Latent Consistency Models: synthesizing high-resolution images with few-step inference

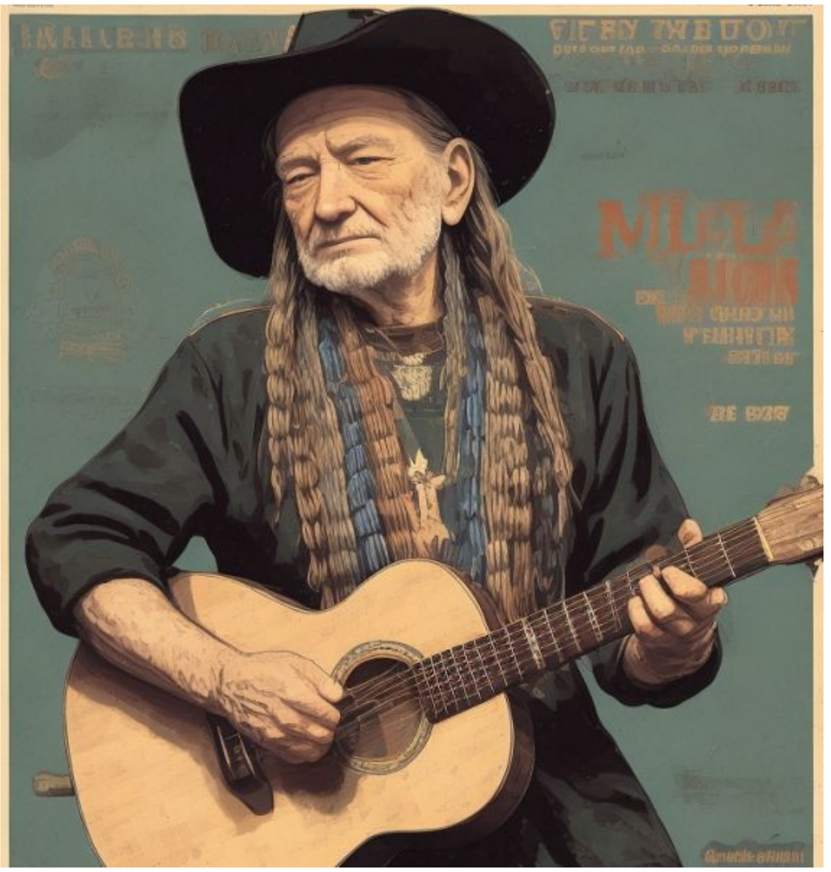
Simian Luo - Yiqin Tan - Longbo Huang - Jian Li - Hang Zhao
Institute for Interdisciplinary Information Sciences, Tsinghua University

Preprint - October 2023

Presented by: Amirhossein Alimohammadi

What is LCM?

Latent Consistency Model (LCM) is basically a consistency model enabling swift inference with minimal steps on any pre-trained LDMs instead of DMs in consistency models.



Main Objective

We are diving into the math behind LCMs to figure out why they are so fast.

Our goal is to really understand a complex formula that plays a key role in their performance.

$$
\begin{aligned}
\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega} - \boldsymbol{z}_{t_{n+1}} &= \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \tilde{\boldsymbol{\epsilon}}_\theta \left(\boldsymbol{z}_t, \omega, \boldsymbol{c}, t \right) \right) \mathrm{d}t \\
&= (1+\omega) \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \boldsymbol{\epsilon}_\theta \left(\boldsymbol{z}_t, \boldsymbol{c}, t \right) \right) \mathrm{d}t - \omega \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \boldsymbol{\epsilon}_\theta \left(\boldsymbol{z}_t, \varnothing, t \right) \right) \mathrm{d}t \\
&\approx (1+\omega)\Psi(\boldsymbol{z}_{t_{n+1}}, t_{n+1}, t_n, \boldsymbol{c}) - \omega\Psi(\boldsymbol{z}_{t_{n+1}}, t_{n+1}, t_n, \varnothing).
\end{aligned}
$$

Background

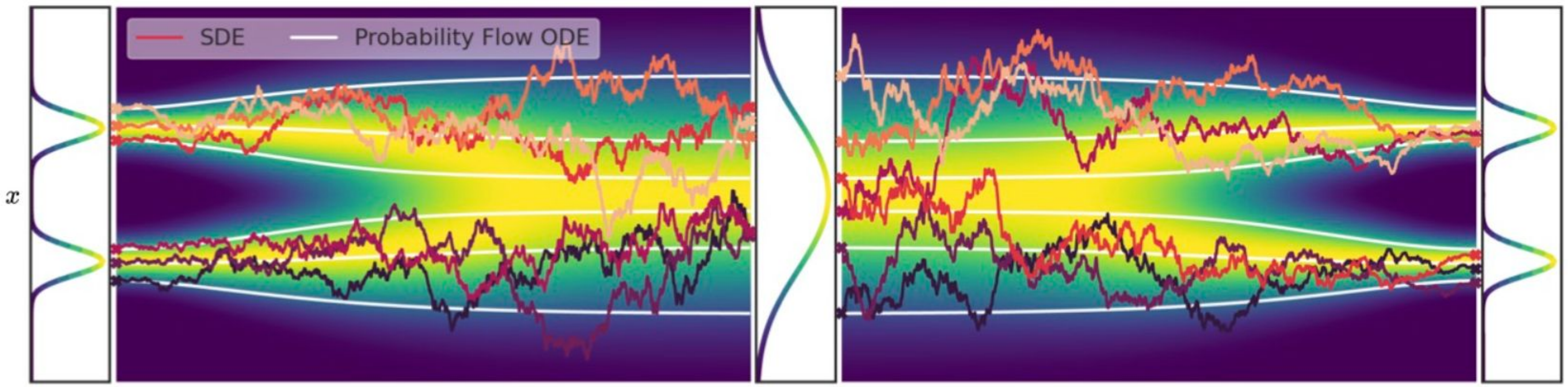
Diffusion Models

Diffusion models start by diffusing $p_{data}(x)$ with a stochastic differential equation (SDE) (Song et al., 2021)

$$\mathrm{d}\mathbf{x}_t = \boldsymbol{\mu}(\mathbf{x}_t, t)\,\mathrm{d}t + \sigma(t)\,\mathrm{d}\mathbf{w}_t$$

A remarkable property of this SDE is the existence of an ordinary differential equation (ODE), dubbed the Probability Flow (PF) ODE.

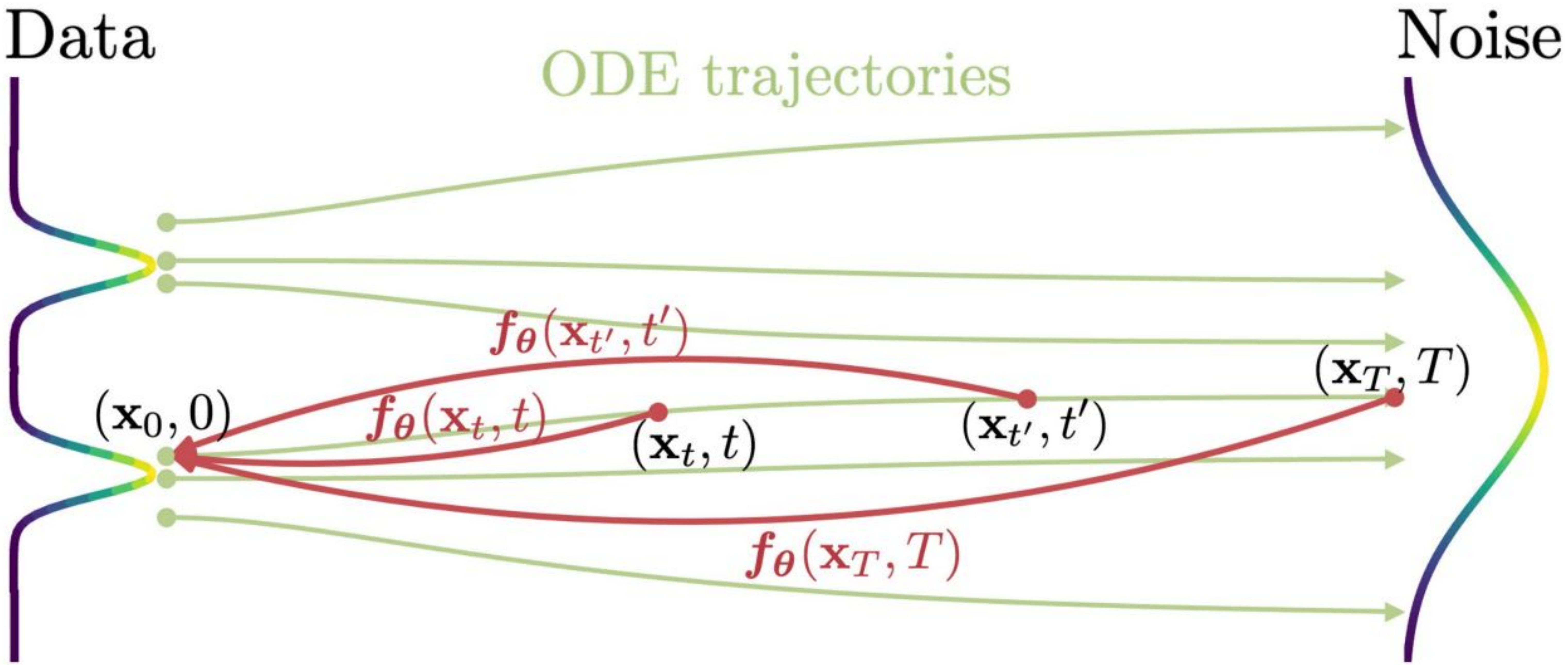
$$\mathrm{d}\mathbf{x}_t = \left[\boldsymbol{\mu}(\mathbf{x}_t, t) - \frac{1}{2}\sigma(t)^2 \nabla \log p_t(\mathbf{x}_t)\right] \mathrm{d}t$$



Consistency Models

They have shown **great potential** as a new type of generative model for **faster sampling** while **preserving generation quality**.

Consistency models are trained to map points on any trajectory of the *PF-ODE* to the trajectory's origin.



Parametrization

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}, t) = c_{\text{skip}}(t)\mathbf{x} + c_{\text{out}}(t)F_{\boldsymbol{\theta}}(\mathbf{x}, t)$$

Benefits:

1. Differentiable formula
2. Enables us to train continuous-time CMs

Algorithm 1 Multistep Consistency Sampling

Input: Consistency model $\boldsymbol{f_\theta}(\cdot, \cdot)$, sequence of time points $\tau_1 > \tau_2 > \cdots > \tau_{N-1}$, initial noise $\hat{\mathbf{x}}_T$

$\mathbf{x} \leftarrow \boldsymbol{f_\theta}(\hat{\mathbf{x}}_T, T)$

for $n = 1$ **to** $N - 1$ **do**

  Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

  $\hat{\mathbf{x}}_{\tau_n} \leftarrow \mathbf{x} + \sqrt{\tau_n^2 - \epsilon^2}\mathbf{z}$

  $\mathbf{x} \leftarrow \boldsymbol{f_\theta}(\hat{\mathbf{x}}_{\tau_n}, \tau_n)$

end for

Output: $\mathbf{x}$

Algorithm 2 Consistency Distillation (CD)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, ODE solver $\Phi(\cdot, \cdot; \boldsymbol{\phi})$, $d(\cdot, \cdot)$, $\lambda(\cdot)$, and μ

$\boldsymbol{\theta}^{-} \leftarrow \boldsymbol{\theta}$

repeat

Sample $\mathbf{x} \sim \mathcal{D}$ and $n \sim \mathcal{U}[\![1, N-1]\!]$

Sample $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 \boldsymbol{I})$

$\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}} \leftarrow \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{\phi})$

$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \boldsymbol{\phi}) \leftarrow$

$\lambda(t_n) d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n))$

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \boldsymbol{\phi})$

$\boldsymbol{\theta}^{-} \leftarrow \operatorname{stopgrad}(\mu \boldsymbol{\theta}^{-} + (1-\mu)\boldsymbol{\theta})$

until convergence

Further details of CD

Running one discretization step of a numerical ODE solver:

$$\hat{\mathbf{x}}_{t_n}^{\phi} := \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \phi)$$

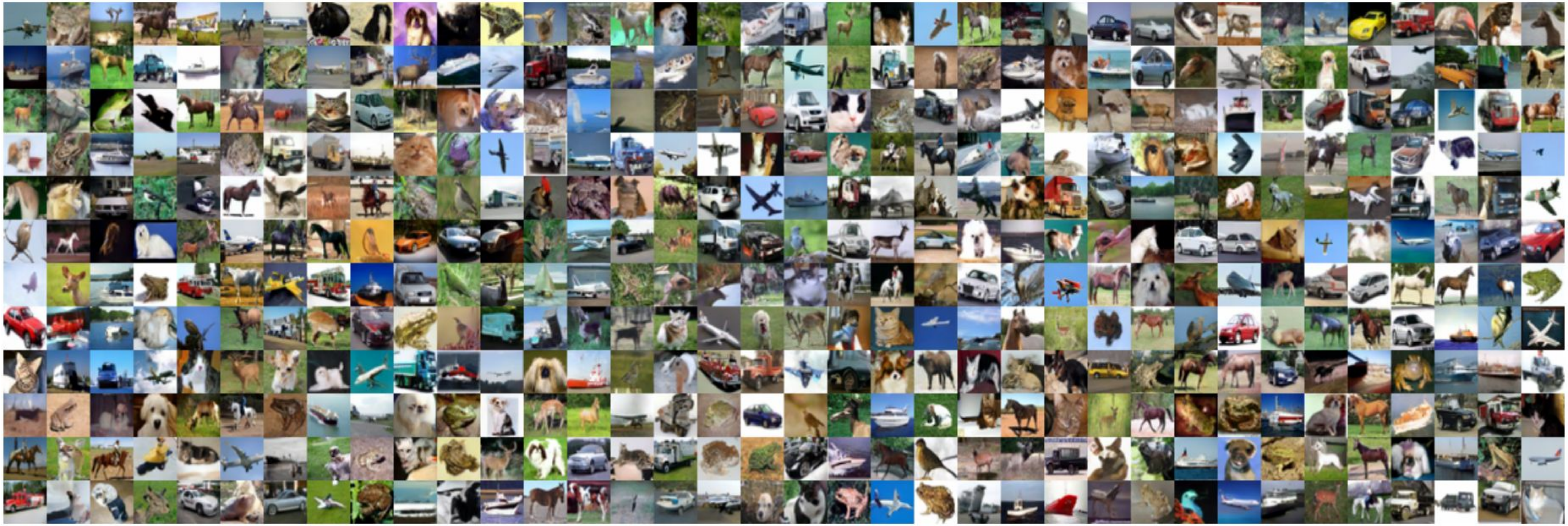
Further details of CD

The consistency distillation loss:

$$\mathcal{L}_{CD}^{N}(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \phi) :=$$
$$\mathbb{E}[\lambda(t_n) d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\mathbf{x}}_{t_n}^{\phi}, t_n))]$$

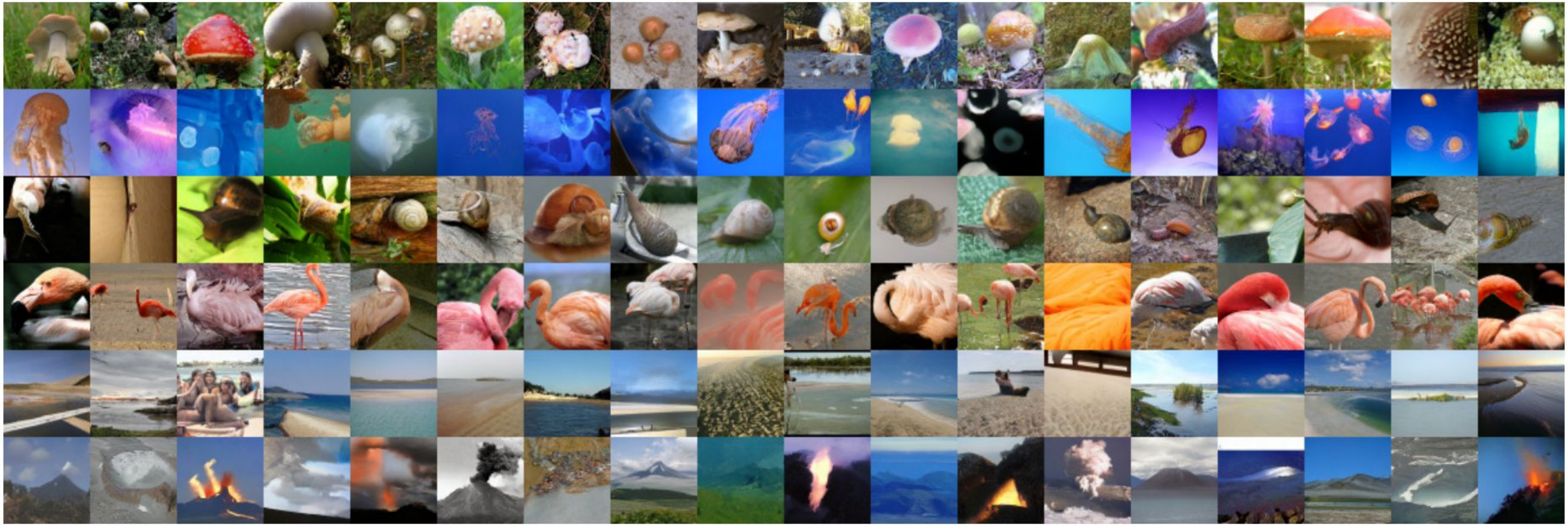
Consistency Models Results (two-step generation)

CIFAR-10 32 * 32



Consistency Models Results (two-step generation)

ImageNet 64 * 64



Latent Consistency Models

Latent Consistency Models Results (1-Step Inference)

Latent Consistency Models Results (2-Steps Inference)

Latent Consistency Models Results (4-Steps Inference)

What are the new features?

Fast, high-resolution image generation

A simple and efficient one-stage guided consistency distillation method

SKIPPING-STEP technique to converge even faster

Again: SDE Equation

In continuous time perspective, the forward process can be described by a stochastic differential equation (SDE):

$$\mathrm{d}\boldsymbol{x}_t = f(t)\boldsymbol{x}_t \,\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t$$

Again: Probability Flow (PF) ODE

$$\frac{\mathrm{d}\boldsymbol{x}_t}{\mathrm{d}t} = f(t)\boldsymbol{x}_t - \frac{1}{2}g^2(t)\nabla_{\boldsymbol{x}} \log q_t\left(\boldsymbol{x}_t\right), \ \ \boldsymbol{x}_T \sim q_T\left(\boldsymbol{x}_T\right)$$

Training the noise prediction model $\epsilon_\theta(x_t, t)$ to fit $-\nabla_x \log q_t(x_t)$ (score function)

$$\frac{\mathrm{d}\boldsymbol{x}_t}{\mathrm{d}t} = f(t)\boldsymbol{x}_t + \frac{g^2(t)}{2\sigma_t}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right), \quad \boldsymbol{x}_T \sim \mathcal{N}\left(\mathbf{0}, \tilde{\sigma}^2\boldsymbol{I}\right)$$

Changing X_t

z_t is image latents, $\epsilon_\theta(z_t, c, t)$ is the noise prediction model, and c is the given condition (e.g text).

$$\frac{\mathrm{d}\boldsymbol{z}_t}{\mathrm{d}t} = f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{z}_t, \mathbf{c}, t\right), \quad \boldsymbol{z}_T \sim \mathcal{N}\left(\mathbf{0}, \tilde{\sigma}^2\boldsymbol{I}\right)$$

Parametrization

$$\boldsymbol{f_\theta}(\boldsymbol{z}, \mathbf{c}, t) = c_{\text{skip}}(t)\boldsymbol{z} + c_{\text{out}}(t)\left(\frac{\boldsymbol{z} - \sigma_t \hat{\boldsymbol{\epsilon}}_\theta(\boldsymbol{z}, \mathbf{c}, t)}{\alpha_t}\right)$$

Loss

LCM aims to predict the solution of the PF-ODE by minimizing the consistency distillation loss:

$$\mathcal{L}_{\mathcal{CD}}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \Psi\right) = \mathbb{E}_{\boldsymbol{z},\boldsymbol{c},n}\left[d\left(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t_{n+1}}, \boldsymbol{c}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\boldsymbol{z}}^{\Psi}_{t_n}, \boldsymbol{c}, t_n)\right)\right]$$

ODE solver

$\hat{\boldsymbol{z}}_{t_n}^{\Psi}$ is an estimation of the evolution of the PF-ODE from $t_{n+1} \rightarrow t_n$ using ODE solver:

$$\hat{\boldsymbol{z}}_{t_n}^{\Psi} - \boldsymbol{z}_{t_{n+1}} = \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \boldsymbol{\epsilon}_\theta \left(\boldsymbol{z}_t, \mathbf{c}, t \right) \right) \mathrm{d}t \approx \Psi(\boldsymbol{z}_{t_{n+1}}, t_{n+1}, t_n, \mathbf{c})$$

One-Stage Guided Distillation

Classifier-free guidance (CFG) is crucial for synthesizing high-quality text-aligned images in SD, typically needing a CFG scale ω over 6.

Previous method (Guided-Distill [Meng et al., 2023]) introduces a two-stage distillation.

It needs at least 45 A100 GPUs Days for 2-step inference while the new method demands merely 32 A100 GPUs Hours training for 2-step inference.

CFG used in reverse diffusion process

CFG used in reverse diffusion process:

$$\tilde{\boldsymbol{\epsilon}}_\theta\left(\boldsymbol{z}_t, \omega, \mathbf{c}, t\right) := (1+\omega)\boldsymbol{\epsilon}_\theta\left(\boldsymbol{z}_t, \mathbf{c}, t\right) - \omega\boldsymbol{\epsilon}_\theta\left(\boldsymbol{z}_t, \varnothing, t\right)$$

To sample from the guided reverse process, we need to solve the following augmented PF-ODE:

$$\frac{\mathrm{d}\boldsymbol{z}_t}{\mathrm{d}t} = f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t}\tilde{\boldsymbol{\epsilon}}_\theta\left(\boldsymbol{z}_t, \omega, \mathbf{c}, t\right), \quad \boldsymbol{z}_T \sim \mathcal{N}\left(\mathbf{0}, \tilde{\sigma}^2\boldsymbol{I}\right)$$

New Loss

The consistency loss is the same as page 25 except that we use augmented consistency function $f_\theta(z_t, \omega, c, t)$.

$$\mathcal{L}_{CD}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \Psi\right) = \mathbb{E}_{\boldsymbol{z},\boldsymbol{c},\omega,n}\left[d\left(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t_{n+1}}, \omega, \boldsymbol{c}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega}, \omega, \boldsymbol{c}, t_n)\right)\right]$$

New ODE solver

$\hat{\boldsymbol{z}}^{\Psi}_{t_n}$ is an estimation of the evolution of the PF-ODE from $t_{n+1} \rightarrow t_n$ using ODE solver:

$$
\begin{aligned}
\hat{\boldsymbol{z}}^{\Psi,\omega}_{t_n} - \boldsymbol{z}_{t_{n+1}} &= \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \tilde{\boldsymbol{\epsilon}}_\theta \left(\boldsymbol{z}_t, \omega, \boldsymbol{c}, t\right) \right) \mathrm{d}t \\
&= (1+\omega) \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \boldsymbol{\epsilon}_\theta \left(\boldsymbol{z}_t, \boldsymbol{c}, t\right) \right) \mathrm{d}t - \omega \int_{t_{n+1}}^{t_n} \left(f(t)\boldsymbol{z}_t + \frac{g^2(t)}{2\sigma_t} \boldsymbol{\epsilon}_\theta \left(\boldsymbol{z}_t, \varnothing, t\right) \right) \mathrm{d}t \\
&\approx (1+\omega)\Psi(\boldsymbol{z}_{t_{n+1}}, t_{n+1}, t_n, \boldsymbol{c}) - \omega\Psi(\boldsymbol{z}_{t_{n+1}}, t_{n+1}, t_n, \varnothing).
\end{aligned}
$$

Distillation Problem

What is the problem?

DDM typically train noise prediction models with a long time-step schedule.(SD - 1000 steps)

LCM needs to sample across all steps.

Since $t_n - t_{n+1}$ is tiny, z_{t_n} and $z_{t_{n+1}}$ are already close to each other.

→ Small consistency loss leading to slow convergence.

Skipping Time Steps

Setting k=1 leading to slow convergence.
Very large k leading to large approximation errors of the ODE solvers.
Setting k=20, drastically reducing the length of time schedule from thousands to dozens.

$$\mathcal{L}_{\mathcal{CD}}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \Psi\right) = \mathbb{E}_{\boldsymbol{z},\boldsymbol{c},\omega,n}\left[d\left(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t_{n+k}}, \omega, \boldsymbol{c}, t_{n+k}), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega}, \omega, \boldsymbol{c}, t_n)\right)\right]$$

$$\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega} \longleftarrow \boldsymbol{z}_{t_{n+k}} + (1+\omega)\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \boldsymbol{c}) - \omega\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \varnothing)$$

DDIM PF-ODE solver

$$\Psi_{\text{DDIM}}(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \boldsymbol{c}) = \underbrace{\frac{\alpha_{t_n}}{\alpha_{t_{n+k}}}\boldsymbol{z}_{t_{n+k}} - \sigma_{t_n}\left(\frac{\sigma_{t_{n+k}} \cdot \alpha_{t_n}}{\alpha_{t_{n+k}} \cdot \sigma_{t_n}} - 1\right)\hat{\boldsymbol{\epsilon}}_\theta(\boldsymbol{z}_{t_{n+k}}, \boldsymbol{c}, t_{n+k})}_{\text{DDIM Estimated } \boldsymbol{z}_{t_n}} - \boldsymbol{z}_{t_{n+k}}$$

Algorithm 3 Latent Consistency Distillation (LCD)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, ODE solver $\Psi(\cdot,\cdot,\cdot,\cdot)$, distance metric $d(\cdot,\cdot)$, EMA rate μ, noise schedule $\alpha(t), \sigma(t)$, guidance scale $[w_{\text{min}}, w_{\text{max}}]$, skipping interval k, and encoder $E(\cdot)$

Encoding training data into latent space: $\mathcal{D}_z = \{(\boldsymbol{z}, \boldsymbol{c}) | \boldsymbol{z} = E(\boldsymbol{x}), (\boldsymbol{x}, \boldsymbol{c}) \in \mathcal{D}\}$

$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$

repeat

  Sample $(\boldsymbol{z}, \boldsymbol{c}) \sim \mathcal{D}_z$, $n \sim \mathcal{U}[1, N-k]$ and $\omega \sim [\omega_{\text{min}}, \omega_{\text{max}}]$

  Sample $\boldsymbol{z}_{t_{n+k}} \sim \mathcal{N}(\alpha(t_{n+k})\boldsymbol{z}; \sigma^2(t_{n+k})\mathbf{I})$

  $\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega} \leftarrow \boldsymbol{z}_{t_{n+k}} + (1+\omega)\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \boldsymbol{c}) - \omega\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \varnothing)$

  $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \Psi) \leftarrow d(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t_{n+k}}, \omega, \boldsymbol{c}, t_{n+k}), \boldsymbol{f}_{\boldsymbol{\theta}^-}(\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega}, \omega, \boldsymbol{c}, t_n))$

  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-)$

  $\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu\boldsymbol{\theta}^- + (1-\mu)\boldsymbol{\theta})$

until convergence

Result

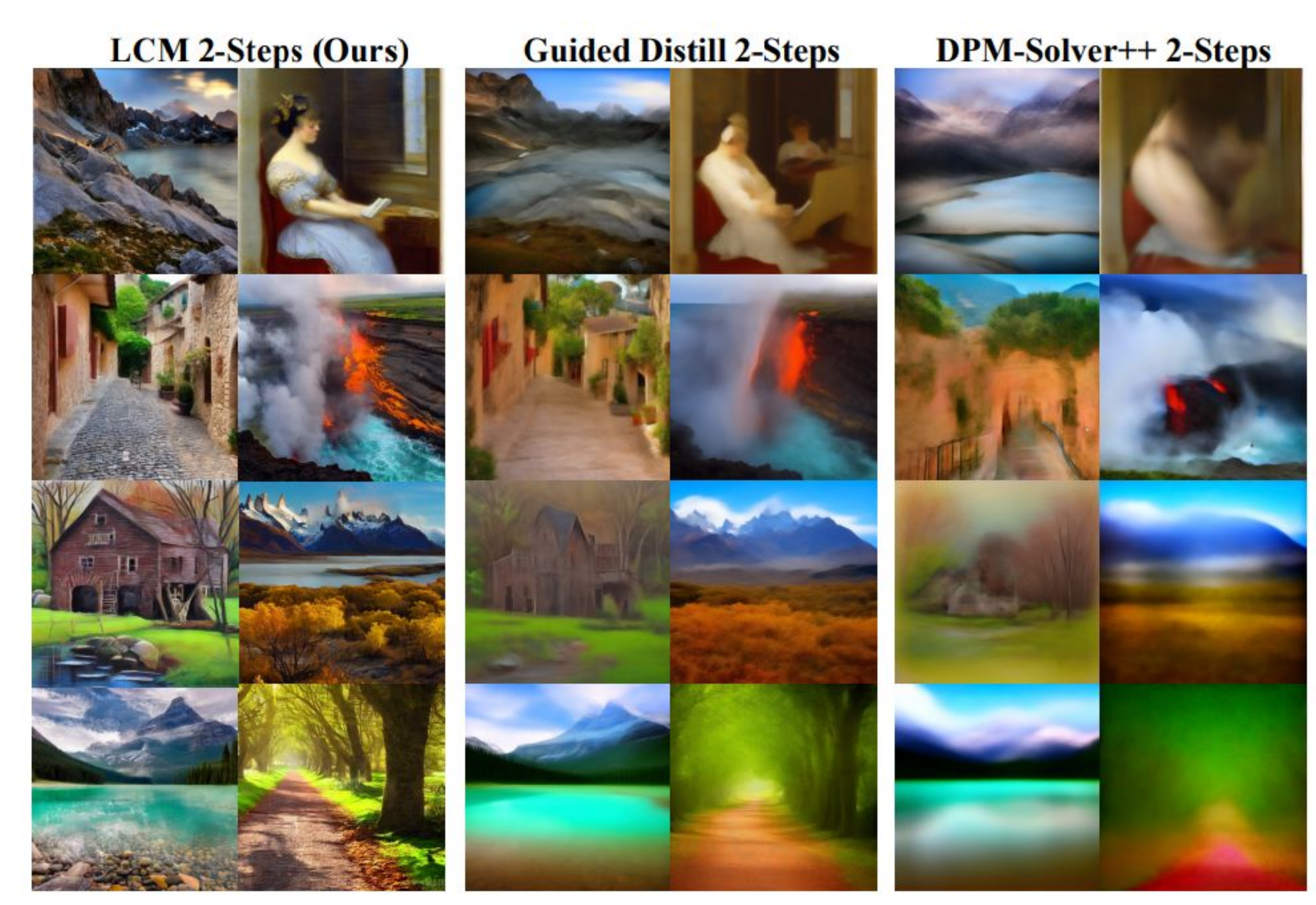
Model (512 × 512) Reso	FID ↓				CLIP Score ↑			
	1 Step	2 Steps	4 Steps	8 Steps	1 Steps	2 Steps	4 Steps	8 Steps
DDIM (Song et al., 2020a)	183.29	81.05	22.38	13.83	6.03	14.13	25.89	29.29
DPM (Lu et al., 2022a)	185.78	72.81	18.53	12.24	6.35	15.10	26.64	29.54
DPM++ (Lu et al., 2022b)	185.78	72.81	18.43	12.20	6.35	15.10	26.64	**29.55**
Guided-Distill (Meng et al., 2023)	108.21	33.25	15.12	13.89	12.08	22.71	27.25	28.17
LCM (Ours)	**35.36**	**13.31**	**11.10**	**11.84**	**24.14**	**27.83**	**28.69**	28.84

Table 1: Quantitative results with $\omega = 8$ at 512×512 resolution. LCM significantly surpasses baselines in the 1-4 step region on LAION-Aesthetic-6+ dataset. For LCM, DDIM-Solver is used with a skipping step of $k = 20$.

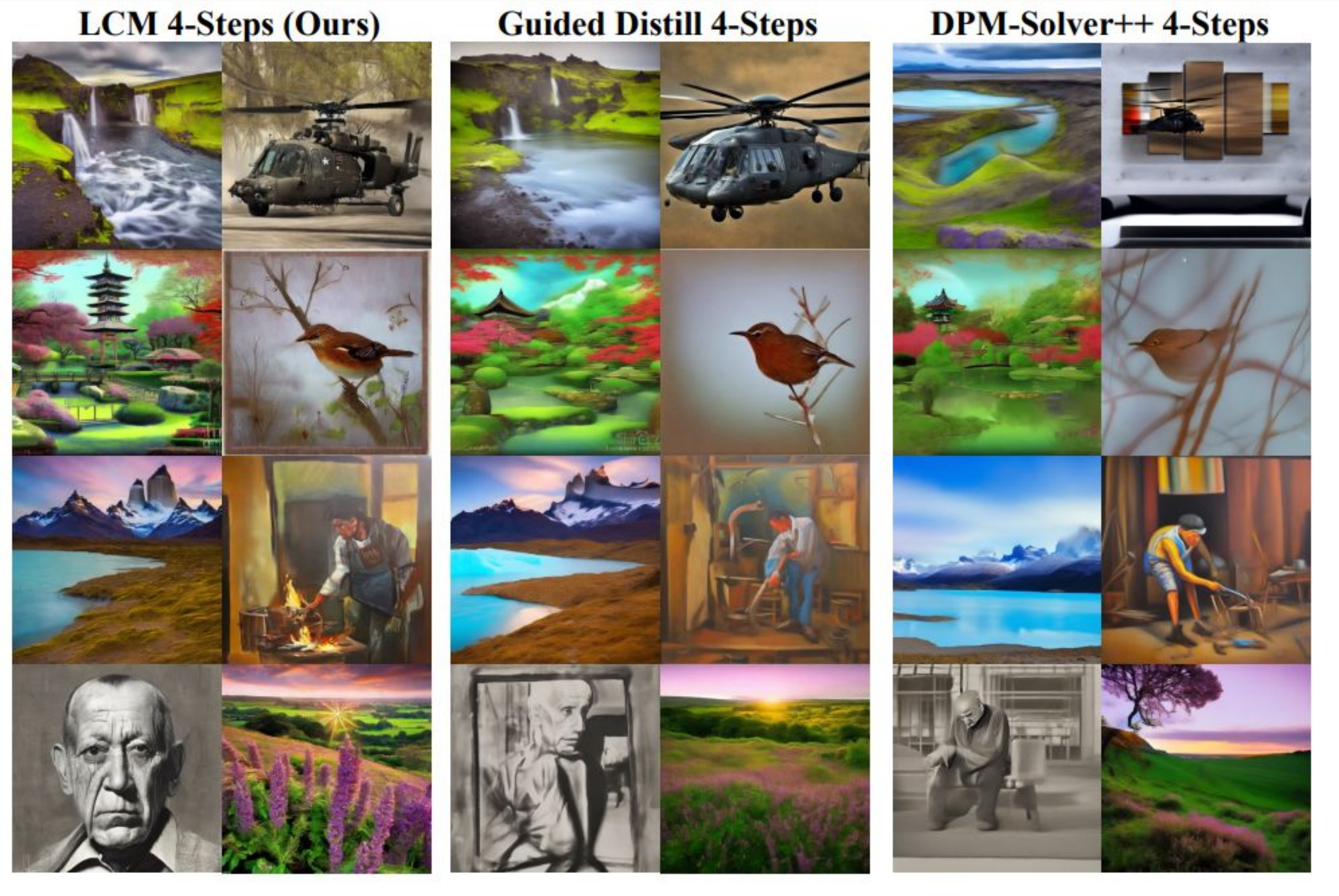
Result

Model (768 × 768) Reso	FID ↓				CLIP Score ↑			
	1 Step	2 Steps	4 Steps	8 Steps	1 Steps	2 Steps	4 Steps	8 Steps
DDIM (Song et al., 2020a)	186.83	77.26	24.28	15.66	6.93	16.32	26.48	29.49
DPM (Lu et al., 2022a)	188.92	67.14	20.11	**14.08**	7.40	17.11	27.25	29.80
DPM++ (Lu et al., 2022b)	188.91	67.14	20.08	14.11	7.41	17.11	27.26	**29.84**
Guided-Distill (Meng et al., 2023)	120.28	30.70	16.70	14.12	12.88	24.88	28.45	29.16
LCM (Ours)	**34.22**	**16.32**	**13.53**	14.97	**25.32**	**27.92**	**28.60**	28.49

Table 2: Quantitative results with $\omega = 8$ at 768×768 resolution. LCM significantly surpasses the baselines in the 1-4 step region on LAION-Aesthetic-6.5+ dataset. For LCM, DDIM-Solver is used with a skipping step of $k = 20$.

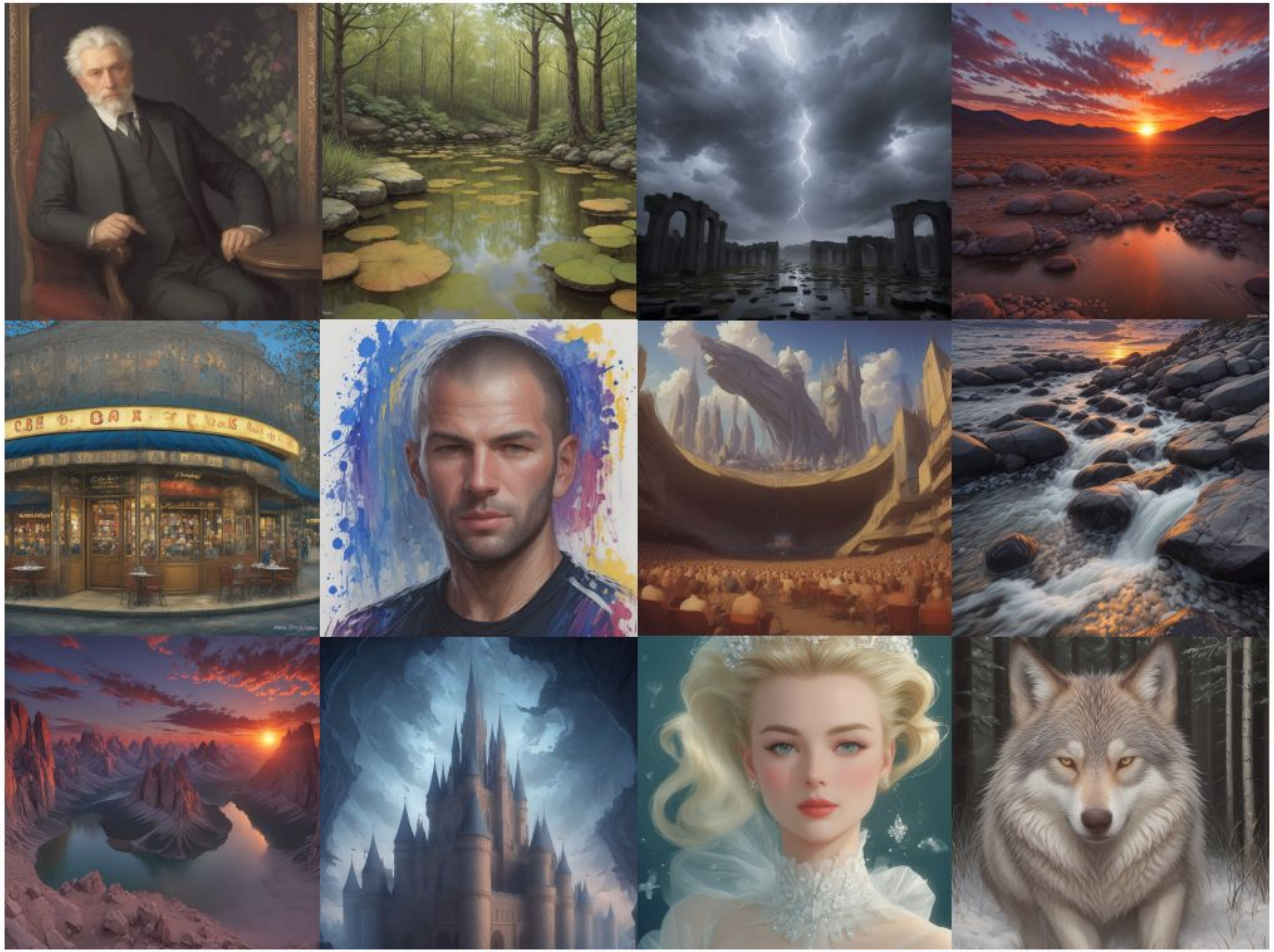


Text-to-Image generation results on LAION-Aesthetic-6.5+ with 2-, 4-step inference.

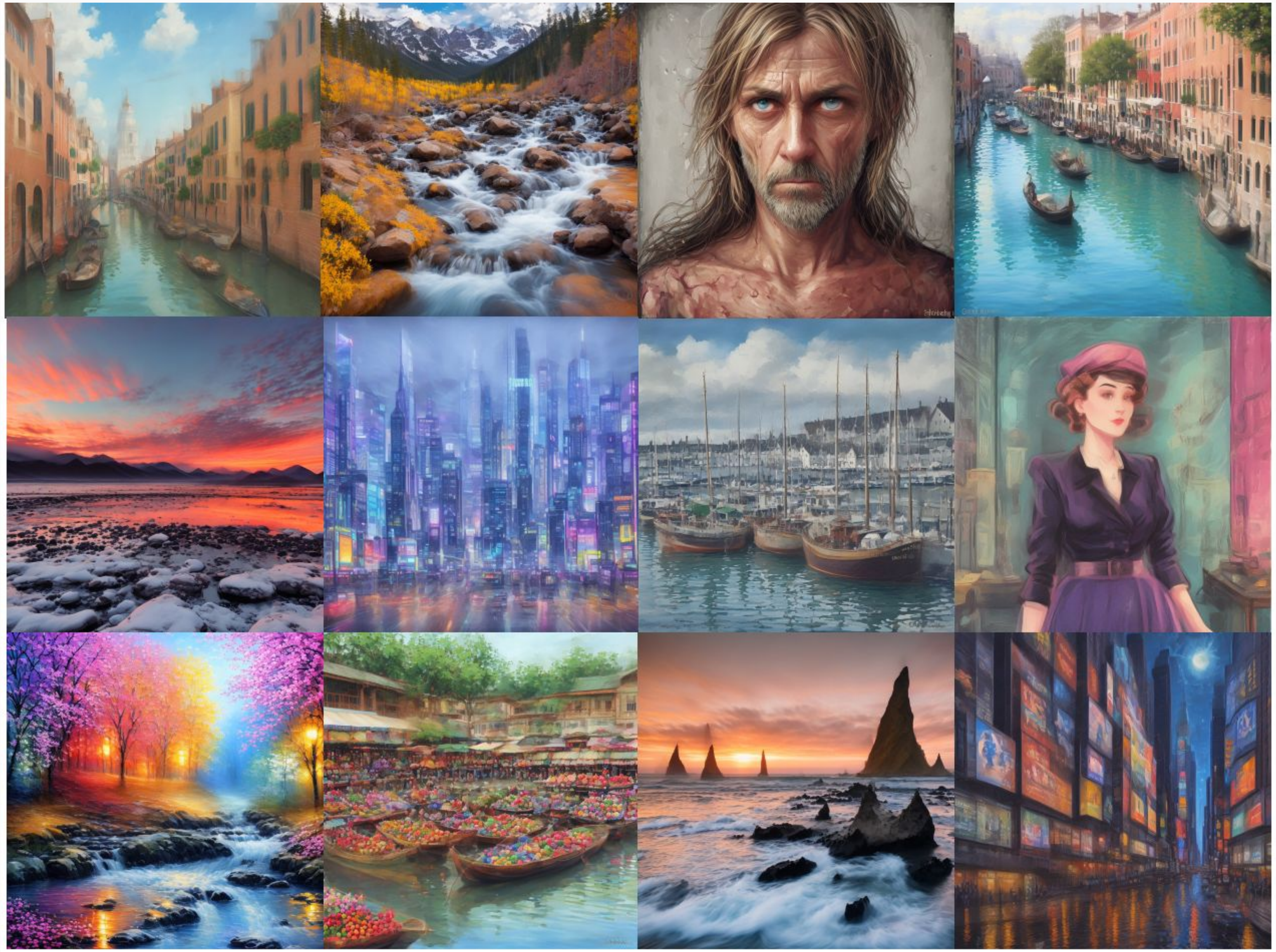


Text-to-Image generation results on LAION-Aesthetic-6.5+ with 2-, 4-step inference.

4-step LCMs using different CFG scales ω. LCMs utilize one-stage guided distillation to directly incorporate CFG scales ω. Larger ω enhances image quality.



More generated images results with LCM 4-steps inference (768×768 Resolution). We employ LCM to distill the Dreamer-V7 version of SD in just 4,000 training iterations.



More generated images results with LCM 2-steps inference (768×768 Resolution). We employ LCM to distill the Dreamer-V7 version of SD in just 4,000 training iterations.

How can we even do better?

By using LORA, we can expand LCM's scope to larger models with significantly less memory consumption, achieving superior image generation quality.

It is basically what *LCM-LORA: A UNIVERSAL STABLE-DIFFUSION ACCELERATION MODULE* does.

Model	SD-V1.5	SSD-1B	SDXL
# Full Parameters	0.98B	1.3B	3.5B
# LoRA Trainable Parameters	67.5M	105M	197M

Thank you!

- Thank you for your attention!
- I appreciate your time and interest.
- If you have any questions, please feel free to ask.
- Contact information: alimohammadiamirhossein@gmail.com