Accelerating Diffusion Sampling with Optimized Time Steps

Shuchen Xue[1] · Zhaoqiang Liu[2] · Fei Chen[3] · Shifeng Zhang[3] · Tianyang Hu[3] · Enze Xie[3] · Zhenguo Li[3]

1AMSS, CAS 2UESTC 3Huawei Noah's Ark Lab

Diffusion Models

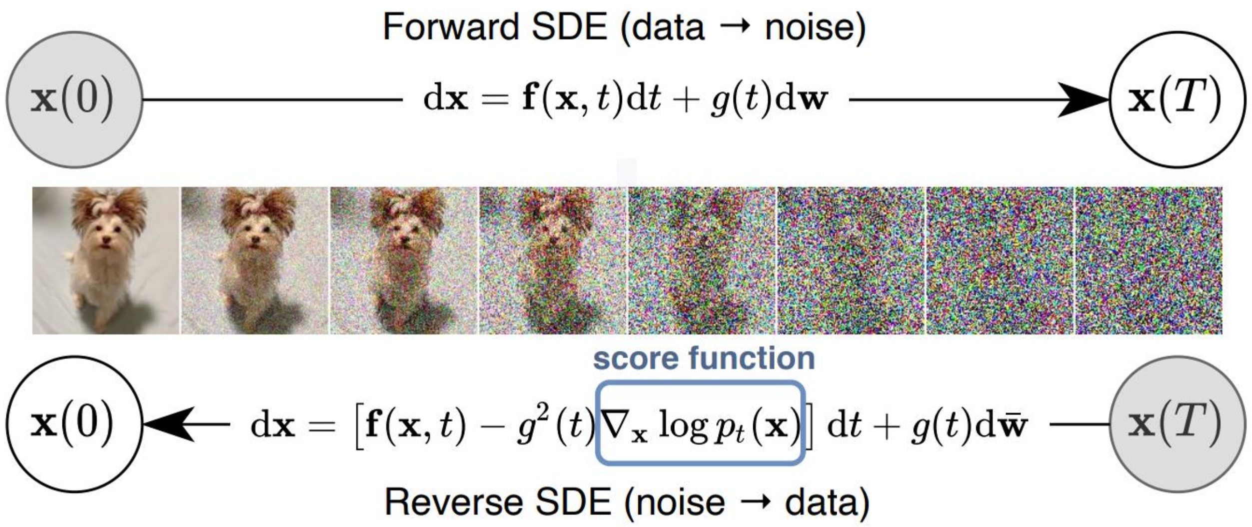


Image from Song et al., 2020

- Getting noise from data is easy (Forward SDE).
- Generating data by reversing the forward process.

Estimating the score function by Denoising Score matching (Vincent 2010).

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_t \Big\{ \lambda(t) \mathbb{E}_{\boldsymbol{x}_0} \mathbb{E}_{\boldsymbol{x}_t|\boldsymbol{x}_0} \Big[\big\| \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}, t) - \nabla_{\boldsymbol{x}_t} \log p_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0) \big\|_2^2 \Big] \Big\}$$

Accelerating Diffusion Sampling with Optimized Time Steps

Target: Given pretrained diffusion model, budget of N network inferences and a prescribed diffusion sampler, what is the optimal discretization scheme of time steps?

Method: Derive a theory-based surrogate optimization objective which can be solved utilizing existing optimization method (e.g. constrained trust region method)

Advantage: Negligible computational cost lightweight optimization can be solved within tens of seconds.

Optimization objective: minimize which is the distance between ground truth solution and the numerical solution (related to selected timesteps)

Assumption: the score matching error are bounded in L2 norm

Assumption 1. *For any $t \in \{t_0, t_1, \ldots, t_N\}$, the error in the score estimate is bounded in $L^2(q_t)$:*

$$\begin{aligned} &\|\nabla_{\mathbf{x}} \log q_t - \mathbf{s}_{\boldsymbol{\theta}}(\cdot, t)\|^2_{L^2(q_t)} \\ &= \mathbb{E}_{q_t}\left[\|\nabla_{\mathbf{x}} \log q_t(\mathbf{x}) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, t)\|^2\right] \le \eta^2 \varepsilon_t^2, \end{aligned}$$

where $\eta > 0$ is an absolute constant.

Accelerating Diffusion Sampling with Optimized Time Steps

Lemma 1. *For any* $\mathbf{x}_0 \sim q_0$ *and* $P_0 \in (0,1)$, *with probability at least* $1 - P_0$, *the following event occurs: For all* $t \in \{t_0, t_1, \ldots, t_N\}$ *and* $\mathbf{x}_t \sim q_t$, *we have*

$$\|\mathbf{x}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \mathbf{x}_0\| \leq \tilde{\eta}\tilde{\varepsilon}_t,$$

where $\tilde{\eta} := \sqrt{\frac{N+1}{P_0}}\eta$ *and* $\tilde{\varepsilon}_t := \frac{\varepsilon_t \sigma_t^2}{\alpha_t}$.

$$\tilde{\mathbf{x}}_\epsilon = \frac{\sigma_\epsilon}{\sigma_T}\mathbf{x}_T + \sigma_\epsilon \sum_{n=1}^{N} \sum_{j=0}^{k_n-1} w_{n;k_n,j}\boldsymbol{f}(\lambda_{n-k_n+j})$$

$$\|\tilde{\mathbf{x}}_\epsilon - \mathbf{x}_0\|$$

$$\min_{\lambda_{t_1},\ldots,\lambda_{t_{N-1}}} \sum_{i=0}^{N-1} \tilde{\varepsilon}_{t_i} \cdot \Big| \sum_{n-k_n+j=i} w_{n;k_n,j} \Big|$$
$$\text{s.t.} \quad \lambda_{t_{n+1}} > \lambda_{t_n}, \text{ for } n = 0, 1, \ldots, N-1,$$

Solve the problem use constrained trust region method

Algorithm 1 *Finding the time steps via* (35)

Require: Number of time steps N, initial time of sampling T, end time of sampling ϵ, any sampling algorithm that is characterized by local polynomials $\boldsymbol{\mathcal{P}}_{n;k_n-1}(\lambda)$ of the form (10), the function $\tilde{\varepsilon}_t := \frac{\sigma_t^p}{\alpha_t}$ with a fixed positive integer p for score approximation

1: Set $\lambda_{t_0} = \lambda_T$ and $\lambda_{t_N} = \lambda_\epsilon$ and calculate $\tilde{\varepsilon}_{t_i}$ for $i = 0, 1, , \ldots, N-1$

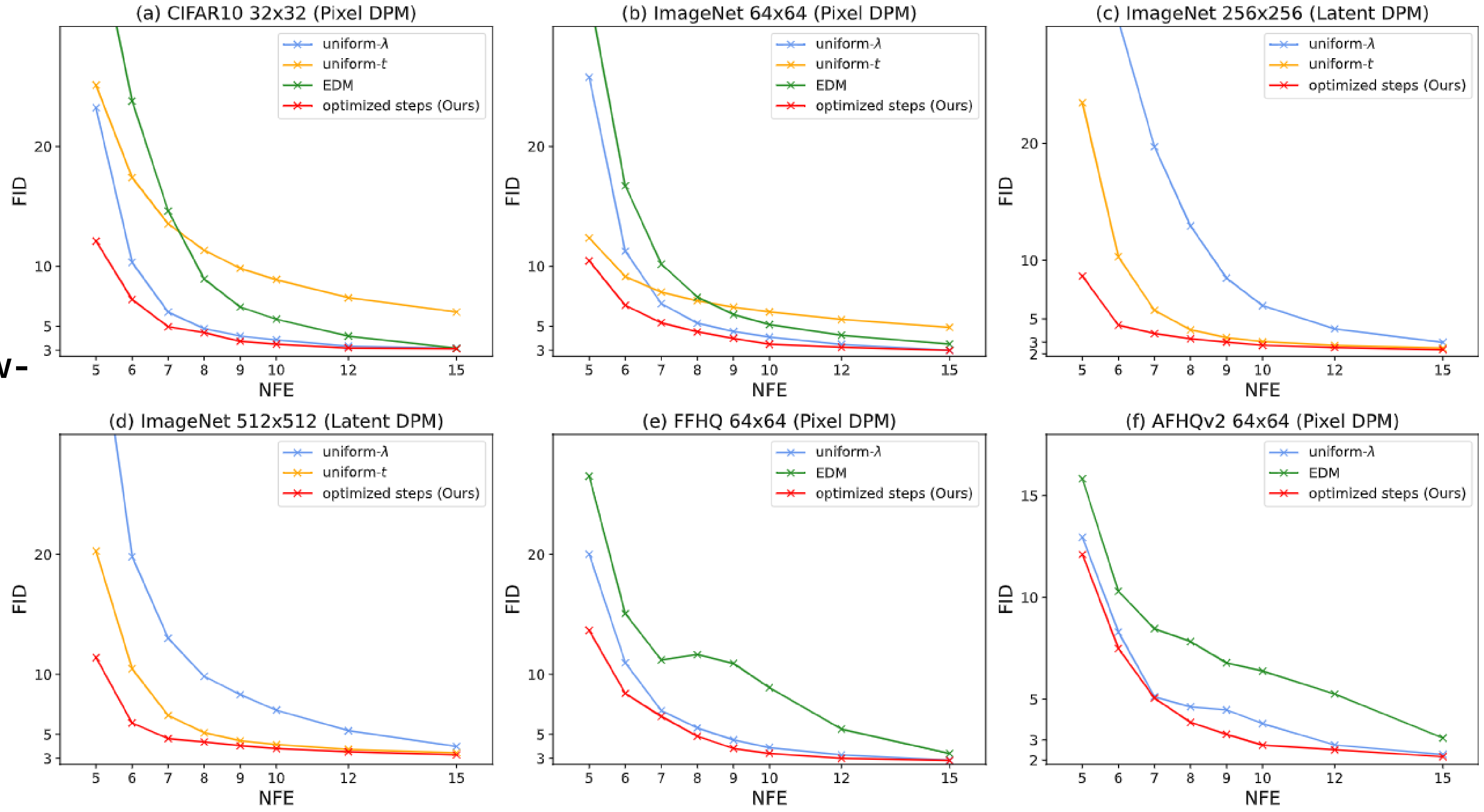
2: Calculate the weights $w_{n;k_n,j}$ from (18).

3: Solve the optimization problem in (35) via the constrained trust region method.

4: **Return**: Optimized λ (or equivalently, time) steps $\hat{\lambda}_{t_1}, \ldots, \hat{\lambda}_{t_{N-1}}$.

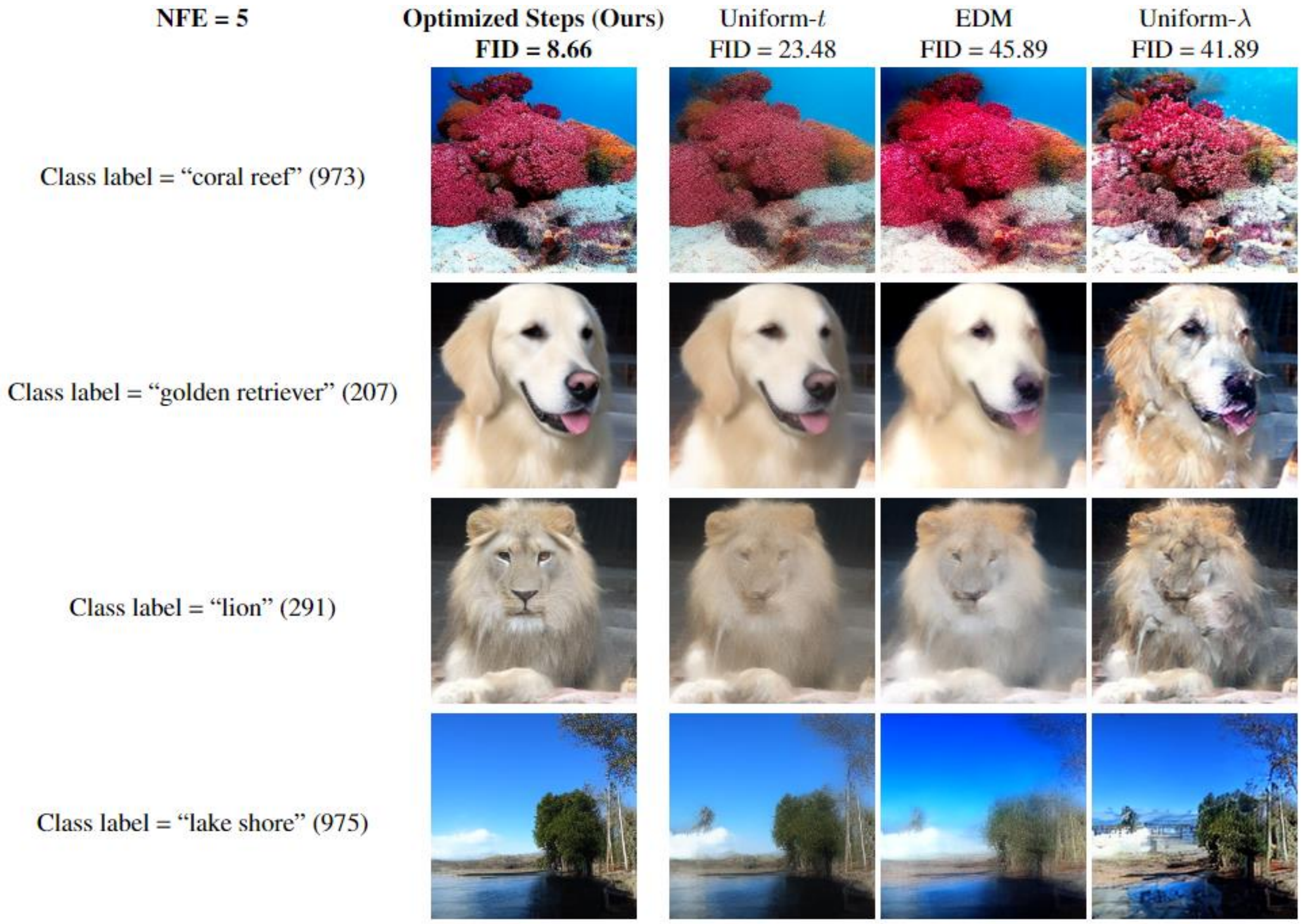
Results on benchmarks

SOTA for training-free few-step diffusion sampling

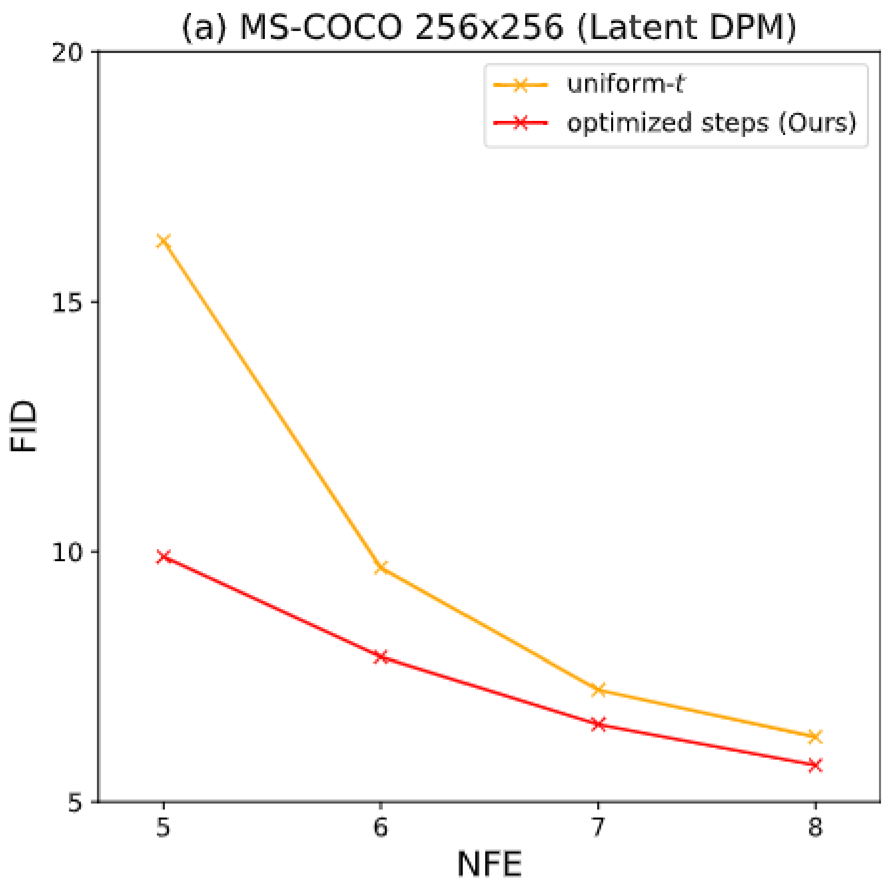


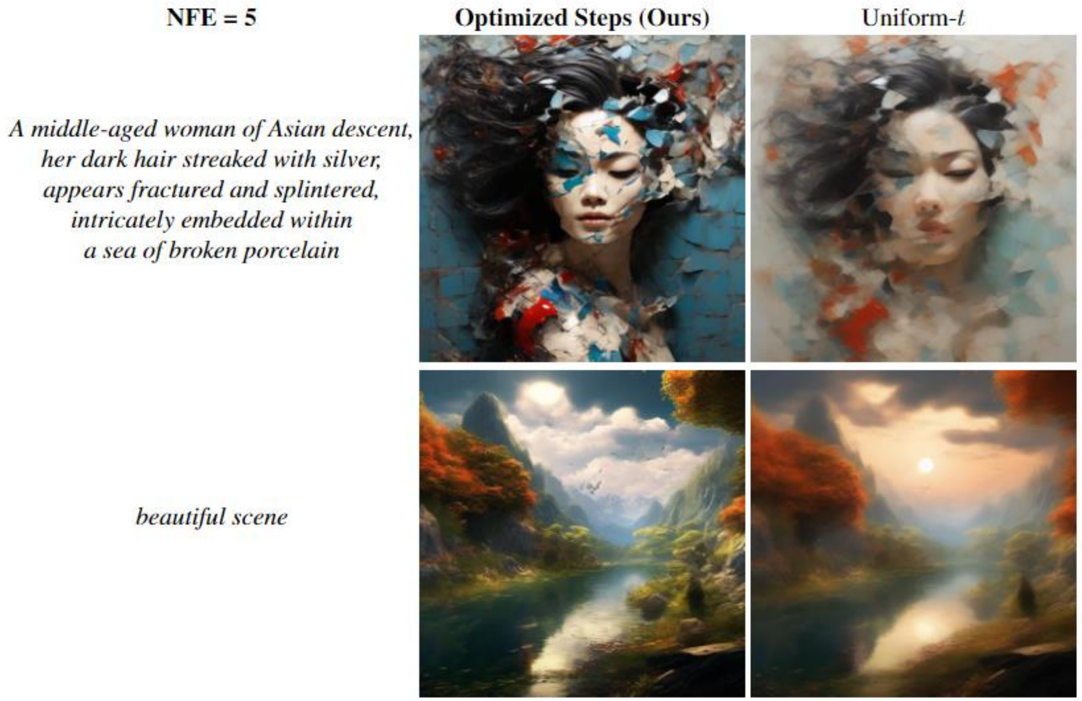
Visualization

Visualization on ImageNet 256
DIT-XL-2 model



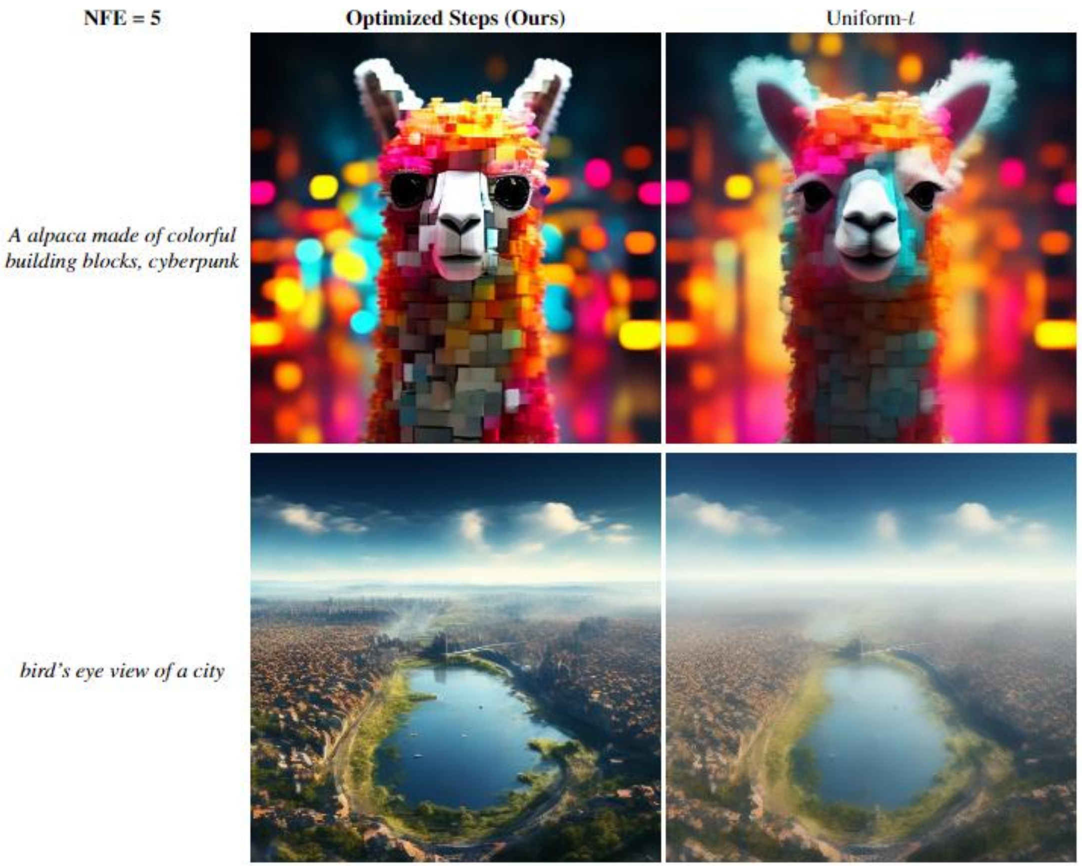
Results on t2i

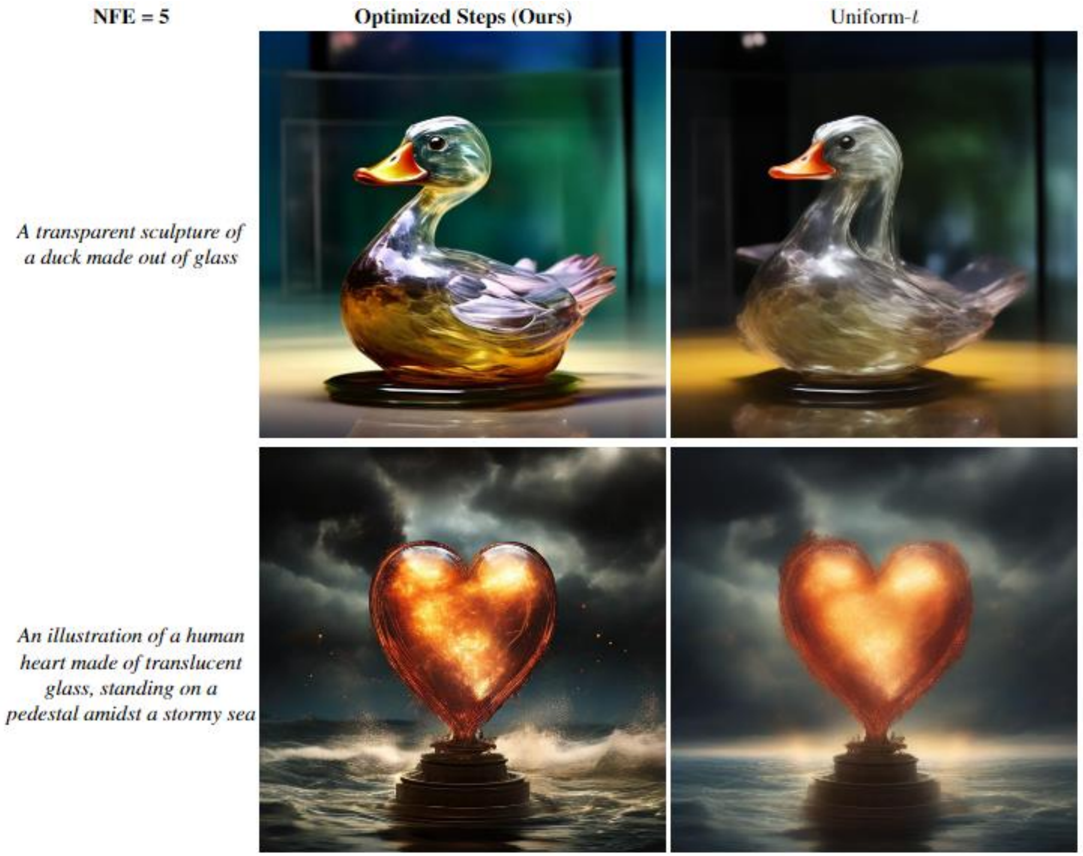




Text-to-Image generation results on PixArt-α

Results on t2i





Text-to-Image generation results on PixArt-α

Running time and Summary

seconds on CPU

NFEs	5	6	7	8	9	10	12	15
Time(s)	1.9	2.3	5.3	5.9	7.8	8.8	11.0	14.1

Table 2. Running time of our optimization algorithm.

- We propose an optimization-based method to find appropriate time steps to accelerate the sampling of diffusion models.
- Experimental results on popular image datasets demonstrate that our method can be employed in a plug-and-play manner and achieves state-of-the-art sampling performance based on various pre-trained diffusion models
- Code is available at https://github.com/scxue/DM-NonUniform