.22/0.46	0.17/0.30	0.18/0.32	0.14/0.24	0.14/0.24	**0.12**/0.24	0.17/0.33	0.13/0.27	0.13/**0.22**
AVG	0.61/1.21	0.41/0.82	0.26/0.53	0.29/0.48	0.30/0.51	0.23/0.39	**0.21**/0.35	**0.21**/0.38	0.25/0.44	**0.21**/0.38	**0.21/0.33**

Experiment

- Ablation on KEY components

Each component is beneficial!

Table 5. Ablation of leapfrog initializer in the leapfrog diffusion model on NFL with various prediction numbers K. Each module in the leapfrog initializer is beneficial.

Mean μ_θ	Variance σ_θ	Sample $\widehat{\mathbb{S}}_\theta$	K=2	K=4
✓		correlated	2.04±0.18/4.08±0.48	1.63±0.13/3.05±0.16
	✓	correlated	1.95±0.08/3.90±0.22	1.49±0.01/2.86±0.02
✓	✓	i.i.d	2.36±0.13/4.31±0.22	1.90±0.07/3.31±0.05
✓	✓	correlated	**1.84**±0.05/**3.61**±0.11	**1.47**±0.01/**2.83**±0.02
Mean μ_θ	**Variance σ_θ**	**Sample $\widehat{\mathbb{S}}_\theta$**	**K=8**	**K=20**
✓		correlated	1.25±0.02/2.31±0.04	0.99±0.03/1.68±0.04
	✓	correlated	1.23±0.01/2.20±0.01	0.95±0.01/1.54±0.02
✓	✓	i.i.d	1.51±0.04/2.67±0.07	1.18±0.02/1.90±0.03
✓	✓	correlated	**1.18**±0.01/**2.19**±0.01	**0.89**±0.01/**1.51**±0.02

Experiment

- Ablation on steps (time consumption) and performance

Table 6. Different steps Γ/τ in the standard/leapfrog diffusion model on NBA. $\tau = 5$ provides the best performance.

Method	Steps	1.0s	2.0s	3.0s	Total(4.0s)	Inference (ms)
Standard Diffusion (Γ)	10	0.45/0.51	0.98/1.55	1.62/2.56	2.21/2.77	$\sim$87
	50	0.26/0.36	0.56/0.91	0.89/1.42	1.21/1.73	$\sim$446
	100	0.21/0.28	0.44/0.64	0.69/0.95	0.94/1.21	$\sim$886
	200	0.21/0.29	0.44/0.65	0.69/0.97	0.94/1.21	$>$1s
	500	0.21/0.30	0.45/0.68	0.70/0.99	0.95/1.23	$>$1s
Leapfrog Diffusion (τ)	3	0.20/0.31	0.40/0.62	0.62/0.88	0.84/1.10	$\sim$30
	5	0.18/**0.27**	**0.37/0.56**	**0.58/0.84**	**0.81**/1.10	$\sim$46
	10	**0.17**/0.27	0.37/0.58	0.59/0.85	0.82/**1.08**	$\sim$89

Experiment

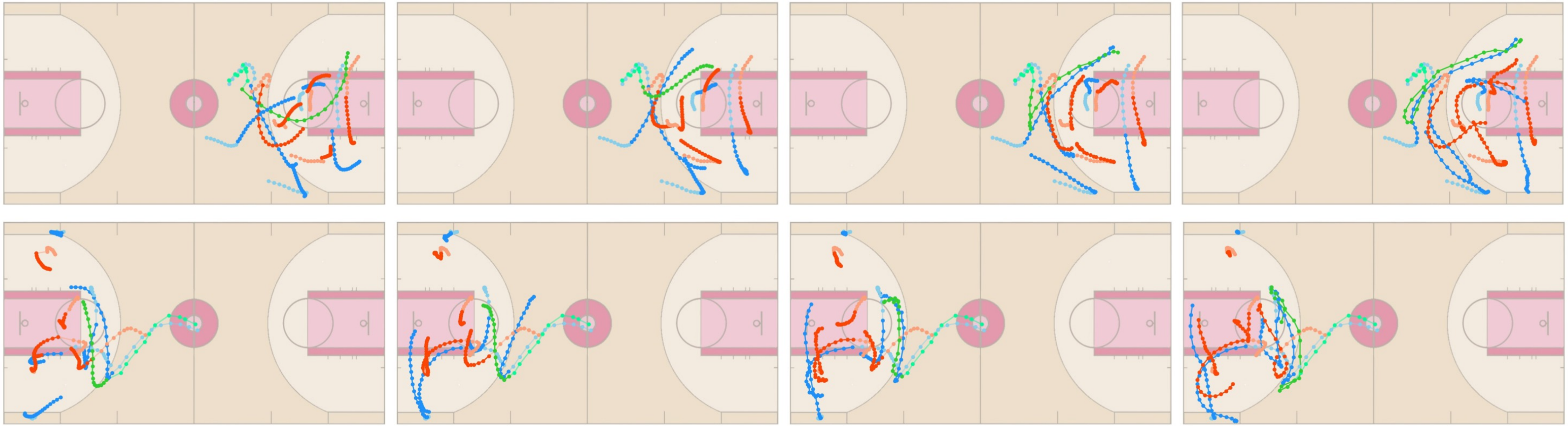
- Ablation on fast sampling methods

Table 7. Comparison to other fast sampling methods on NBA. $\eta = 1$ in DDIM. Our method achieves the best performance.

Method	1.0s	2.0s	3.0s	Total(4.0s)	Inference (ms)
PD (K=1)	0.20/0.33	0.45/0.75	0.72/1.13	0.98/1.39	$\sim$ 452
PD (K=2)	0.21/0.34	0.46/0.78	0.73/1.15	0.98/1.41	$\sim$230
PD (K=3)	0.23/0.37	0.48/0.79	0.73/1.15	0.98/1.43	$\sim$121
PD (K=4)	0.25/0.38	0.50/0.80	0.75/1.16	0.99/1.44	$\sim$64
DDIM (S=2)	0.20/0.29	0.42/0.65	0.66/0.96	0.91/1.21	$\sim$530
DDIM (S=10)	0.22/0.32	0.44/0.71	0.69/1.04	0.93/1.31	$\sim$107
DDIM (S=20)	0.24/0.35	0.49/0.81	0.76/1.21	1.02/1.51	$\sim$54
Ours	**0.18/0.27**	**0.37/0.56**	**0.58/0.84**	**0.81/1.10**	**$\sim$46**

Experiment

- Visualization comparison on NBA

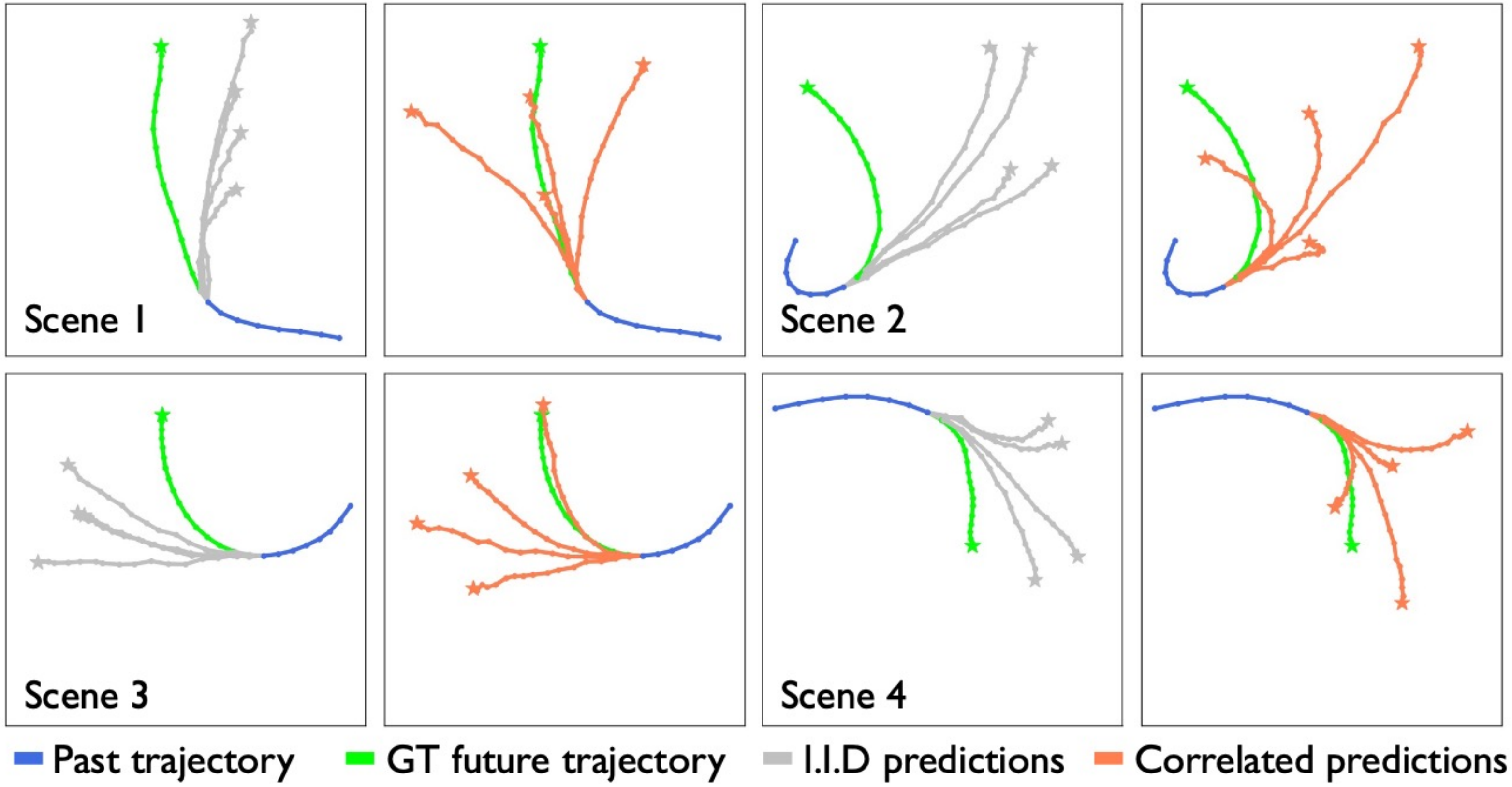


(a) PECNet (b) GroupNet (c) Ours (d) GT

Light color: past trajectory; blue/red/green color: two teams and the basketball.

Experiment

- Visualization comparison of different sampling mechanism



Thanks for your listening!