CCM: Real-Time Controllable Visual Content Creation Using Text-to-Image Consistency Models

Jie Xiao [1] **Kai Zhu** [2] **Han Zhang** [3] **Zhiheng Liu** [1] **Yujun Shen** [4] **Zhantao Yang** [2] **Ruili Feng** [2] **Yu Liu** [2] **Xueyang Fu** [1] **Zheng-Jun Zha** [1]

Abstract

Consistency Models (CMs) have showed a promise in creating high-quality images with few steps. However, the way to add new conditional controls to the pre-trained CMs has not been explored. In this paper, we explore the pivotal subject of leveraging the generative capacity and efficiency of consistency models to facilitate controllable visual content creation via ControlNet. First, it is observed that ControlNet trained for diffusion models (DMs) can be directly applied to CMs for high-level semantic controls but sacrifice image low-level details and realism. To tackle with this issue, we develop a CMs-tailored training strategy for ControlNet using the consistency training (Song et al., 2023). It is substantiated that ControlNet can be successfully established through the consistency training technique. Besides, a unified adapter can be trained utilizing the consistency training, which enhances the adaptation of DM's ControlNet. We quantitatively and qualitatively evaluate all strategies across various conditional controls, including sketch, hed, canny, depth, human pose, low-resolution image and masked image, with the pre-trained text-to-image latent consistency models.

1. Introduction

Consistency Models (CMs) (Song et al., 2023; Song & Dhariwal, 2023; Luo et al., 2023a;b) have emerged as a competitive family of generative models that can generate high-quality images in one or few steps. CMs can be distilled from a pre-tranined diffusion model or trained in isolation from data (Song et al., 2023; Song & Dhariwal, 2023). Recently, latent consistency models (LCMs) (Luo et al., 2023a;b) have been successfully distilled from Stable Diffusion (SD) (Rombach et al., 2022), achieving significant acceleration in the speed of text conditioned image generation. Compared with the glorious territory of diffusion models (DMs), an essential concern is whether there exists effective solutions for CMs to accommodate additional conditional controls. Inspired by the success of ControlNet (Zhang et al., 2023) to text-to-image DMs, we consider to address this issue by training ControlNet for CMs.

In this work, we inverstigate the training strategies of ControlNet for CMs. Given the connection that CMs directly project any point of a probability flow ordinary differential equation (PF ODE) trajectory to data and DMs produce data by iterating an ODE solver along the PF ODE (Song et al., 2020), we assume that the learned knowledge of ControlNet is (partially) transferable to CMs. Therefore, the first attempt is to train ControlNet based on DMs and then directly apply the trained ControlNet to CMs. The advantage is that one can readily re-use the off-the-shelf ControlNet of DMs, but meanwhile at the cost of: i) sub-optimal performance. Due to the gap between CMs and DMs, the transfer may be imperfect; ii) indirect training when adding new controls. That is, one has to utilize DMs as an agent to train a new ControlNet and then rely on the strong generalization ability of ControlNet to apply to CMs.

Recent researches (Song et al., 2023; Song & Dhariwal, 2023) point out that CMs, as a new family of generative models, can be trained in isolation from data by the consistency training technique. Inspired by this, we treat the integration of the pre-trained text-to-image CM and ControlNet as a new conditional CM with only ControlNet trainable. Our solution is simple: directly training ControlNet using the consistency training. We find that ControlNet can be successfully established from scratch without reliance on DMs[1]. Figure 1 shows the performance and efficiency comparison among different strategies. The results reveal that the consistency model can create controllable visual content with comparable performance and much faster generation

[1]University of Science and Technology of China, Hefei, China [2]Alibaba Group [3]Shanghai Jiao Tong University [4]Ant Group. Correspondence to: Xueyang Fu <xyfu@ustc.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

[1]Even if CMs may be trained by the consistency distillation from DMs.

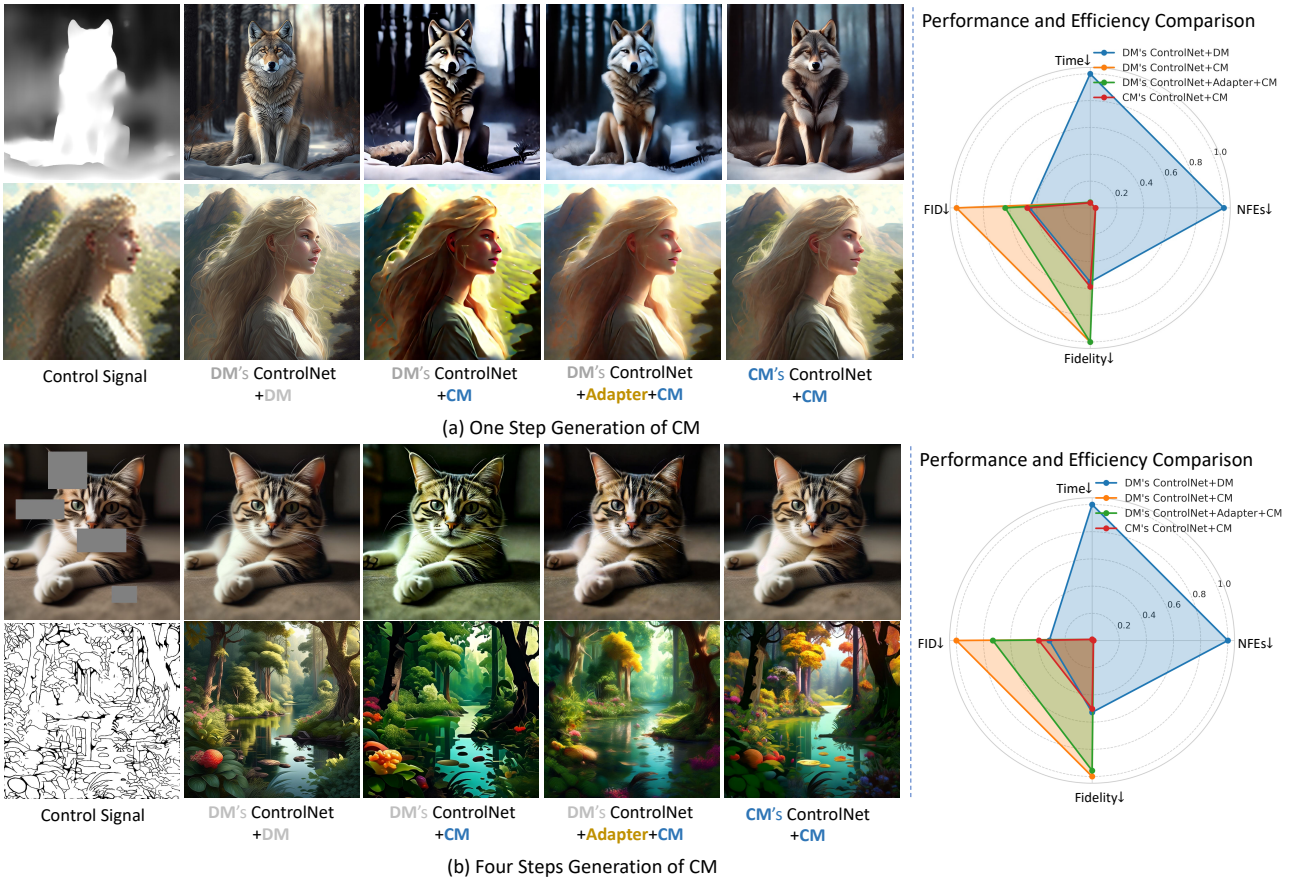


Figure 1. Visual comparison of different strategies of adding controls. Image resolution: 1024x1024. To measure performance, we use FID to assess image quality and Fidelity to evaluate consistency with the control signals (Equation (16)). We employ the number of function evaluations (NFEs) and running time to measure efficiency. All scores are normalized to $[0, 1]$. It can be observed that compared with the diffusion model, the consistency model can achieve comparable performance with much lower cost.

speed compared to the diffusion model. Based on these results, we also train a multi-condition shared adapter using the consistency training for better adaptation of DM's ControlNet. Experiments across various conditions including sketch, hed, canny, depth, human pose, low-resolution image (*i.e.*, $16\times$ image super-resolution) and masked image (*i.e.*, image inpainting) suggest that:

- ControlNet of DM can transfer high-level semantic controls to CM; however, it often fails to accomplish low-level fine controls;
- CM's ControlNet can be trained from scratch using the consistency training technique. Empirically, we can find that consistency training can accomplish more satisfactory conditional generation;
- A unified adapter trained with the consistency training technique is capable of mitigating the discrepancy between DMs and CMs, thereby facilitating to transfer DM's ControlNet.

2. Preliminary

Denoising diffusion probabilistic models (DDPMs) represent a category of latent variable models designed to approximate the true data distribution, denoted as $q(\boldsymbol{x}_0)$, with a learned model distribution $p(\boldsymbol{x}_0)$ (Ho et al., 2020). DDPMs comprise a forward diffusion process that progressively injects Gaussian noise into the data over a series of T steps, and a reverse generative process that synthesizes data by progressively removing noise across the same number of steps. Formulaly, the forward diffusion process is a Markov chain which is of the form

$$\begin{aligned} q\left(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0\right) &:= \prod_{t=1}^{T} q\left(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}\right), \\ q\left(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}\right) &:= \mathcal{N}\left(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right), \end{aligned} \tag{1}$$

where $\{\beta_t\}_{t=0}^T$ is the variance schedule. $\{\boldsymbol{x}_t\}_{t=0}^T$ are latent variables. A notable property of diffusion process is that the conditional distribution of x_t given x_0 is

$$\begin{aligned} &q\left(\boldsymbol{x}_t|\boldsymbol{x}_0\right)=\mathcal{N}\left(\boldsymbol{x}_t;\sqrt{\alpha_t}\boldsymbol{x}_0,(1-\alpha_t)\boldsymbol{I}\right), \\ &\text{where}\quad \alpha_t=\prod_{i=0}^{t}\left(1-\beta_i\right). \end{aligned} \tag{2}$$

The reverse generative process starts by sampling a Gaussian noise $p(\boldsymbol{x}_T)=\mathcal{N}(\boldsymbol{x}_T;\boldsymbol{0},\boldsymbol{I})$ and further proceeding with the transition probability density $p_\theta\left(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t\right)$, where

$$\begin{aligned} p_\theta\left(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t\right)&\approx q\left(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0\right) \\ &=\mathcal{N}\left(\boldsymbol{x}_{t-1};\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,\boldsymbol{x}_0),\sigma_t^2\boldsymbol{I}\right), \end{aligned} \tag{3}$$

The mean $\boldsymbol{\mu}_\theta\left(\boldsymbol{x}_t,\boldsymbol{x}_0\right)$ and variance σ_t^2 have the closed form

$$\begin{aligned} \boldsymbol{\mu}_\theta\left(\boldsymbol{x}_t,\boldsymbol{x}_0\right)&=\frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t-\frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t,t\right)\right), \\ \sigma_t^2&=\frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t. \end{aligned} \tag{4}$$

Song et al. (2021b) further figures out that the aforementioned reverse process is a discretization of a continuous-time stochastic process, described by the following reverse-time stochastic differential equation (SDE):

$$\mathrm{d}\boldsymbol{x}_t=\left[f\left(t\right)\boldsymbol{x}_t-g^2\left(t\right)\nabla_{\boldsymbol{x}_t}\log p_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t+g\left(t\right)\mathrm{d}\bar{\boldsymbol{w}}_t, \tag{5}$$

where $\bar{\boldsymbol{w}}_t$ is a standard Wiener process in the reverse time, $f(t)=\frac{1}{2}\frac{\mathrm{d}\log\alpha(t)}{\mathrm{d}t}$, $g(t)=(1-\alpha(t))\frac{\mathrm{d}}{\mathrm{d}t}\frac{1-\alpha(t)}{\alpha(t)}$, and $\alpha(t)$ is a continuous version of α_t. For the reverse-time SDE, Song et al. (2021b) further prove that there exists a corresponding probability flow ODE (PF ODE) that shares the same marginal distribution:

$$\mathrm{d}\boldsymbol{x}_t=\left[f(t)\boldsymbol{x}_t-\frac{1}{2}g^2(t)\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]\mathrm{d}t. \tag{6}$$

With this probability flow ODE, one can generate an image from a Gaussian noise and vice versa.

3. Method

Our method consists of four parts. First, we briefly describe how to train a text-to-image consistency model $\boldsymbol{f}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t,t;\boldsymbol{c}_{\text{txt}}\right)$ from a pre-trained text-to-image diffusion model $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}\left(\boldsymbol{x}_t,t;\boldsymbol{c}_{\text{txt}}\right)$ in Section 3.1. We next introduces the first approach to train a ControlNet for a new condition $\boldsymbol{c}_{\text{ctrl}}$ by utilizing the diffusion model as an agent in Section 3.2. Then, we propose to use the consistency training to train a ControlNet from scratch for the pre-trained text-to-image consistency model in Section 3.3. Last, we introduce a unified adapter that enables the rapid swift of multiple DMs-based ControlNets to CMs in Section 3.4. Figure 2 presents overview of the proposed strategies. We summarize the involved symbols in Table 1 to help with readability.

Table 1. Summary of symbols.

$\boldsymbol{\phi}$	trainable paramaters of diffusion model
$\boldsymbol{\theta}$	trainable paramaters of consistency model
$\boldsymbol{\psi}$	trainable paramaters of ControlNet
$\boldsymbol{\Delta\psi}$	trainable paramaters of adapter
$\boldsymbol{\theta}^-$	exponential moving average of $\boldsymbol{\theta}$
$\boldsymbol{\epsilon}_{\{\cdot\}}$	noise-prediction diffusion model
$\boldsymbol{f}_{\{\cdot\}}$	consistency model
$\boldsymbol{h}$	condition extractor
$\boldsymbol{c}_{\text{txt}}$	text prompt
$\boldsymbol{c}_{\text{ctrl}}$	new contional control
$\boldsymbol{x}$ / $\boldsymbol{x}_t$	image / latent (noisy image)

3.1. Text-to-Image Consistency Model

The first step is to acquire a foundational text-to-image consistency model. Consistency models (CMs) (Song et al., 2023), a new family of generative models, can achieve high sample quality with few sampling steps. Given the PF ODE trajectory $\{\boldsymbol{x}_t\}_{t\in[\epsilon,T]}$, the core of CMs, symbolized as $\boldsymbol{f}_{\boldsymbol{\theta}}$, is to estimate the consistency function, which is defined as $\boldsymbol{f}:(\boldsymbol{x}_t,t)\mapsto\boldsymbol{x}_\epsilon$. The consistency function should satisfy the *boundary condition*:$\boldsymbol{f}\left(\boldsymbol{x}_\epsilon,\epsilon\right)=\boldsymbol{x}_\epsilon$. To implement this constraint, consistency model is parameterized as

$$\boldsymbol{f}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t,t\right)=c_{\text{skip}}\left(t\right)\boldsymbol{x}_t+c_{\text{out}}\left(t\right)F_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t,t\right). \tag{7}$$

By adopting variance preserving noise schedule: $\boldsymbol{x}_t=\alpha_t\boldsymbol{x}_\epsilon+\sqrt{1-\alpha_t^2}\boldsymbol{\epsilon}$, we re-derive (see Appendix A) that $c_{\text{skip}}\left(t\right)$ and $c_{\text{out}}\left(t\right)$ take the form of

$$\begin{aligned} c_{\text{skip}}\left(t\right)&=\frac{\alpha_t\sigma_{\text{data}}^2}{1-\alpha_t^2+\alpha_t^2\sigma_{\text{data}}^2}, \\ c_{\text{out}}\left(t\right)&=\frac{\sqrt{1-\alpha_t^2}\sigma_{\text{data}}}{\sqrt{1-\alpha_t^2+\alpha_t^2\sigma_{\text{data}}^2}}. \end{aligned} \tag{8}$$

Song et al. (2023) introduced two methods to train consistency models: consistency distillation from pre-trained text-to-image diffusion models or consistency training from data. Consistency distillation uses the pre-trained diffusion models to estimate score function (parameterized by $\boldsymbol{\phi}$). Given an arbitrary noisy latent $\left(\boldsymbol{x}_{t_{n+1}},t_{n+1}\right)$, an ODE solver is employed to estimate the adjacent latent with less noise, denoted as $\left(\hat{\boldsymbol{x}}_{t_n}^{\boldsymbol{\phi}},t_n\right)$. $\left\{\left(\boldsymbol{x}_{t_{n+1}},t_{n+1}\right),\left(\hat{\boldsymbol{x}}_{t_n}^{\boldsymbol{\phi}},t_n\right)\right\}$ belongs to the same PF ODE trajectory. Then, consistency models can be trained by enforcing self-consistency property: the outputs are consistent for arbitrary pairs of $(\boldsymbol{x}_t,t)$ of the same PF ODE trajectory. The final consistency distillation loss for the consistency model $\boldsymbol{f}_{\boldsymbol{\theta}}$ is defined as

$$\begin{aligned} \mathcal{L}_{\text{CD}}^N\left(\boldsymbol{\theta},\boldsymbol{\theta}^-;\boldsymbol{\phi}\right)=\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}_{t_{n+1}},\boldsymbol{c}_{\text{txt}},n}\Big[\lambda\left(t_n\right)\cdot \\ d\left(\boldsymbol{f}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_{t_{n+1}},t_{n+1};\boldsymbol{c}_{\text{txt}}\right),\boldsymbol{f}_{\boldsymbol{\theta}^-}\left(\hat{\boldsymbol{x}}_{t_n}^{\boldsymbol{\phi}},t_n;\boldsymbol{c}_{\text{txt}}\right)\right)\Big], \end{aligned} \tag{9}$$

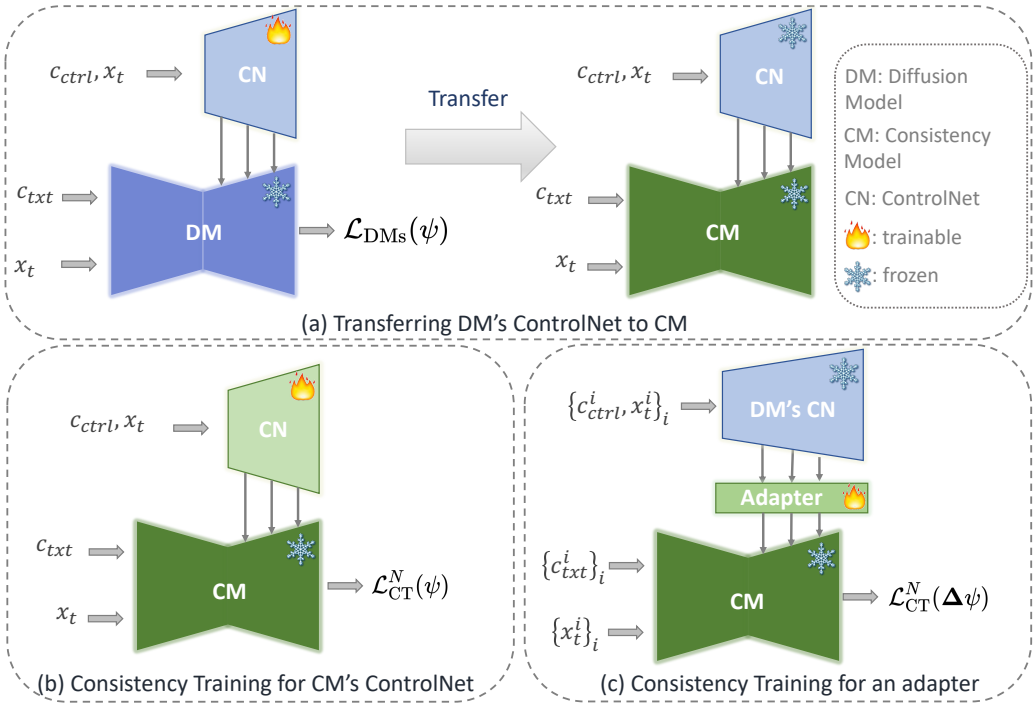


Figure 2. Overview of training strategies for ControlNet. (a) Training a ControlNet based on the text-to-image diffusion model (DM) and directly applying it to the text-to-image consistency model (CM); (b) consistency training for ControlNet based on the text-to-image consistency model; (c) consistency training for a unified adapter to utilize better transfer of DM's ControlNet.

where $\boldsymbol{x} \sim p_{\text{data}}$, $\boldsymbol{x}_t \sim \mathcal{N}\left(\sqrt{\alpha_t}\boldsymbol{x}, (1-\alpha_t)\boldsymbol{I}\right)$ and $n \sim \mathcal{U}\left([1, N-1]\right)$. $\mathcal{U}\left([1, N-1]\right)$ denotes the uniform distribution over $\{1, 2, \ldots, N-1\}$. According to the convention in Song et al. (2023), $\boldsymbol{f}_{\boldsymbol{\theta}^-}$ is the "teacher network" which evolves according to $\boldsymbol{\theta}^- = \text{stopgrad}\left(\mu\boldsymbol{\theta}^- + (1-\mu)\,\boldsymbol{\theta}\right)$ and $\boldsymbol{f}_{\boldsymbol{\theta}}$ is the "student network".

3.2. Applying ControlNet of Text-to-Image Diffusion Models

Given a pre-trained text-to-image diffusion model $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}\left(\boldsymbol{x}_t, t; \boldsymbol{c}_{\text{txt}}\right)$, to add a new control $\boldsymbol{c}_{\text{ctrl}}$, a ControlNet (parameterized by $\boldsymbol{\psi}$) can be trained by minimizing the mean square error function of diffusion models $\mathcal{L}_{\text{DMs}}\left(\boldsymbol{\psi}\right)$ (Ho et al., 2020), where $\mathcal{L}_{\text{DMs}}\left(\boldsymbol{\psi}\right)$ takes the form of

$$\mathcal{L}_{\text{DMs}}\left(\boldsymbol{\psi}\right) = \mathbb{E}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\{\boldsymbol{\phi},\boldsymbol{\psi}\}}\left(\boldsymbol{x}_t, t; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}}\right)\right\|_2^2\right]. \quad (10)$$

In Equation (10), $t \sim \text{Uniform}\left(\{1, \ldots, T\}\right)$, $\boldsymbol{x} \sim p_{\text{data}}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{I}\right)$. Suppose that $\boldsymbol{\psi}^* = \arg\min_{\boldsymbol{\psi}} \mathcal{L}_{\text{DMs}}\left(\boldsymbol{\psi}\right)$, the trained ControlNet $\{\boldsymbol{\psi}^*\}$ is directly applied to the pre-trained text-to-image consistency model $\boldsymbol{f}_{\boldsymbol{\theta}}$ to construct a new $\boldsymbol{c}_{\text{ctrl}}$ conditioned consistency model $\boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi}^*\}}$. We assume that the learned knowledge to control image generation can be transfered to the text-to-image consistency model if the ControlNet can generalizes well enough. Empirically, we find this approach can successfully transfer high-level semantic control but often generate unrealistic images. We suspect the sub-optimal performance can be attributed to the intrinsic gap between CMs and DMs. The more in-depth analysis on the gap is provided in Section 4.3.

3.3. Consistency Training for ControlNet

Recent works (Song et al., 2023; Song & Dhariwal, 2023) figure out that except relying on the score function provided by pre-trained diffusion models, consistency models, as an independent class of generative models, can be trained from scratch using the consistency training technique. The core of the consistency training is to use an estimator of the score function, which has the form of

$$\nabla \log p_t\left(\boldsymbol{x}_t\right) = \mathbb{E}\left[\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{x})|\boldsymbol{x}_t\right] \quad (11)$$

$$= -\mathbb{E}\left[\frac{\boldsymbol{x}_t - \sqrt{\alpha_t}\boldsymbol{x}_t}{1-\alpha_t}|\boldsymbol{x}_t\right], \quad (12)$$

where $\boldsymbol{x} \sim p_{\text{data}}$ and $\boldsymbol{x}_t \sim \mathcal{N}\left(\sqrt{\alpha_t}\boldsymbol{x}, (1-\alpha_t)\boldsymbol{I}\right)$. By the Monte Carlo estimation of Equation (11), the resulting consistency training loss takes the mathematical form of

$$\mathcal{L}^N_{\text{CT}}\left(\boldsymbol{\theta}\right) = \mathbb{E}\left[\lambda\left(t_n\right) d\left(\boldsymbol{f}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_{t_{n+1}}, t_{n+1}\right), \boldsymbol{f}_{\boldsymbol{\theta}^-}\left(\boldsymbol{x}_t, t\right)\right)\right], \quad (13)$$

where the expectation is taken with respect to $\boldsymbol{x} \sim p_{\text{data}}$, $\boldsymbol{x}_t \sim \mathcal{N}\left(\sqrt{\alpha_t}\boldsymbol{x}, (1-\alpha_t)\boldsymbol{I}\right)$ and $n \sim \mathcal{U}\left([1, N-1]\right)$.

$\mathcal{U}([1, N-1])$ denotes the uniform distribution over $\{1, 2, \ldots, N-1\}$ and N is a hyper-parameter.

To train a ControlNet for the pre-trained text-to-image consistency model (denoted as $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t; \boldsymbol{c}_{\text{txt}})$ with the text prompt $\boldsymbol{c}_{\text{txt}}$), we consider to add a conditional control $\boldsymbol{c}_{\text{ctrl}}$ and define a new conditional consistency model $\boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi}\}}(\boldsymbol{x}_t, t; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}})$ by integrating the trainable ControlNet (parameterized by $\boldsymbol{\psi}$) and the original frozen CM (parameterized by $\boldsymbol{\theta}$). The resulting training loss for ControlNet is

$$\mathcal{L}_{\text{CT}}^{N}(\boldsymbol{\psi}) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{x}_t,\boldsymbol{c}_{\text{txt}},\boldsymbol{c}_{\text{ctrl}},n}\Big[\lambda(t_n) \cdot d\big(\boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi}\}}(\boldsymbol{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}}), \boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi}\}^-}(\boldsymbol{x}_{t_n}, t_n; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}})\big)\Big]. \tag{14}$$

Note that in Equation (14), only the ControlNet is trainable. We simply set $\{\boldsymbol{\theta}, \boldsymbol{\psi}\}^- = \text{stopgrad}(\{\boldsymbol{\theta}, \boldsymbol{\psi}\})$ for the teacher model since recent research (Song & Dhariwal, 2023) reveals that omitting Exponential Moving Average (EMA) is both theoretically and practically beneficial for training consistency models.

3.4. Consistency Training for A Unified Adapter.

We find that DM's ControlNet can provide high-level semantic controls to CM. However, due to the intrinsic gap between CM and DM, the control is sub-optimal, *i.e.*, it often causes unexpected deviation of image details and generate unrealistic images (as shown in Figure 1). To overcome this issue, we train a unified adapter to implement better adaption of DM's ControlNet $\{\boldsymbol{\psi_1}, \ldots \boldsymbol{\psi_K}\}$ to CM using the consistency training technique. Formally, suppose the trainable parameter of the adapter is $\boldsymbol{\Delta\psi}$, the training loss for the adapter is:

$$\mathcal{L}_{\text{CT}}^{N}(\boldsymbol{\Delta\psi}) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{x}_t,\boldsymbol{c}_{\text{txt}},\boldsymbol{c}_{\text{ctrl}},n,k}\Big[\lambda(t_n) \cdot d\big(\boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi_k},\boldsymbol{\Delta\psi}\}}(\boldsymbol{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}}), \boldsymbol{f}_{\{\boldsymbol{\theta},\boldsymbol{\psi_k},\boldsymbol{\Delta\psi}\}^-}(\boldsymbol{x}_{t_n}, t_n; \boldsymbol{c}_{\text{txt}}, \boldsymbol{c}_{\text{ctrl}})\big)\Big], \tag{15}$$

where $k \sim \mathcal{U}([1, K])$ and K denotes the number of involved conditions. $\boldsymbol{\Delta\psi} + \boldsymbol{\psi_k}$ constructs a new ControlNet, which facilitates the adaption of DM's ControlNet.

4. Experiments

4.1. Implementation Details

Text-to-Image Consistency Model. To train the foundational consistency model, we set $\boldsymbol{\theta}^- = \text{stopgrad}(\boldsymbol{\theta})$, $N = 200$, CFG $= 5.0$, and $\lambda(t_n) = 1.0$ for all $n \in \mathcal{U}([1, N-1])$. We enforce zero-terminal SNR (Lin et al., 2023) during training to align training with inference. The distance function is chosen as the ℓ_1 distance: $d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_1$. The batch size is 128 and the learning rate is $8e^{-6}$. The image resolution is 1024×1024. This training process costs about 160 A100 GPU days.

Consistency Training. To train ControlNets by consistency training, we set $\boldsymbol{\theta}^- = \text{stopgrad}(\boldsymbol{\theta})$, $N = 50$ and $\lambda(t_n) = 1.0$ for all $n \in \mathcal{U}([1, N-1])$. The distance function is chosen as the ℓ_1 distance $d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_1$. We train on a combination of public datasets, including ImageNet21K (Russakovsky et al., 2015), WebVision (Li et al., 2017), and a filter version of LAION dataset (Schuhmann et al., 2022). We elinimate duplicates, low resolution images, and images potentially contain harmful content from LAION dataset. For each ControlNet, the total training process involves 100K training steps and the batch size is 32. We utilize seven conditions in this work:

- Sketch: we use a pre-trained edge detection model (Su et al., 2021) in combination with a simplification algorithm to extract sketches;
- Canny: a canny edge detector (Canny, 1986) is employed to extract canny edges;
- Hed: a holistically-nested edge detection model (Xie & Tu, 2015) is utilized for the purpose;
- Depthmap: we employ the Midas model (Ranftl et al., 2020) for depth estimation;
- Mask: images are randomly masked. We use a 4-channel representation, where the first 3 channels correspond to the masked RGB image, while the last channel corresponds to the binary mask;
- Human Pose: a pre-trained human-pose detection model (Cao et al., 2017) is employed to generate human skeleton labels;
- Super-resolution: we use a bicubic kernel to downscale the images by a factor of 16 as the condition and hence the condition is of resolution 64×64.

4.2. Experimental Results

4.2.1. Quantitative Results

We quantitatively evaluate the proposed three methods: DM's ControNet+CM, DM's ControlNet+Adapter, CM's ControlNet+CM. For reference, we also test the performance of the classic diffusion's ControlNet (DM's ControlNet+DM). We use DDIM algorithm (Song et al., 2021a) with 50 steps and classifier-free guidance strength 5.0 to sample the diffusion model. We employ two metrics: FID is used to assess image quality and Fidelity is responsible

Table 2. Quantitative comparison of different methods. NFEs means the number of function evaluations. $\times 2$ for the diffusion model because classifier-free guidance is used. Time is recorded based on the generation of a 1024×1024 image.

Task Method	NFEs↓	Time(s)↓	Sketch2Image FID↓/Fidelity↓	Depth2Image FID↓/Fidelity↓	Mask2Image FID↓/Fidelity↓	16×SR FID↓/Fidelity↓	Average FID↓/Fidelity↓
DM's ControlNet+DM	50×2	23.6	8.40/0.106	11.48/0.177	4.37/0.085	5.01/0.121	**7.31/0.122**
DM's ControlNet+CM	1	0.2	30.71/0.083	26.08/0.193	14.67/0.431	21.32/0.237	23.19/0.231
DM's ControlNet+CM+Adapter	1	0.2	20.43/0.111	19.75/0.176	13.95/0.413	13.73/0.168	16.96/0.221
CM's ControlNet+CM	1	0.2	10.39/0.095	12.94/0.169	5.44/0.082	7.60/0.118	**9.09/0.116**
DM's ControlNet+CM	4	0.9	21.88/0.091	21.12/0.190	10.27/0.457	11.41/0.146	16.16/0.221
DM's ControlNet+CM+Adapter	4	1.0	11.91/0.113	12.83/0.175	9.16/0.452	7.21/0.146	10.27/0.221
CM's ControlNet+CM	4	0.9	9.30/0.103	9.87/0.175	4.98/0.110	6.31/0.134	**7.61/0.130**

for evaluating consistency degree with the control conditions. FID score is computed with 10000 samples. Suppose the model for extracting condition is denoted as $h(\cdot)$ and generated image $\boldsymbol{y}$, the fidelity is computed according to

$$\text{Fidelity} = \|\boldsymbol{h}(\boldsymbol{y}) - \boldsymbol{c}_{\text{ctrl}}\|_1. \tag{16}$$

The final Fidelity score is averaged over 10000 images. To compare efficiency, we report the the number of function evaluations (NFEs) and measure the time consumption on a single A100 GPU. A unified adapter is trained jointly with five conditions, *i.e.*, sketch, canny, mask, pose and SR, to learn to mitigate the gap betweens CM and DM.

Table 2 shows the FID/Fidelity results and efficiency comparison on four typical conditions. It can be found that i) DM's ControlNet+DM vs. CM's ControlNet+CM: CM's ControlNet trained with consistency training can significantly save overall NFEs ($\times$100/50) and running time ($\times$100/26) while maintaining comparable performance with diffusion models. ii) CM's ControlNet+CM vs. DM's ControlNet+CM: compared with directly utilizing diffusion's ControlNet, ControlNet obtained by consistency training can achieve better performance. iii) DM's ControlNet+CM vs. DM's ControlNet+CM+Adapter: a unified adapter can facilitate the transfer from DM to CM. It is notable that the adapter can still be effective when applied to untrained conditions (*i.e.*, Depth2image).

4.2.2. Qualitative Results

Figure 3 presents visual results of different strategies. We can find that DM's ControlNet can deliver high-level controls to CM. Nevertheless, this approach often generates unrealistic images. Consistency training directly based on CM can generate more visually-pleasing images. Therefore, it is verified that the consistency training offers a way to train the customized ControlNet for CMs. When compared to the direct transfer method, a unified adapter trained under five conditions (*i.e.*, sketch, canny, mask, pose and super-resolution) enhances the visual quality of both in-context images (*i.e.*, sketch and mask) and training-free conditions (*i.e.*, depth and hed), showing promising prospects.

Table 3. Quantitative results with different prompts.

Prompts NFEs	Paired caption FID↓/Fidelity↓	"A high-quality and professional image" FID↓/Fidelity↓
1	10.39/0.095	11.25/0.094
4	9.30/0.103	10.64/0.103

4.3. Analytic Experiments

Delving into the transferable property of DM's ControlNet. We provide more evidence to support that DM's ControlNet often suffers from low-level and realism control in comparison with the consistency training method. Given customized training and better performance, it is assumed the CM's ControlNet can provide both high- and low-level controls. We employ cosine similarity $\frac{<\boldsymbol{c}_{\text{dm}}, \boldsymbol{c}_{\text{cm}}>}{\|\boldsymbol{c}_{\text{dm}}\|_2 \cdot \|\boldsymbol{c}_{\text{cm}}\|_2}$ as as the correlation measure between two signals $(\boldsymbol{c}_{\text{dm}}, \boldsymbol{c}_{\text{cm}})$. Specifically, we compute the cosine similarity between the yielded control signals from DM's and CM's ControlNet across the network depth (see Figure 8 for the origin and the defined direction). Section 4.3a shows the correlation tendency over the depth. We can find that i) the correlation value is always positive, which means both ControlNets generally agree with each other. This partially explains that DM's ControlNet can be directly applied to CM. ii) the correlation at shallow layer is significantly larger than at deep layer (*e.g.*, 0.55 at $\text{depth} = 0.0$ vs. 0.16 at $\text{depth} = 1.0$). The shallow layer locates at the bottleneck layer of the U-Net, thereby corresponding to the high-level semantic controls and the deep layer receives the low-level control. The discrepancy of both ControlNets increases when offering low-level controls, which is consistent with our experimental observations. Section 4.3b shows amplitude of Fourier-transformed signals from CM's and DM's ControlNet. We can find that at the deep layer, the amplitude of two signals fluctuates similarly but with different scale while

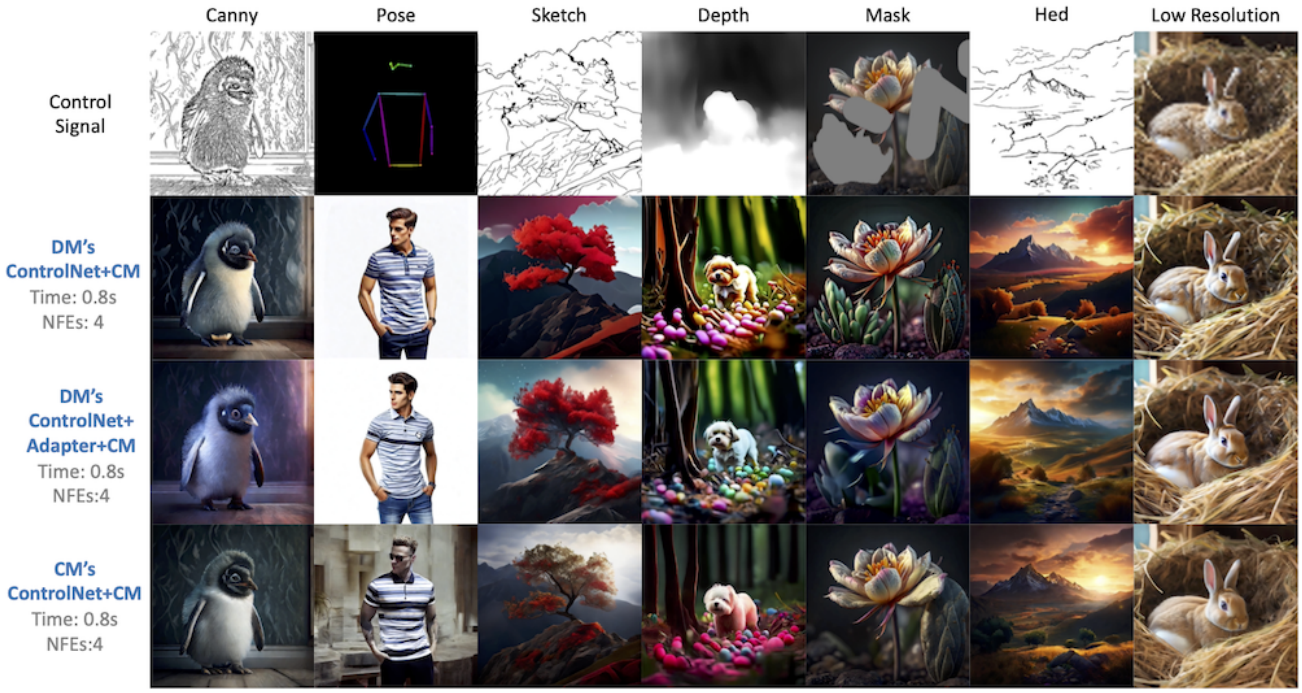


Figure 3. Visual comparison of different methods of adding controls. Image resolution: 1024×1024. NFEs: 4.

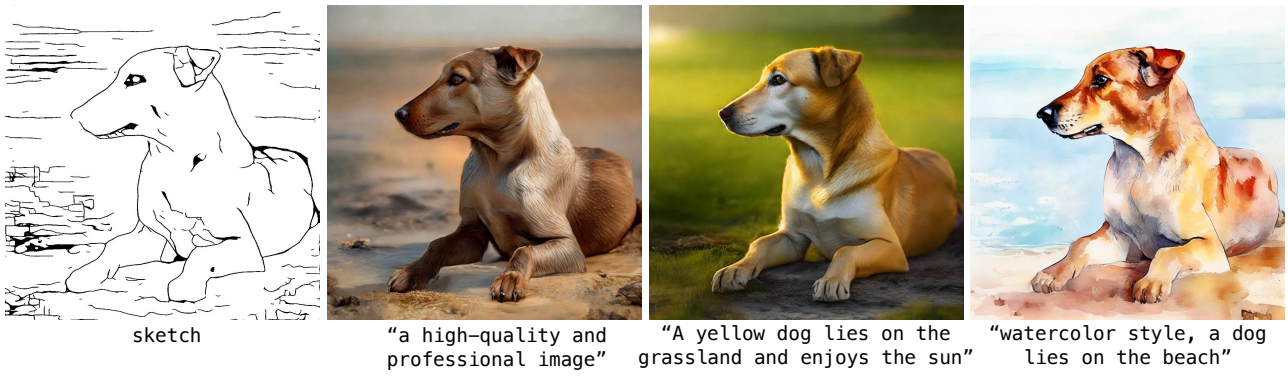


Figure 4. Visual results of CM's ControlNet with different prompts. Image resolution: 1024×1024. NFEs: 4.

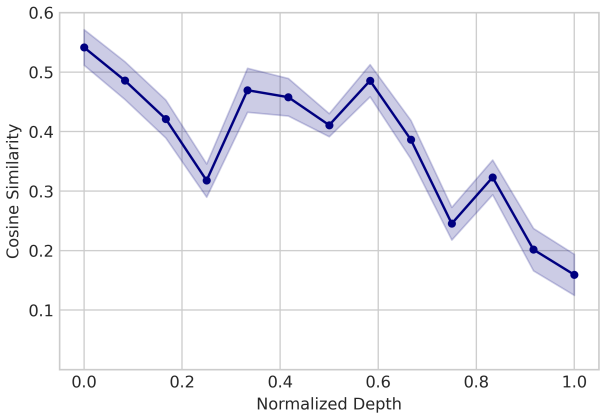
their amplitude is much closer at the shallow layer.

Robustness to language prompts. The input conditions consist of a text prompt and extra spatial condition. ControlNet offers a solution to absorbs spatial condition and here we further validate whether the efficacy heavily relies on the specific prompt. Following Zhang et al. (2023), we adopt the general prompt "A high-quality and professional image" and evaluate FID/Fidelity score. Table 3 reveals that the general prompt impairs slightly the FID score compared with the paired image caption. Figure 4 shows controllable generation results with different prompts.

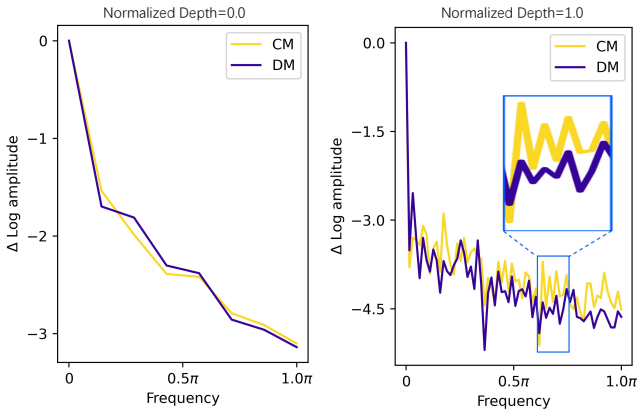
Ablation on distance function. We study the effect of the distance function. Three options are considered: mean square error function, $\mathcal{L}_1$ function and Charbonnier function (Charbonnier et al., 1994)

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\|\boldsymbol{x}' - \boldsymbol{y}\|_2^2 + \epsilon^2}, \tag{17}$$

where the constant is empirically set to $\epsilon = 10^{-3}$. We further consider the randomness of distance function and develop the "random pick" variant, which means the distance function is randomly picked from aforementioned options at each optimization step. The four-steps performance in Table 4 suggests that $\mathcal{L}_1$ surpasses other functions.



(a) Cosine similarity across network depth between CM's ControlNet and DM's ControlNet



(b) Log amplitude of Fourier-transformed control features from CM's and DM's ControlNet

Figure 5. Correlation analysis between CM's and DM's ControlNet. (a) shows the decreased correlation along the depth. (b) shows amplitude of Fourier-transformed features. These results validate that both ControlNets generally agree on high-level controls but differs on low-level controls. Please refer to Section 4.3 for details.

Table 4. Ablation study of different distance functions.

$d(\boldsymbol{x}, \boldsymbol{y})$	MSE	Charbonnier	$\mathcal{L}_1$	Random pick
Fidelity↓	0.147	0.117	0.103	0.116
FID↓	20.16	11.91	9.30	11.37

Table 5. Ablation study of the hyper-parameter N.

N	50	100	200	500	1000	Progress.
Fidelity↓	0.103	0.112	0.116	0.123	0.124	0.115
FID↓	9.30	9.68	10.31	10.25	10.37	10.21

Ablation on N. We study the effect of the hyperparameter N. We consider a range of values: $N = [50, 100, 200, 500, 1000]$ and a progressive strategy that increasing N at training stage progressively. The total training steps is 100000 for fair comparison. The four-steps performance in Table 5 suggests that $N = 50$ is the best choice.

CM'ControlNet + DM. Here, we are interested in transferring ControlNet trained on consistency model with consistency training to diffusion model. The result is shown in Figure 6, which reveals that the performance is still suboptimal: CM's ControlNet can transfer semantic-level control to DM but still fails to modulate image details.

Customizing Image with CT. We validate that the consistency training loss can also be compatible with image customization process. Specifically, we the employ consistency training loss to enable CM-based customized generation using Dreambooth (Ruiz et al., 2022) and Figure 7 shows the visual result.

5. Related Work

Real-time Generation Despite the impressive capabilities of diffusion models in generating (Ho et al., 2020; Rombach et al., 2022; Podell et al., 2023; Ramesh et al., 2021; 2022; Saharia et al., 2022) and manipulating high-resolution images (Mokady et al., 2023; Hertz et al., 2022; Roich et al., 2022), the intrinsic iterative process hinders their deployment in real-time application. We briefly review recent advancements in accelerating DMs for real-time generation. Progressive distillation (Salimans & Ho, 2022) and guidance distillation (Meng et al., 2023) introduce a method to distill knowledge from a trained deterministic diffusion sampler, which involves multiple sampling steps, into a more efficient diffusion model that requires only half the number of sampling steps. InstaFlow (Liu et al., 2023a; Liu, 2022; Liu et al., 2022) turns SD into an ultra-fast one-step model by optimizing transport cost and distillation. Consistency Models (CMs) (Song et al., 2023; Song & Dhariwal, 2023) propose a new class of generative models by enforcing self-consistency along a PF ODE trajectory. Latent Consistency Models (LCMs) (Luo et al., 2023a) and LCM LoRA (Luo et al., 2023b) extend CMs to enable large-scale text-to-image generation. There are also several approaches that utilize adversarial training to enhance the distillation process, such as UFOGen (Xu et al., 2023), CTM (Kim

Figure 6. Visual results of CM's ControlNet + DM. Image resolution: 1024×1024. NFEs: 4.

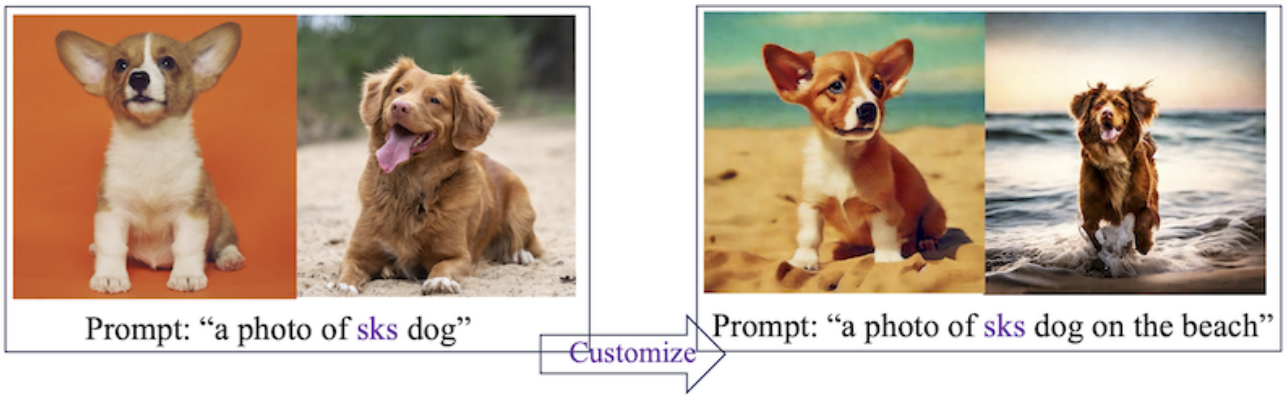


Figure 7. Visual results of customizing images using consistency training. Image resolution: 1024×1024. NFEs: 4.

et al., 2023), and ADD (Sauer et al., 2023).

Controllable Generation Recent years have witnessed significant advancements (Bhat et al., 2023; Sun et al., 2023; Hu et al., 2024) in diffusion models based controllable generation. ControlNet (Zhang et al., 2023) leverages both visual and text conditions, resulting in impressive controllable image generation. Composer (Huang et al., 2023) explores the integration of multiple distinct control signals along with textual descriptions, training the model from scratch on datasets of billions of samples. UniControl (Qin et al., 2023) and Uni-ControlNet (Zhao et al., 2023) not only enable composable control but also handle various conditions within a single model. They are also capable of achieving zero-shot learning on previously unseen tasks. There are also several customized methods, such as DreamBooth (Ruiz et al., 2022), Custom Diffusion (Kumari et al., 2022), Cones (Liu et al., 2023b;c), and Anydoor (Chen et al., 2023), that cater to user-specific controls and requirements. However, these methods require iterative steps to sample satisfactory results, which hinders their real-time application. To overcome this issue, we explore real-time controllable generation with consistency model.

6. Conclusion

In this work, we unlock the great potential of consistency models in the field of real-time controllable generation. Diffusion model's ControlNet can be directly applied to consistency model while with sub-optimal performance. We figure out that consistency training can establish the customized ControlNet from scratch for consistency models. This reveals that consistency models, as an independent family of generative models, can also be compatible with ControlNet, which extends the generative scope. Lastly, we introduce a unified adapter to facilitate adapation of diffusion model's ControlNet, resulting in promising performance.

Impact Statement

The research on real-time controllable image generation has profound implications for many fields, offering advancements that could democratize creativity, enhance interactive experiences, and streamline content creation. Nevertheless, the capability to generate images instantaneously also raises concerns regarding the propagation of misinformation through hyper-realistic fabrications and ethical considerations in the replication of human likenesses. To this issue, it is essential to establish ethical guidelines and promote transparency in its applications.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 62225207 and 62276243.

References

Bhat, S. F., Mitra, N. J., and Wonka, P. Loosecontrol: Lifting controlnet for generalized depth conditioning. *arXiv preprint arXiv:2312.03079*, 2023.

Canny, J. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Charbonnier, P., Blanc-Feraud, L., Aubert, G., and Barlaud, M. Two deterministic half-quadratic regularization algorithms for computed imaging. In *IEEE International Conference on Image Processing*, volume 2, pp. 168–172 vol.2, 1994.

Chen, X., Huang, L., Liu, Y., Shen, Y., Zhao, D., and Zhao, H. Anydoor: Zero-shot object-level image customization. *arXiv preprint arXiv:2307.09481*, 2023.

Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., and Cohen-Or, D. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Hu, H., Chan, K. C., Su, Y.-C., Chen, W., Li, Y., Sohn, K., Zhao, Y., Ben, X., Gong, B., Cohen, W., et al. Instruct-imagen: Image generation with multi-modal instruction. *arXiv preprint arXiv:2401.01952*, 2024.

Huang, L., Chen, D., Liu, Y., Shen, Y., Zhao, D., and Zhou, J. Composer: Creative and controllable image synthesis with composable conditions. *arXiv preprint arXiv:2302.09778*, 2023.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35: 26565–26577, 2022.

Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., and Ermon, S. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279*, 2023.

Kumari, N., Zhang, B., Zhang, R., Shechtman, E., and Zhu, J.-Y. Multi-concept customization of text-to-image diffusion. *arXiv preprint arXiv:2212.04488*, 2022.

Li, W., Wang, L., Li, W., Agustsson, E., and Van Gool, L. Webvision database: Visual learning and understanding from web data. *arXiv preprint arXiv:1708.02862*, 2017.

Lin, S., Liu, B., Li, J., and Yang, X. Common diffusion noise schedules and sample steps are flawed. *arXiv preprint arXiv:2305.08891*, 2023.

Liu, Q. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*, 2022.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Liu, X., Zhang, X., Ma, J., Peng, J., and Liu, Q. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. *arXiv preprint arXiv:2309.06380*, 2023a.

Liu, Z., Feng, R., Zhu, K., Zhang, Y., Zheng, K., Liu, Y., Zhao, D., Zhou, J., and Cao, Y. Cones: Concept neurons in diffusion models for customized generation. *arXiv preprint arXiv:2303.05125*, 2023b.

Liu, Z., Zhang, Y., Shen, Y., Zheng, K., Zhu, K., Feng, R., Liu, Y., Zhao, D., Zhou, J., and Cao, Y. Cones 2: Customizable image synthesis with multiple subjects. *arXiv preprint arXiv:2305.19327*, 2023c.

Luo, S., Tan, Y., Huang, L., Li, J., and Zhao, H. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023a.

Luo, S., Tan, Y., Patil, S., Gu, D., von Platen, P., Passos, A., Huang, L., Li, J., and Zhao, H. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023b.

Meng, C., Rombach, R., Gao, R., Kingma, D., Ermon, S., Ho, J., and Salimans, T. On distillation of guided diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Mokady, R., Hertz, A., Aberman, K., Pritch, Y., and Cohen-Or, D. Null-text inversion for editing real images using guided diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6038–6047, 2023.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Qin, C., Zhang, S., Yu, N., Feng, Y., Yang, X., Zhou, Y., Wang, H., Niebles, J. C., Xiong, C., Savarese, S., et al. Unicontrol: A unified diffusion model for controllable visual generation in the wild. *arXiv preprint arXiv:2305.11147*, 2023.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., and Koltun, V. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Roich, D., Mokady, R., Bermano, A. H., and Cohen-Or, D. Pivotal tuning for latent-based editing of real images. *ACM Transactions on graphics*, 42(1):1–13, 2022.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022.

Salimans, T. and Ho, J. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Sauer, A., Lorenz, D., Blattmann, A., and Rombach, R. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023.

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 2022.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Song, Y. and Dhariwal, P. Improved techniques for training consistency models. *arXiv preprint arXiv:2310.14189*, 2023.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Song, Y., Dhariwal, P., Chen, M., and Sutskever, I. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Su, Z., Liu, W., Yu, Z., Hu, D., Liao, Q., Tian, Q., Pietikäinen, M., and Liu, L. Pixel difference networks for efficient edge detection. In *International Conference on Computer Vision*, 2021.

Sun, Q., Cui, Y., Zhang, X., Zhang, F., Yu, Q., Luo, Z., Wang, Y., Rao, Y., Liu, J., Huang, T., et al. Generative multimodal models are in-context learners. *arXiv preprint arXiv:2312.13286*, 2023.

Xie, S. and Tu, Z. Holistically-nested edge detection. In *International Conference on Computer Vision*, 2015.

Xu, Y., Zhao, Y., Xiao, Z., and Hou, T. Ufogen: You forward once large scale text-to-image generation via diffusion gans. *arXiv preprint arXiv:2311.09257*, 2023.

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *International Conference on Computer Vision*, 2023.

Zhao, S., Chen, D., Chen, Y.-C., Bao, J., Hao, S., Yuan, L., and Wong, K.-Y. K. Uni-controlnet: All-in-one control to text-to-image diffusion models. *arXiv preprint arXiv:2305.16322*, 2023.

A. Proof to Equation (8)

Proof. Given the PF ODE trajectory $\{\boldsymbol{x}_t\}_{t\in[\epsilon,T]}$, the consistency model $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ aims to learn to project any point $(\boldsymbol{x}_t, t)$ to $\boldsymbol{x}_\epsilon$. That is, $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ should be as close as possible to $\boldsymbol{x}_\epsilon$:

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \to \boldsymbol{x}_\epsilon \tag{18}$$

$$\Rightarrow \quad c_{\text{skip}}(t)\,\boldsymbol{x}_t + c_{\text{out}}(t)\,F_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \to \boldsymbol{x}_\epsilon \tag{19}$$

$$\Rightarrow \quad F_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \to \frac{\boldsymbol{x}_\epsilon - c_{\text{skip}}(t)\,\boldsymbol{x}_t}{c_{\text{out}}(t)}. \tag{20}$$

Suppose the variance preserving noise schedule $\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_\epsilon + \sqrt{1-\alpha_t^2}\boldsymbol{\epsilon}$, we require that the learning target has unit variance:

$$\text{Var}\left[\frac{\boldsymbol{x}_\epsilon - c_{\text{skip}}(t)\,\boldsymbol{x}_t}{c_{\text{out}}(t)}\right] = 1 \tag{21}$$

$$\Rightarrow \quad \frac{1}{c_{\text{out}}^2(t)}\text{Var}\left[\boldsymbol{x}_\epsilon - c_{\text{skip}}(t)\left(\alpha_t \boldsymbol{x}_\epsilon + \sqrt{1-\alpha_t^2}\boldsymbol{\epsilon}\right)\right] = 1 \tag{22}$$

$$\Rightarrow \quad \frac{1}{c_{\text{out}}^2(t)}\text{Var}\left[\left(1-\alpha_t c_{\text{skip}}(t)\right)\boldsymbol{x}_\epsilon - c_{\text{skip}}(t)\sqrt{1-\alpha_t^2}\boldsymbol{\epsilon}\right] = 1. \tag{23}$$

Suppose that ϵ is small enough, therefore $\text{Var}[\boldsymbol{x}_\epsilon] \approx \sigma_{\text{data}}^2$ where σ_{data}^2 is the variance of data distribution p_{data}. Since data distribution p_{data} and the injected Gaussian noise are independent, we have

$$\frac{1}{c_{\text{out}}^2(t)}\left[\left(1-\alpha_t c_{\text{skip}}(t)\right)^2\sigma_{\text{data}}^2 + c_{\text{skip}}^2(t)\left(1-\alpha_t^2\right)\right] = 1 \tag{24}$$

$$\Rightarrow \quad c_{\text{out}}^2(t) = \left(1-\alpha_t c_{\text{skip}}(t)\right)^2\sigma_{\text{data}}^2 + c_{\text{skip}}^2(t)\left(1-\alpha_t^2\right) \tag{25}$$

c_{out} determines the amplifying factor of $F_{\boldsymbol{\theta}}$. Following (Karras et al., 2022) we require that $c_{\text{skip}}(t)$ and $c_{\text{out}}(t)$ should amplify errors in $F_{\boldsymbol{\theta}}$ as little as possible. Hence

$$\frac{\mathrm{d}c_{\text{out}}^2(t)}{\mathrm{d}c_{\text{skip}}(t)} = 0 \tag{26}$$

$$\Rightarrow \quad c_{\text{skip}}(t)\left(1-\alpha_t^2\right) - \alpha_t\left(1-\alpha_t c_{\text{skip}}(t)\right)\sigma_{\text{data}}^2 = 0 \tag{27}$$

$$\Rightarrow \quad c_{\text{skip}}(t) = \frac{\alpha_t \sigma_{\text{data}}^2}{1-\alpha_t^2+\alpha_t^2\sigma_{\text{data}}^2}. \tag{28}$$

Substituting $c_{\text{skip}}(t)$ in Equation (28) back to Equation (25), we have

$$c_{\text{out}}(t) = \frac{\sqrt{1-\alpha_t^2}\sigma_{\text{data}}}{\sqrt{1-\alpha_t^2+\alpha_t^2\sigma_{\text{data}}^2}}. \tag{29}$$

□

B. Architecture of ControlNet

Figure 8 presents the architecture of ControlNet proposed in (Zhang et al., 2023). The trainable blue blocks and the white zero convolution layers are added to build a ControlNet. We mark the origin and positive direction of depth in Figure 8.

C. Architecture of Adapter

Figure 9 presents the architecture of the unified adapter. The number of trainable residual blocks is consistent with the output length of the corresponding ControlNet, and each block consists of two convolution modules and one skip connection.

D. Performance of the Re-trained Consistency Model

Real-time CM Generation. To comprehensively evaluate the quality of images generated under the aforementioned conditions, Figure 15 presents the effects of our re-trained text-to-image CM model during four-step inference.

E. More Visual Results

Applying DM's ControlNet without Modification. Figure 10 presents visual results of applying DM's ControlNet to CM. We can find that DM's ControlNet can deliver high-level controls to CM. Nevertheless, this approach often generates unrealistic images, *e.g.*, Sketch in Figure 10. Moreover, DM's ControlNet of masked images causes obvious changes outsides the masked region (Mask inpainting in Figure 10). This sub-optimal control may explained that there exists the gap between CM and DM, which further causes imperfect adaptation of DM's ControlNet to CM.

Consistency Training for CM's ControlNet. For fair comparison, Figure 11 shows corresponding visual results of consistency training for ControlNet. We can find that consistency training directly based on CM can generate more realistic images. Therefore, we can conclude that consistency training offers a way to train the customized ControlNet for CMs. More generative results can be found in Figure 12.

Transferring DM's ControlNet with a Unified Adapter. When compared to direct transfer method, a unified adapter trained under five conditions (*i.e.*, sketch, canny, mask, pose and super-resolution) enhances the visual quality of both in-context images (*i.e.*, sketch and mask conditions in Figure 13) and training-free conditions (*i.e.*, depthmap and hed conditions in Figure 14), showing promising prospects.

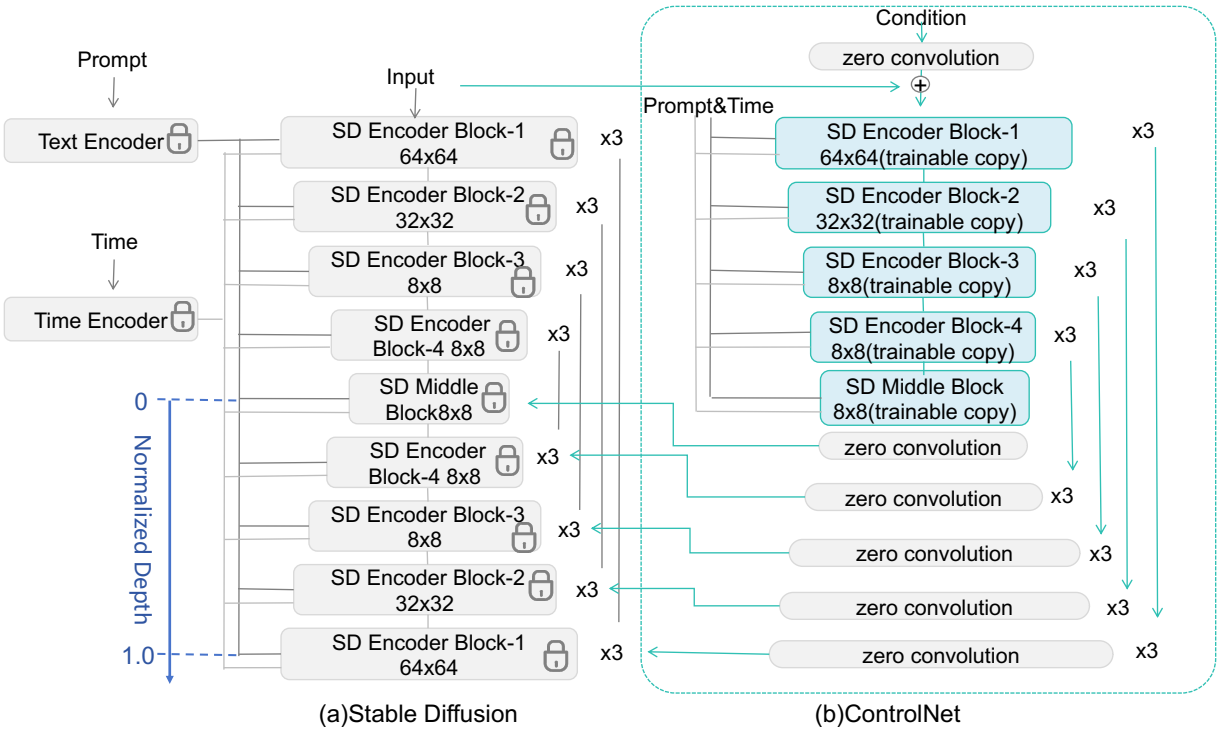


Figure 8. Architecture of ControlNet.

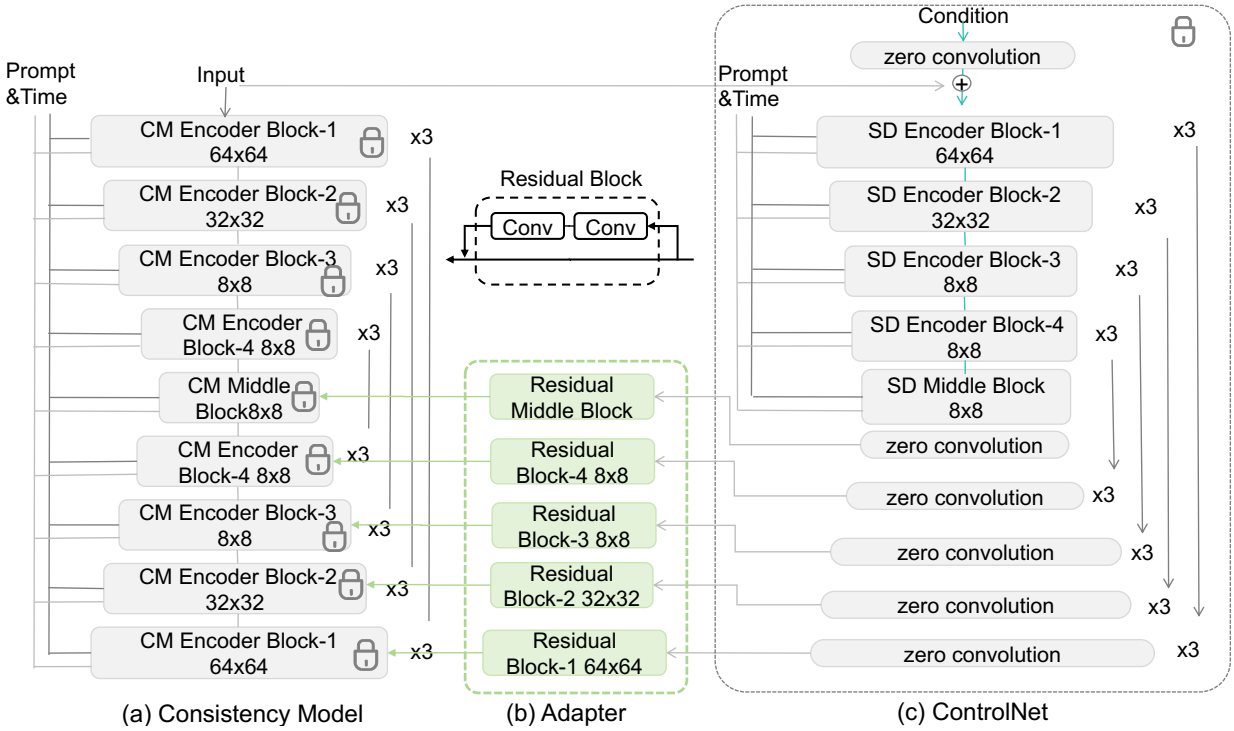


Figure 9. Architecture of Adapter.

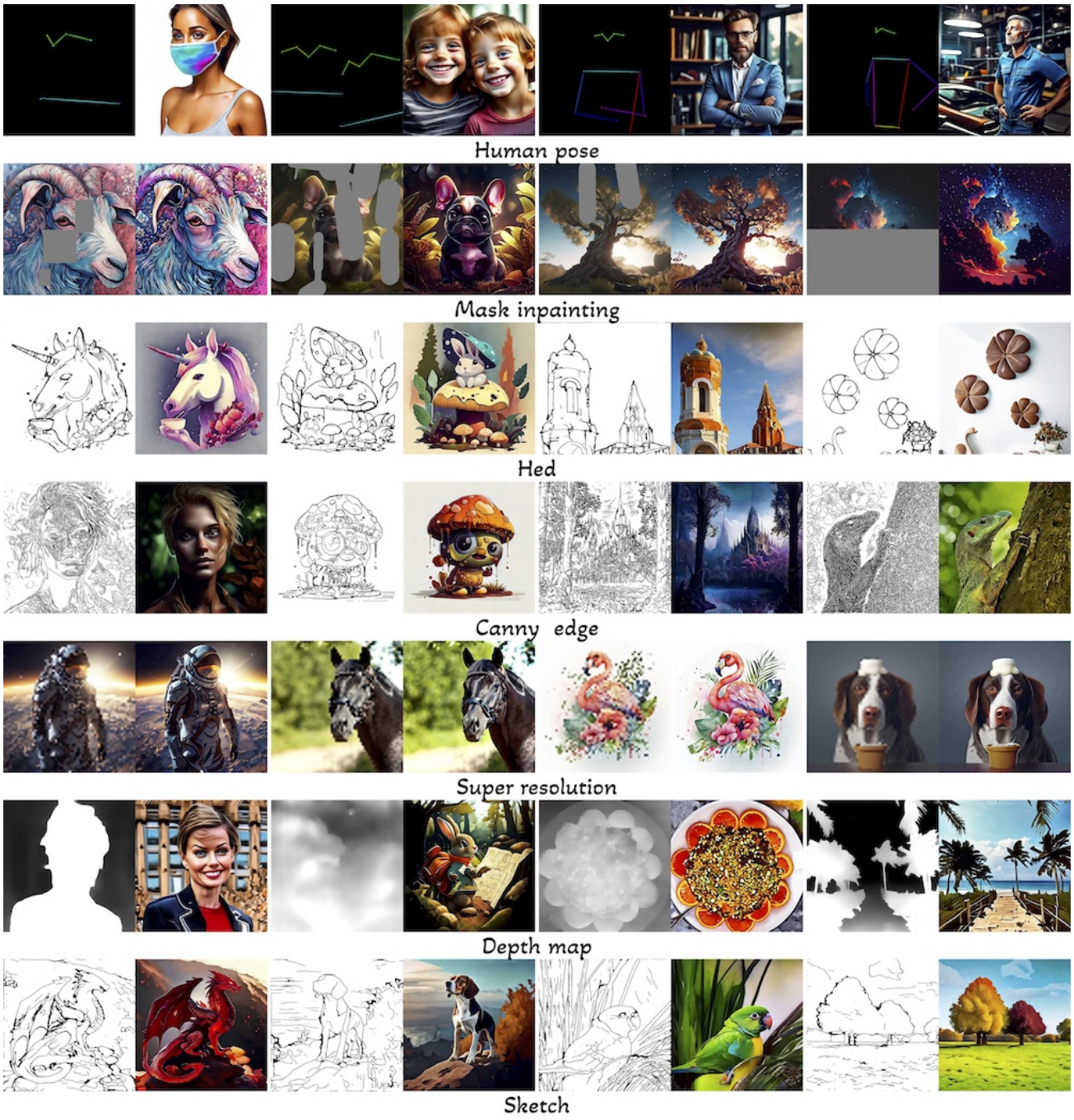


Figure 10. Images sampled by applying DM's ControlNet to CM at 1024x1024 resolution. NFEs=4.

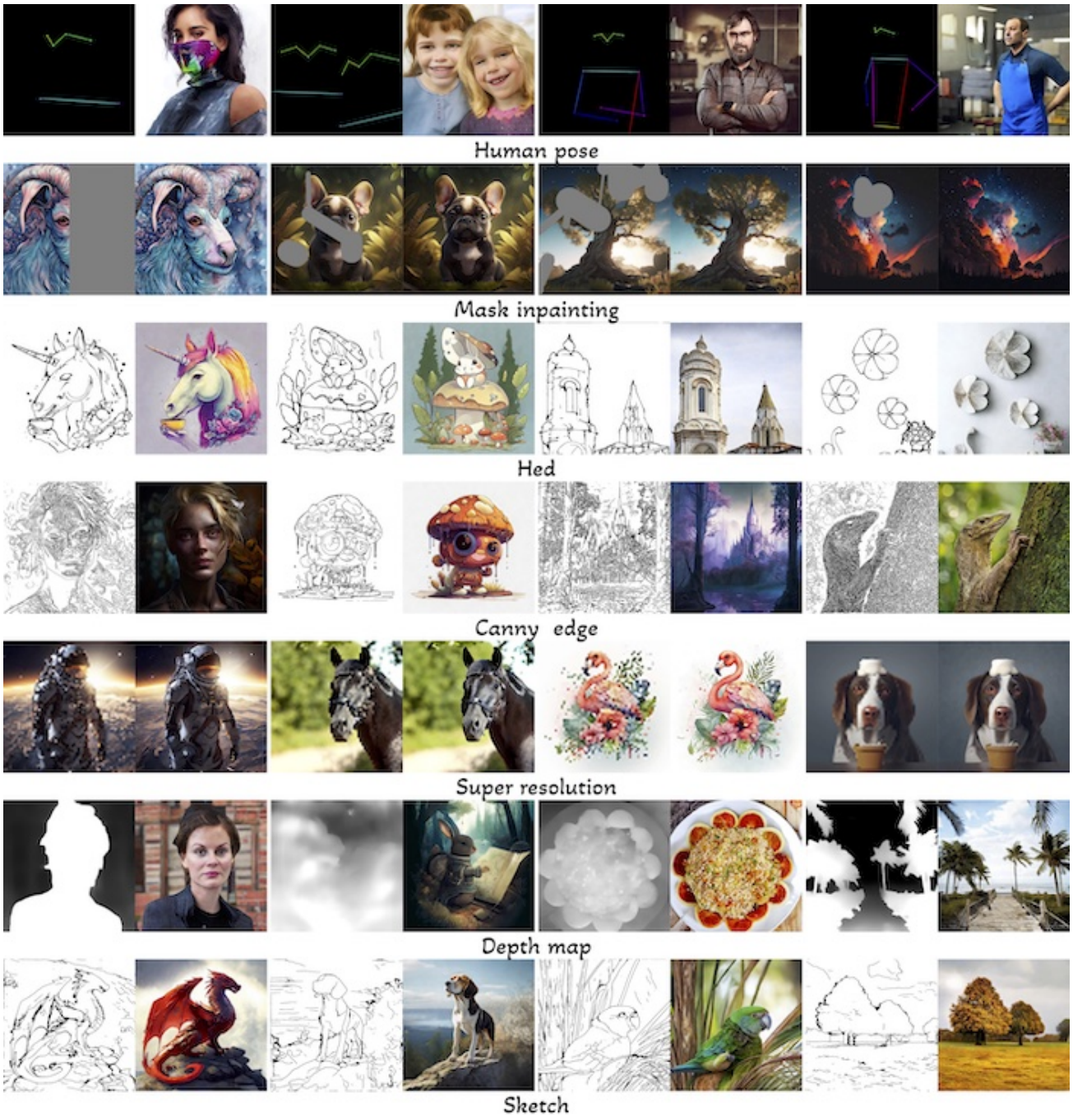


Figure 11. Visual results of consistency training at 1024x1024 resolution. The conditions are the same with those in Figure 10. It can be observed that CM's ControlNet using consistency training can generate more visually pleasing images compared to DM's ControlNet. NFEs=4.

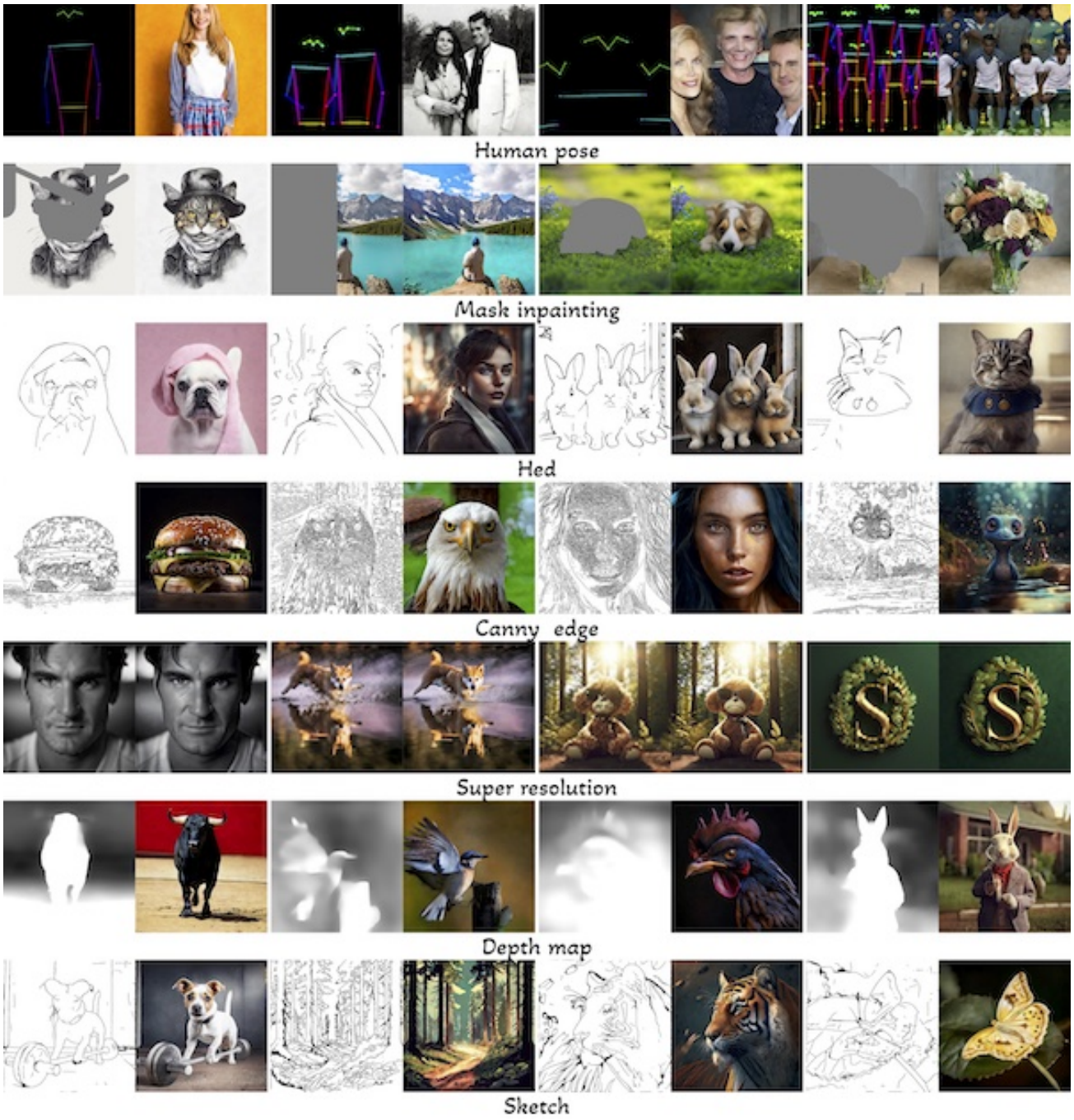


Figure 12. More visual results of CM's ControlNet using consistency training strategy at 1024x1024 resolution. NFEs=4.

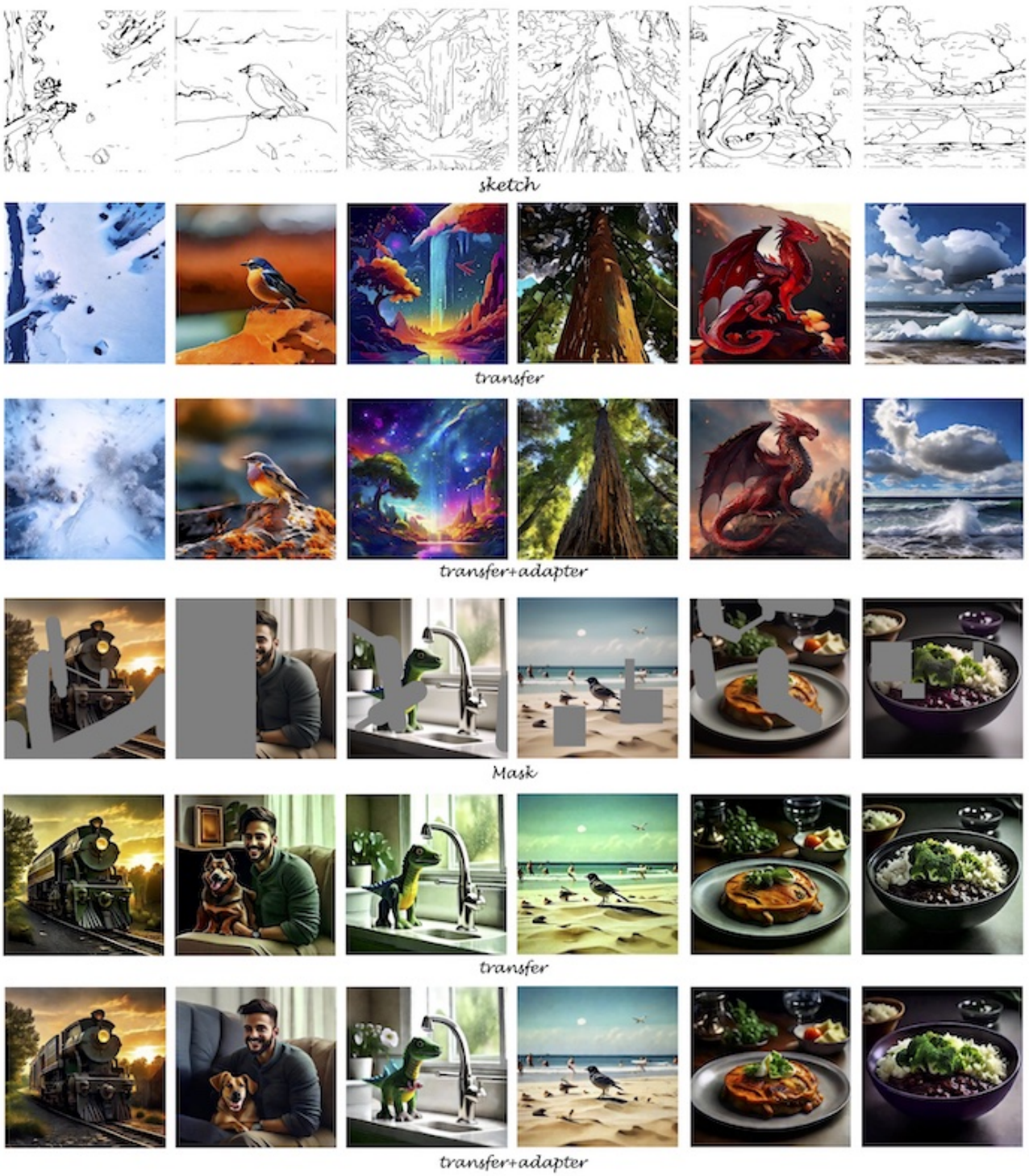


Figure 13. Visual results of DM's ControlNet without/with a unified adapter at 1024x1024 resolution. NFEs=4. "transfer" means directly transferring DM's ControlNet to CM (DM's ControlNet + CM). "transfer+adapter" means directly transferring DM's ControlNet to CM with a unified adapter (DM's ControlNet + CM+adapter).

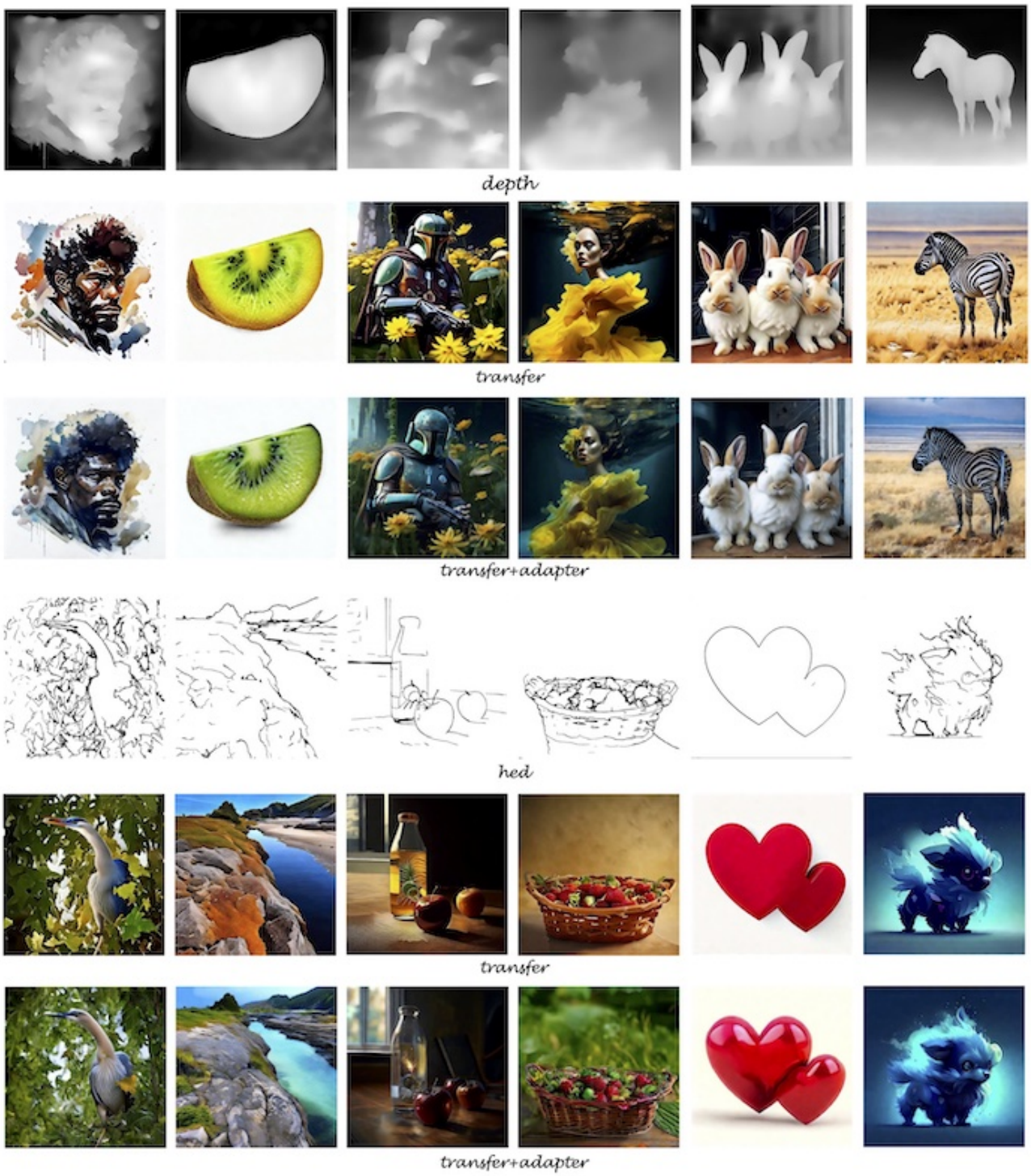


Figure 14. Visual results of DM's ControlNet without/with a unified adapter on training-free conditions at 1024x1024 resolution. NFEs=4. "transfer" means directly transferring DM's ControlNet to CM (DM's ControlNet + CM). "transfer+adapter" means directly transferring DM's ControlNet to CM with a unified adapter (DM's ControlNet + CM+adapter).

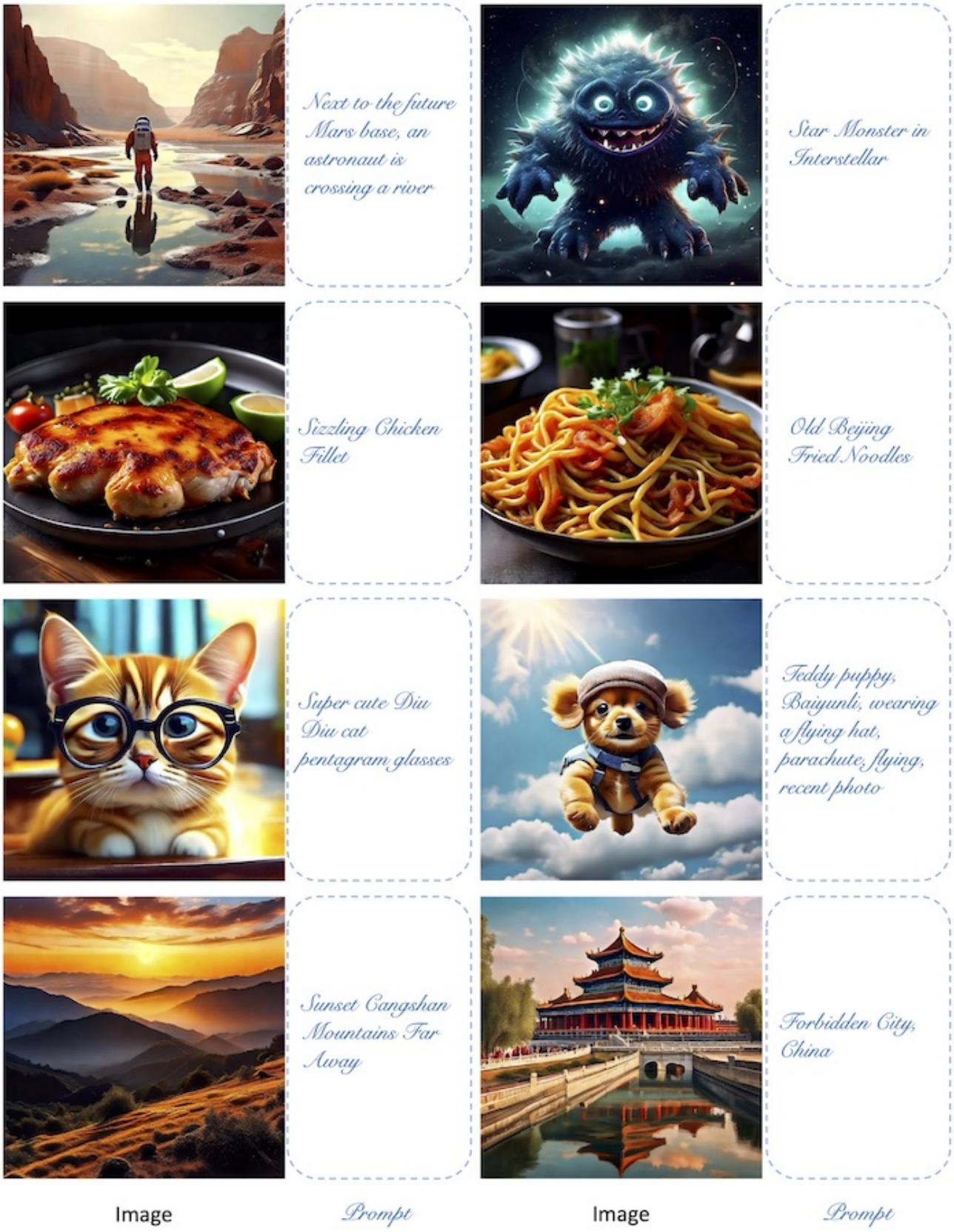


Figure 15. Images generated using our re-trained Text-to-Image CM with 4-step inference at 1024x1024 resolution.