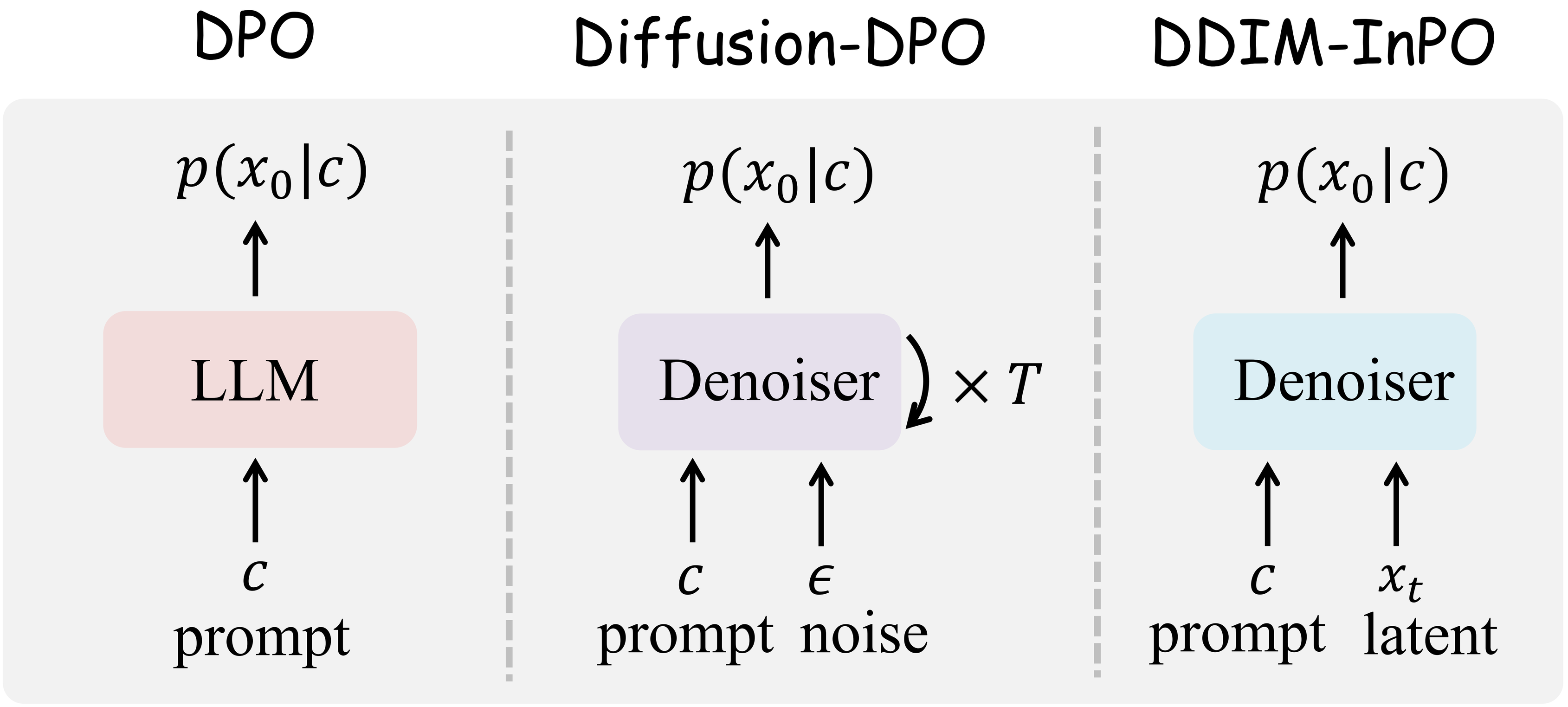
InPO: Inversion Preference Optimization with Reparametrized DDIM for Efficient Diffusion Model Alignment

Project Page:
https://jaydenlyh.github.io/InPO-project-page/

Yunhong Lu, Qichao Wang, Hengyuan Cao, Xierui Wang, Xiaoyin Xu, Min Zhang

Motivation:

- **Efficiency:** How to Mitigate the Impact of Long-Chain Markov Processes in Diffusion Models Alignment?



Solution:

- **Step-1**: Diffusion Model Reparameterization for One-Step Generation.

$$x_0(t) = \frac{x_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\epsilon_\theta^t(x_t)$$

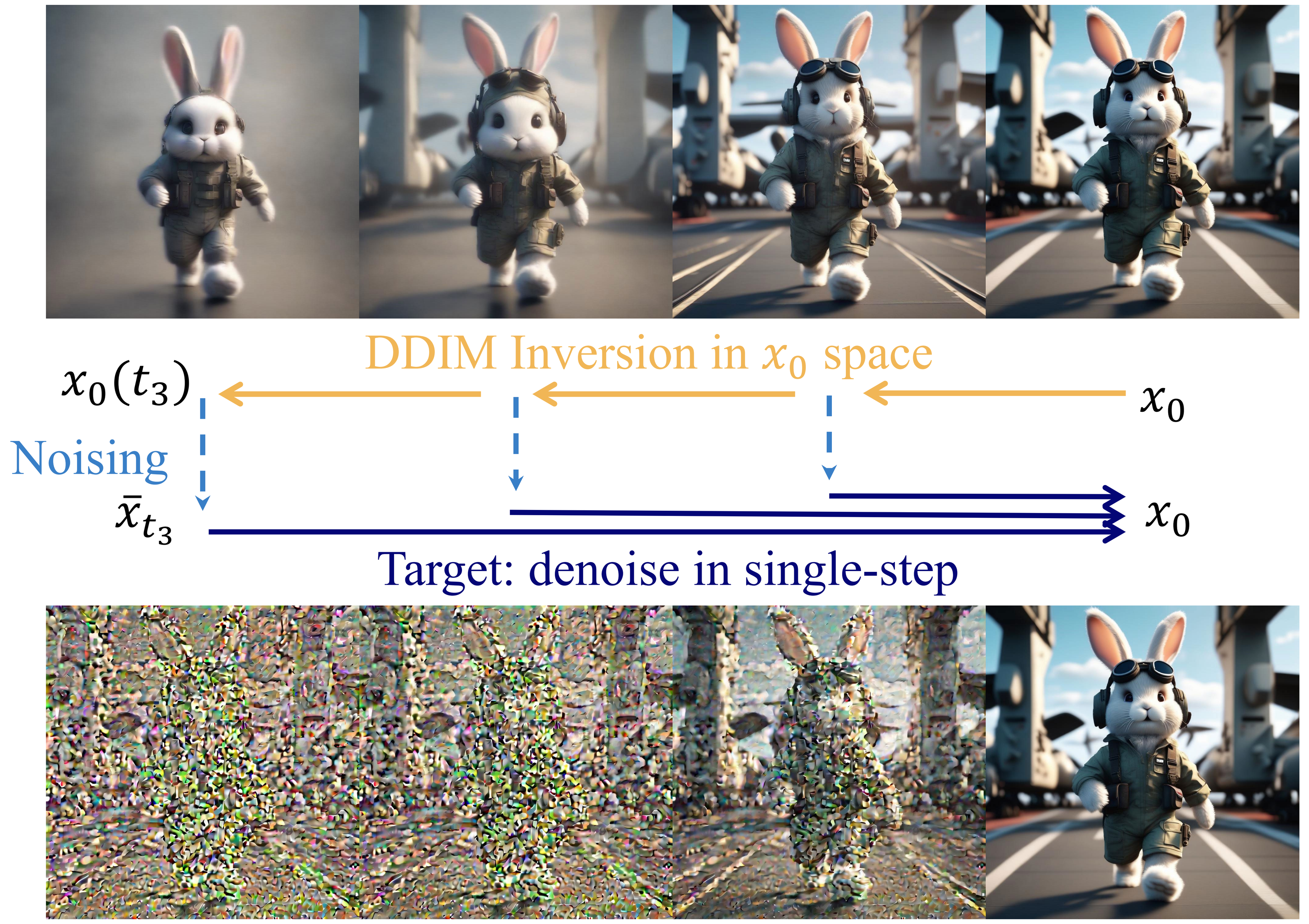
- **Step-2**: Finding Appropriate Latent Variables x_t at any timestep t via DDIM Inversion.

$$x_t = \sqrt{\frac{\alpha_t}{\alpha_{t+1}}}x_{t+1} + \left(\sqrt{\frac{1-\alpha_t}{\alpha_t}} - \sqrt{\frac{1-\alpha_{t+1}}{\alpha_{t+1}}}\right)\epsilon_\theta^{t+1}(x_{t+1})$$

- **Step-3**: Joint Distribution DPO Training Framework.

$$\mathcal{L}(\theta) := -\mathbb{E}_{t,(\boldsymbol{x}_0^w,\boldsymbol{x}_0^l,\boldsymbol{c})\sim\mathcal{D}} \log\sigma \left(\beta\mathbb{E}_{\substack{\boldsymbol{x}_t^w\sim p_\theta^{\boldsymbol{c}}(\boldsymbol{x}_t^w|\boldsymbol{x}_0^w)\\ \boldsymbol{x}_t^l\sim p_\theta^{\boldsymbol{c}}(\boldsymbol{x}_t^l|\boldsymbol{x}_0^l)}}\left[\log\frac{p_\theta^{\boldsymbol{c}}(\boldsymbol{x}_0^w,\boldsymbol{x}_t^w)}{p_{\text{ref}}^{\boldsymbol{c}}(\boldsymbol{x}_0^w,\boldsymbol{x}_t^w)} - \log\frac{p_\theta^{\boldsymbol{c}}(\boldsymbol{x}_0^l,\boldsymbol{x}_t^l)}{p_{\text{ref}}^{\boldsymbol{c}}(\boldsymbol{x}_0^l,\boldsymbol{x}_t^l)}\right]\right)$$

DDIM-InPO Framework:



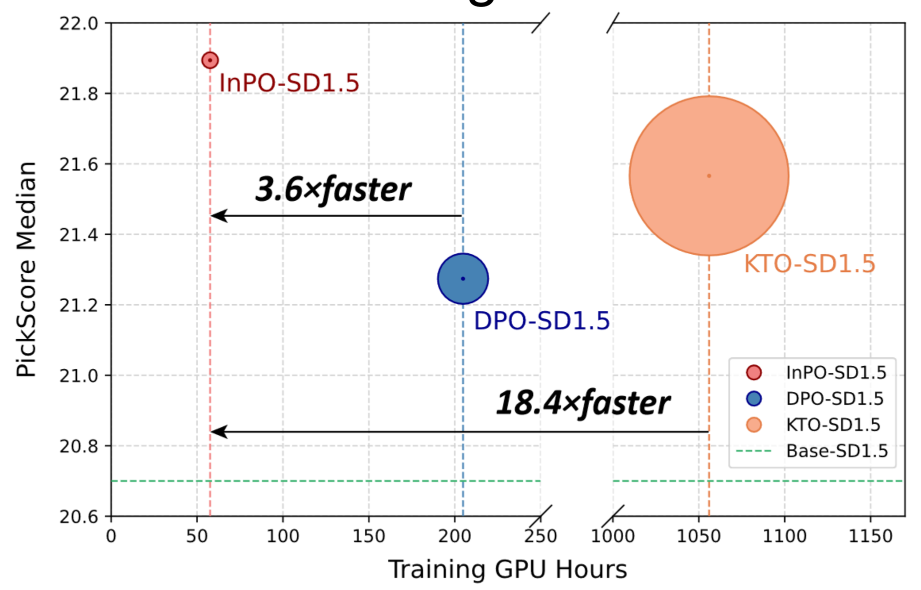
Perceptual Superiority:

Experiments:

Win-rate Comparison with Baselines

Model	Baselines	Win-rate (HPDv2) Aesthetic	PickScore	HPS	CLIP	Win-rate (Parti-Prompts) Aesthetic	PickScore	HPS	CLIP
InPO-SDXL	vs. Base-SDXL	56.37	79.25	85.38	53.37	61.70	72.89	78.31	50.55
	vs. SFT-SDXL	66.66	88.31	89.91	59.00	69.70	88.54	88.91	56.99
	vs. DPO-SDXL	55.87	59.28	65.81	46.91	57.17	56.92	60.11	42.28
InPO-SD1.5	vs. Base-SD1.5	80.19	85.84	90.22	66.44	74.63	73.16	81.86	61.89
	vs. SFT-SD1.5	56.63	66.84	58.91	57.87	52.27	62.50	58.03	57.54
	vs. DPO-SD1.5	70.50	74.94	84.06	61.09	68.01	63.54	75.31	57.54
	vs. KTO-SD1.5	59.81	67.41	60.94	60.03	59.01	62.25	58.76	57.60

Training Costs



Conclusion:

Advantages of Fine-Tuning Data-Dependent Latent Variables in Pre-Trained Diffusion Models:

- Preservation of Pre-Trained Knowledge
- Targeted Performance Improvement
- Optimized Computational Efficiency
- Interpretability and Parametric Control
- Cross-Domain Broader Applicability

Future Work: Targeted fine-tuning of specific latent variables (via **controlled noise injection**) may establish a universal paradigm for future **post-training** adaptation.