

ControlNeXt: Powerful and Efficient Control for Image and Video Generation

Bohao Peng¹ Jian Wang¹ Yuechen Zhang¹ Wenbo Li¹ Ming-Chang Yang¹ Jiaya Jia^{1,2}
¹CUHK ²SmartMore

<https://github.com/dvlab-research/ControlNeXt>

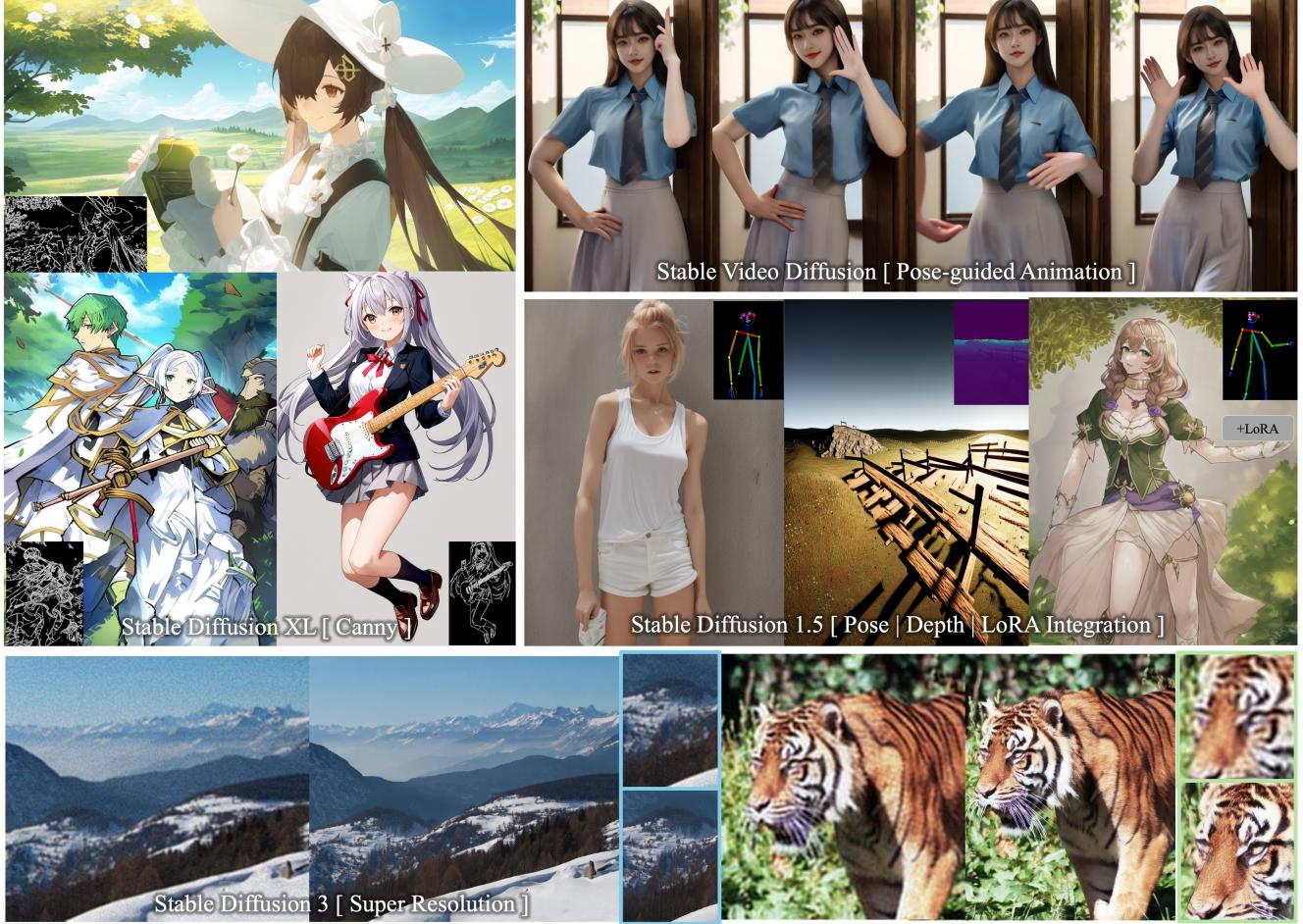


Figure 1. ControlNeXt is a powerful and efficient method for controllable generation, emphasizing improved efficiency and generality. We demonstrate its applicability across diverse tasks and mainstream architectures. More results are provided in the supplementary materials. For more examples, please refer to our project page: <https://pbihao.github.io/projects/controlnext/index.html>

Abstract

Diffusion models have achieved significant success in image and video generation, with conditional architectures such as ControlNet, Adapters, and ReferenceNet advancing spatial conditioning capabilities. However, existing controllable generation systems face limitations due to high computational requirements, slow convergence, and training instability, especially in resource-intensive video generation tasks. While researchers provide some task-specific solu-

tions, they often lack flexibility and generality. To address these challenges, we introduce ControlNeXt—a powerful and efficient method for controllable image and video generation. ControlNeXt employs a lightweight architecture that integrates conditioning seamlessly, reducing learnable parameters by up to 90% compared to other approaches. Additionally, we propose Cross Normalization (CN), a stable and faster alternative to “zero-convolution” that improves training convergence. Extensive experiments across multiple models highlight ControlNeXt’s generality and effectiveness in both image and video generation tasks.

1. Introduction

Diffusion models generate complex, structured data by progressively refining a simple initial distribution, yielding realistic and high-quality results in image and video synthesis [4, 10, 14, 33, 51, 66]. Despite their success, these models often struggle with controllable generation, as achieving specific outcomes typically involves labor-intensive tuning of prompts and seeds. To address this, recent approaches [5, 25, 70] incorporate auxiliary guidance signals, such as depth, pose skeletons, and edge maps, enabling more precise and targeted generation.

Popular controllable generation methods usually incorporate parallel branches or adapters to integrate control signals, as seen in ControlNet [59, 70], T2I-Adapter [41], and ReferenceNet [21]. These architectures process auxiliary controls in parallel while the main base model remains frozen. However, relying solely on the auxiliary components to capture controls always needs numerous parameters and introduces challenges, including increased computational demands, slower convergence, training instability, and limited controllability, as discussed in Secs 4.1 and 4.3. These issues are especially pronounced in resource-intensive video generation tasks. While T2I-Adapter [41] offers an efficient fine-tuning approach optimized for image generation, prioritizing efficiency often compromises controllability, rendering it less suitable for video generation and fidelity-oriented low-level tasks (details provided in the supplementary). Consequently, there is a pressing need for a controllable generation method that balances efficiency with general control capabilities.

This paper presents ControlNeXt, an efficient and general method for controllable generation, highlighting its enhanced performance across various tasks and backbone architectures (see Fig. 1). Previous methods have demonstrated that control can be applied to pre-trained models [4, 46, 51, 74] by fine-tuning on small-scale datasets, suggesting that capturing control signals is not inherently difficult. Therefore, we argue that the base model itself is sufficiently powerful to be fine-tuned directly for controllability, without the need for additional auxiliary control components. This approach not only improves efficiency but also enhances the model’s adaptability to complex tasks. To achieve this, we only use a lightweight convolution module to inject control signals, enabling the pre-trained model itself to learn controllable generation through selective fine-tuning. Specifically, we freeze most of the base model’s parameters and selectively train a smaller subset, mitigating catastrophic forgetting [7, 15, 20, 39] while significantly reducing training costs with minimal latency increase. This is especially crucial for complex tasks, such as video generation, where parameter-efficient fine-tuning (PEFT) methods like adapters and LoRA may fall short.

Furthermore, we introduce Cross Normalization as an

alternative to Zero Convolution [70], which serves as a “bridge layer” connecting the control branch to the base model. Zero Convolution, which initializes weights to zero, allows control signals to gradually influence the model during training. This approach is commonly used when fine-tuning pre-trained generation models, as introducing new components or parameters directly can lead to training collapse [26, 64, 72]. However, it also results in slow convergence as the learnable parameters initially struggle to receive the correct gradients. In this paper, we argue that training collapse primarily arises from the distributional mismatch between control guidance features and the intermediate features of the pre-trained model. This distributional dissimilarity makes the two sets of parameters incompatible. To address this, ControlNeXt introduces Cross Normalization, which aligns the data distributions, leading to more efficient and stable training and mitigating the “sudden convergence” problem observed in [70].

We conduct a series of experiments on various generative backbones for image and video synthesis [4, 46, 51, 56], demonstrating the generality and broad compatibility of ControlNeXt. Its lightweight design makes it a versatile, plug-and-play module that seamlessly integrates with other methods. Additionally, ControlNeXt accommodates LoRA weights [20, 52], allowing for style modification without requiring further training. Our key contributions are summarized as follows:

- We introduce ControlNeXt, a powerful and efficient method for controllable generation that strikes a balance between performance and general control capabilities.
- We propose Cross Normalization for fine-tuning large pre-trained models, enabling fast and stable convergence during training.
- ControlNeXt serves as a lightweight, plug-and-play module that integrates seamlessly with LoRA weights to modify generation styles without additional training.

2. Related Work

Image and video diffusion models. Diffusion probability models [10, 18, 54] are advanced generative models that restore original data from pure Gaussian noise by learning the distribution of noisy data at various levels of noise. With their powerful capability to fit complex data distributions, diffusion models have excelled in several domains, including image and video generation. In the domain of image synthesis, diffusion models have demonstrably outperformed traditional Generative Adversarial Networks (GANs) in both image fidelity and diversity [10]. As research in this field advances, diffusion models continue to push the boundaries of video generation, yielding unprecedented improvements in quality and temporal consistency. The predominant neural network architectures employed in diffusion models include UNet [10, 18, 42, 43]

and DiT [44], with emerging alternatives such as U-ViT [3].

In recent years, latent diffusion models have incorporated variational autoencoders (VAEs) to transfer the diffusion process to latent space, significantly accelerating the model’s training and inference efficiency. Leading image generation models like Stable Diffusion [12, 46, 51, 56] have been widely adopted, used, and modified by the community. This success is attributed to streamlined and efficient model architecture designs, including sampling methods [37, 40, 55], the network structures of diffusion models [45], and various additional and extended components.

Controllable generation. Most recent models are guided by textual information as conditions [9, 48, 49] to extract textual features that guide the generated content. There are two main methods for introducing controllable conditions into image or video generation models: (i) training a large diffusion model from scratch to achieve controllability under multiple conditions [22], (ii) fine-tuning an adapter on a pretrained large model while keeping the original model parameters frozen [41, 70]. Recent studies have attempted to control the outcomes of generative models by integrating additional neural networks into the foundation of diffusion models [65, 73]. ControlNet guides image generation to align with control information by duplicating specific layers from pre-trained large models [59, 70], but this approach introduces substantial parameters and latency. In contrast, the T2I-Adapter [41] employs an adapter for low-cost control, though it minimally affects original model, resulting in weaker control that limit its use for complex tasks.

Distribution alignment in diffusion models. Recent studies have highlighted the importance of distribution alignment [35]. Zhang et al.[71] demonstrated that perturbing the initial noise distribution can mitigate generation issues by altering learned data distributions. In image-to-image translation and inpainting, techniques like[36] achieve improved results by aligning input noise with reference image distributions at intermediate stages. While Nie et al.[8] enhanced sample quality through posterior distribution alignment. These works collectively underscore the critical role of distribution matching in enhancing diffusion model performance and stability.

3. Method

In this section, we provide a detailed technical overview of ControlNeXt. We first introduce the necessary preliminaries for controllable generation in Sec. 3.1. In Sec. 3.2, we delve into the analysis of the architecture design and prune it in order to make a concise and straightforward structure. Next, we introduce *Cross Normalization* in Sec 3.3, which is designed for the efficient fine-tuning of large pre-trained models with additional components.

3.1. Preliminaries

Diffusion model (DM) is a type of generative model that generates data by reversing a gradual noise-adding process, transforming random noise into coherent data samples. The model’s prediction for x_t at time step t depends only on x_{t+1} and t :

$$p_\theta(x_t|x_{t+1}) = \mathcal{N}(x_t; \tilde{\mu}_t, \tilde{\beta}_t I), \quad (1)$$

where θ represents the pre-trained model, $\tilde{\mu}_t$ is the model’s predicted target, and the variance $\tilde{\beta}_t$ is computed from the posterior of forward diffusion:

$$\tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t. \quad (2)$$

The loss function of diffusion models is the MSE loss function on the noise prediction $\hat{\epsilon}_\theta(x_t, t, c_t)$:

$$\mathcal{L} = w \cdot \mathbb{E}_{t, c_t, \epsilon \sim \mathcal{N}(0, 1)} [\|\epsilon - \hat{\epsilon}(x_t, t, c_t)\|^2], \quad (3)$$

where c_t represents text prompts, and w denotes the weight of the loss function. ControlNet [70] introduces controllable generation by integrating conditional control c_f . It calculates the loss function as:

$$\mathcal{L} = w \cdot \mathbb{E}_{t, c_t, c_f, \epsilon \sim \mathcal{N}(0, 1)} [\|\epsilon - \hat{\epsilon}(x_t, t, c_t, c_f)\|^2]. \quad (4)$$

3.2. Architecture

Motivation. ControlNet [70] introduces a control branch to facilitate controllable generation, keeping the base model frozen to maintain its inherent generative quality. This branch, initialized as a replica of the original downsampling blocks, operates in parallel with the base model and employs a *zero convolution* to integrate controls (more details provided in Sec. 3.3). Specifically:

$$\mathbf{y}_c = \mathcal{F}_m(\mathbf{x}) + \mathcal{Z}(\mathcal{F}_{cn}(\mathbf{x}, \mathbf{c}; \Theta_{cn}); \Theta_z), \quad (5)$$

where $\mathcal{F}(\cdot; \Theta)$ denotes a neural model with parameters Θ , $\mathcal{Z}(\cdot; \Theta_z)$ indicates the *zero convolution* layer. \mathbf{x} , $\mathbf{y}_c \in \mathbb{R}^{h \times w \times c}$ and \mathbf{c} are the 2D feature maps and conditional controls, respectively.

Incorporating control capabilities in ControlNet entails significant computational costs. The additional branch increases considerable latency with extensive learnable parameters, particularly affecting video generation. The T2I-adapter [41] improves efficiency by replacing the control branch with an adapter. But this efficiency comes at the cost of reduced controllability and limits its effectiveness for complex tasks such as controllable video generation or low-level visual tasks. Moreover, freezing the base model and optimizing only the auxiliary modules limits overall model performance and slows convergence. To achieve general

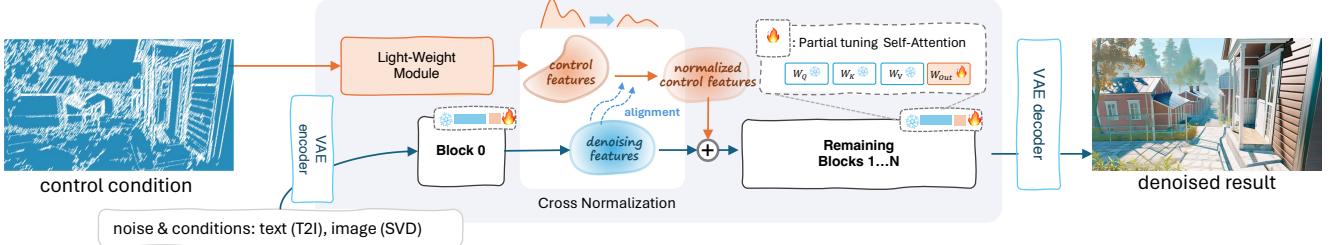


Figure 2. Training pipeline of ControlNeXt. We explore a powerful framework that achieves efficient controllable generation.

and efficient control, we propose allowing the pre-trained model to learn the control function directly.

Architecture. It is important to note that the pretrained model is typically trained on a large-scale dataset, such as LAION-5B [53], whereas fine-tuning is always conducted on a much smaller dataset, often thousands of times smaller. Based on this, we assert that the pre-trained model is sufficiently powerful and general to capture controllability directly, without the need for heavy auxiliary components.

We first eliminate the auxiliary components specifically designed to acquire control capabilities, such as ControlNet and adapters. To integrate the controls, we employ a compact convolution module only composed of multiple convolution blocks [16]. Notably, this module is significant small and solely for extracting and aligning controls. For controllability, we propose directly fine-tuning a subset of the base model to enable it to capture guidance information. During training, we freeze most pretrained modules and optimize a small subset of parameters, such as the linear layers in the attention blocks. More details about the selected parts are provided in the supplementary material. Freezing most parameters also prevents catastrophic forgetting while maintaining training efficiency. And directly fine-tuning a subset of the base model is especially crucial for complex tasks like video generation, where PEFT methods such as LoRA and adapters [20, 34, 58] may fall short. This approach enhances both effectiveness and efficiency, allowing adaptive adjustment of the learnable parameter scale to suit different tasks. Mathematically,

$$\mathbf{y}_c = \mathcal{F}_m(\mathbf{x}, \mathcal{F}_c(\mathbf{c}; \Theta_c); \Theta'_m), \quad (6)$$

where $\Theta'_m \subseteq \Theta_m$ represents a trainable subset of the pretrained parameters, and \mathcal{F}_c is the lightweight convolution module. A more intuitive presentation is shown in Fig. 2.

Regarding control injection, we aim to integrate control information at the earliest stage, allowing the base model to perceive the guiding information from the outset. However, we found that directly adding the controls to the inputs results in training collapse, possibly due to the confusion and overlap between the controls and denoising features. Thus, we inject the controls after the first block, incorporating a residual connection to preserve the integrity of the main branch’s identity transformation. Further details will be

provided in the supplementary. Controls are directly added to the denoising features after Cross Normalization introduced in Sec. 3.3, which further enhances training stability. Based on the above, ControlNeXt functions as a plug-and-play module, designed with a lightweight convolution and learnable parameters, represented as:

$$\mathcal{M}_c = \{\mathcal{F}_c(\cdot; \Theta_c), \Theta'_d\} \quad (7)$$

where $\Theta'_d \subseteq \Theta_d$, and $\Theta_c \ll \Theta_d$.

3.3. Cross Normalization

Motivation. A key challenge in continual training of pre-trained large models is how to appropriately introduce additional parameters and modules. Since directly combining new modules often leads to training collapse, recent works widely adopt zero initialization [70, 72], initializing the bridge layer that connects the based model and the added module to zeros. It ensures that newly introduced modules have no impact at the start of training, facilitating a stable warm-up phase. However, zero initialization also slows convergence and increases training challenges by preventing the modules from receiving accurate gradients at the start. This results in a phenomenon known as “sudden convergence” in controllable generation, where the model doesn’t gradually learn the conditions but abruptly starts to follow them after an extended training [70].

Cross normalization. The unaligned and incompatible data distribution among various features leads to training collapse and slow convergence [23, 35, 71]. After training on the large-scale data, the pretrained generation model typically exhibits stable feature and distributions, characterized by consistent mean and standard deviation. However, the newly introduced neural modules are typically only initialized using random methods [17, 27, 28], such as Gaussian initialization. This leads to the newly introduced neural modules producing feature outputs with significantly different data distributions, causing model instability when these outputs are directly added or combined.

Normalization techniques [2, 23, 62] standardize layer inputs, improving training stability and convergence. Inspired by these methods, we propose *cross normalization* to align processed conditional controls with the main branch features, ensuring stable and efficient training.

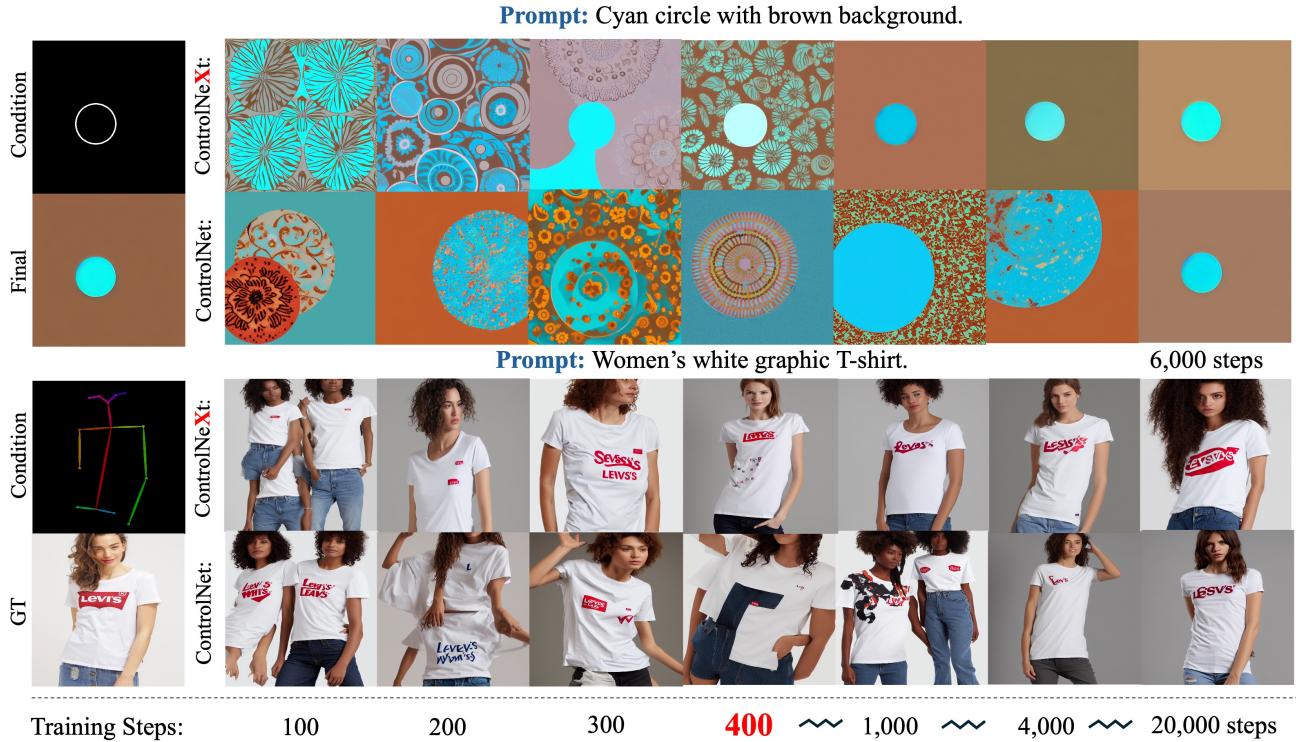


Figure 3. ControlNeXt achieves significantly faster training convergence and data fitting. It can learn to fit the conditional controls with fewer training steps, which significantly alleviates the *sudden convergence* problem.

We represent the feature maps processed by the base model and the lightweight convolution blocks as \mathbf{x}_m and \mathbf{x}_c , respectively, where $\mathbf{x}_m, \mathbf{x}_c \in \mathbb{R}^{h \times w \times c}$. The key to Cross Normalization is to use the mean and variance calculated from the main branch \mathbf{x}_m to normalize the control features \mathbf{x}_c , ensuring their alignment. First, calculate the channel-wise mean and variance of the denoising features,

$$\boldsymbol{\mu}_m = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_{m,i}, \quad (8)$$

$$\sigma_m^2 = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_{m,i} - \boldsymbol{\mu}_m)^2, \quad (9)$$

where $n = h \times w \times c$. Then, we normalize the control features using the mean and variance of the denoising features,

$$\hat{\mathbf{x}}_c = \frac{\mathbf{x}_c - \boldsymbol{\mu}_m}{\sqrt{\sigma_m^2 + \varepsilon}} * \gamma, \quad (10)$$

where ε is a small constant added for numerical stability and γ is a parameter that allows the model to scale the normalized value. $\mathbf{x}_c = \mathcal{F}_c(\mathbf{c}; \Theta_c)$ is the output control feature.

Cross Normalization aligns the distributions of the denoising and control features, serving as a bridge to connect the base model and control blocks. Our experiments in Sec.4.1 show that this approach accelerates the training process, enabling the base model to capture guiding information from the outset. It facilitates early convergence

and significantly alleviates the “sudden convergence” phenomenon.

4. Experiments

In this section, we present a series of experiments across various tasks and backbones. Our method exhibits exceptional efficiency and generality.

4.1. Training Convergence

A typical problem for the controllable generation is the hard training convergence, which means that it requires thousands or more than ten thousands steps training to learn the conditional controls. This phenomenon, known as the *sudden convergence* problem [70], occurs when the model initially fails to learn the control ability and then suddenly acquires this skill. This is caused from such two aspects:

1. *Zero convolution* inhibits the influence of the loss function, resulting in a prolonged warm-up phase where the model struggles to start learning effectively.
2. The pretrained generation model is completely frozen, and ControlNet or the adapter cannot immediately affect the performance of the model.

In ControlNeXt, we eliminate these two limitations, resulting in significantly faster training. We conducted experiments using two types of controls, and the results and comparisons are shown in Fig. 3. It can be seen that ControlNeXt starts to converge after only a few hundred training



Figure 4. Detailed generation results of the stable video diffusion. We utilize the pose sequence as guidance for character animation.

Source ————— Canny

Generated



Figure 5. Detailed generation results of the stable diffusion XL are provided. We extract the Canny edges from the input image and implement the style transfer utilizing the SDXL model integrated with our proposed ControlNeXt framework.

steps, while ControlNet requires thousands of steps. ControlNeXt significantly alleviates the *sudden convergence* problem.

4.2. Generality

To demonstrate the generality of our methods, we first apply our approach to various diffusion-based backbones, including Stable Diffusion 1.5 [18, 56], Stable Diffusion XL [46], Stable Diffusion 3 [12], and Stable Video Diffusion [4]. Our method covers a wide range of tasks, such as image generation, high-resolution generation, and video generation, utilizing various types of conditional controls. Qualitative results are shown in Fig.1. Additionally, more generation results for stable video generation, where we use

pose sequences as guidance for character animation, are presented in Fig.4. The results for SDXL are displayed in Fig. 5, where we implement style transfer by extracting Canny edges from the input images and generating the output with our SDXL model. The results show that our method is adapting to various architectures and tasks.

Various conditional controls. ControlNeXt also supports various types of conditional controls. In this subsection, we choose “mask”, “depth”, “pose” and “canny” as the conditional controls, shown in Fig. 6 from top to bottom, respectively. All the experiments are constructed based on the Stable Diffusion 1.5 architecture [56].

Quantitative Results. Tab.1 show a quantitative comparison on ADE20K and COCO [32, 75] with Stable Diffusion

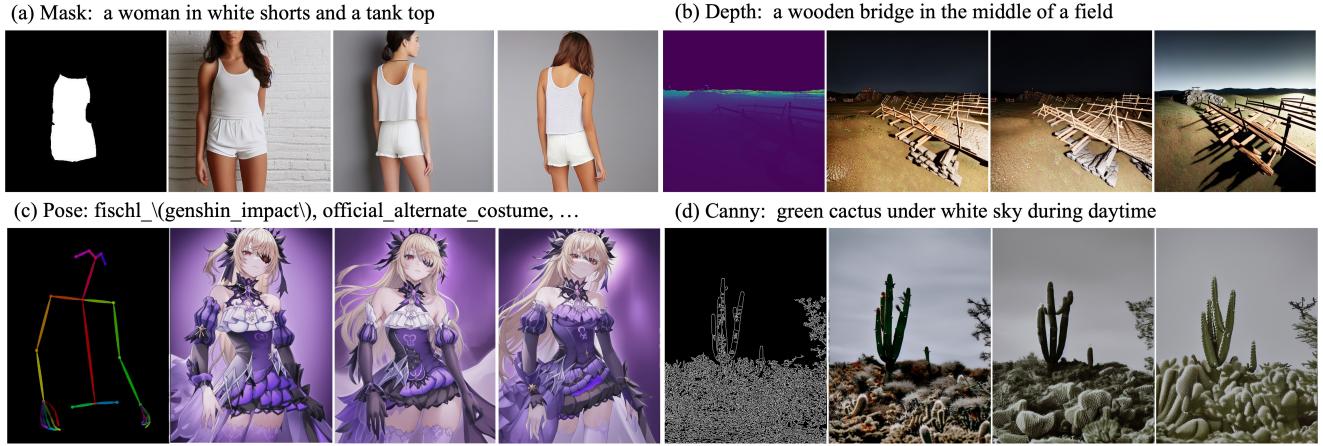


Figure 6. ControlNeXt supports various conditional controls types. We select “mask”, “depth”, “pose” and “canny”, as the conditions.

Metrics Method	Clip-score (\uparrow) Seg. Mask ADE20K COCO		FID (\downarrow) HED MLSD COCO COCO		
	Pose COCO				
Gligen [30]	31.1	-	28.6	-	24.6
ControlNet [70]	31.5	13.3	26.6	31.4	27.8
T2I-Adapter [41]	30.6	-	-	-	29.6
ControlNet++ [29]	31.9	13.3	-	-	-
Uni-Control [47]	30.9	-	17.9	26.2	26.6
ControlNeXt_(SD1.5)	32.7	29.5	20.4	21.1	23.0

Table 1. Comparison of different methods across metrics (Clip-score, FID) using Stable Diffusion 1.5 as the backbone.

SR Method	PSNR \uparrow	SSIM \uparrow	CLIPQA \uparrow	DISTS \downarrow	MUSIQ \uparrow
BSRGAN [69]	26.50	0.69	0.24	0.36	25.22
SinSR [60]	26.83	0.64	0.62	0.29	56.57
SUPIR [68]	25.22	0.61	0.56	0.26	59.02
Ours (SD3)	27.31	0.71	0.64	0.20	62.95

Table 2. Quantitative results on super-resolution, evaluated with DRealSR [61], highlight Stable Diffusion 3 as the backbone.

Method	MagicPose [6]	MuseV [63]	Mimic [73]	ControlNeXt
FVD (\downarrow)	916	754	594	576

Table 3. Comparison of character animation performance [24] using stable video diffusion.

1.5 as backbone. ControlNeXt achieves state-of-the-art results with efficiency and generality. For the super-resolution task, we conduct experiments using the DRealSR benchmark [61] with Stable Diffusion 3 as the backbone in a DiT-based architecture. Results in Tab. 2 highlight our advancements. Following prior work [73], we evaluate video generation on the character animation task using the Tik-Tok dataset [24]. Results are presented in Tab. 3. Furthermore, we apply our method to the DiT-based video generation backbone [19, 31, 67], Open-Sora-Plan, for video out-painting tasks, following the setup of prior work [13]. The results are shown in Tab. 4.

(b) Depth: a wooden bridge in the middle of a field



(c) Pose: fischl_(genshin_impact), official_alternate_costume, ...

(d) Canny: green cactus under white sky during daytime



Figure 6. ControlNeXt supports various conditional controls types. We select “mask”, “depth”, “pose” and “canny”, as the conditions.

Method	DAVIS dataset [50]			YouTube-VOS [11]		
	PSNR \uparrow	SSIM \uparrow	FVD \downarrow	PSNR \uparrow	SSIM \uparrow	FVD \downarrow
SDM [13]	20.02	0.7078	334.6	19.91	0.7277	94.8
M3DDM [13]	20.26	0.7082	300.0	20.20	0.7312	66.6
Ours(Open-Sora-Plan)	20.33	0.7576	290.7	20.23	0.7661	60.3

Table 4. Video outpainting task with Open-Sora-Plan as backbone.

Backbone	ControlNet		ControlNeXt (Ours)		Base Model Total
	Total	Learnable	Total	Learnable	
SD1.5	1,220	361	865	30	859
SDXL	3,818	1,251	2,573	108	2,567
SVD	2,206	682	1,530	55	1,524

Table 5. Comparison of the total and learnable parameters of different methods with various backbones.

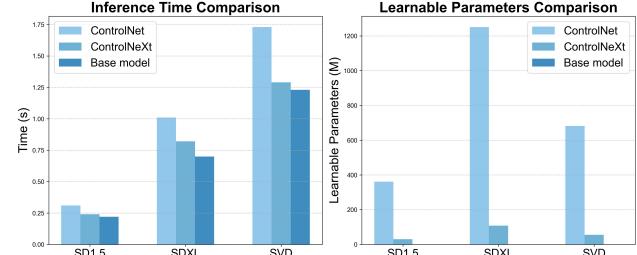


Figure 7. Efficiency comparisons of ControlNeXt.

4.3. Efficiency

In this section, we compare the efficiency of various backbones, focusing primarily on comparison with ControlNet [70] for its representativeness and generalizability. Alternatives such as T2I-Adapter [41] are limited to image generation and lack support for all tasks and backbones. Further details are provided in the supplementary materials. A comprehensive comparison is shown in Fig. 7.

Parameters. We present statistics on the parameters, including the total and learnable parameters, calculated only for the UNet model (excluding the VAE and encoder parts). And the results are shown in Tab. 5. It can be seen that



Figure 8. Our method can serve as a plug-and-play module that adapts to various LoRA weights with training-free.



Figure 9. ControlNeXt is compatible with a variety of backbones.
our method only adds a lightweight module with minimal additional parameters, maintaining consistency with the original pretrained model. As for training, our method requires at most less than 10% of the learnable parameters. You can also adaptively adjust the amount of learnable parameters for various tasks and performance requirements.

Inference time. We compare the inference time of different methods with various base models. The results are shown in Tab. 6, which presents the computational time of one inference step, considering only the UNet and ControlNet. It can be seen that our method increases latency minimally compared to the pretrained base generation model. This ensures outstanding efficiency advantages for our method.

4.4. Additional Studies

Training free integration. We first collected various LoRA weights downloaded from Civitai [1], encompassing diverse generation styles. We then construct experiments on various backbones, including SD1.5 [56], AnythingV3 [57] and DreamShaper [38]. The results are shown in Fig. 8 and Fig. 9. It can be observed that ControlNeXt can integrate with various backbones and LoRA weights in a training-free manner, effectively altering the quality and styles of generated images. It also facilitates stable generation with minimal effort and cost as shown in Fig. 10. We use a simple text prompt, *i.e.*, “one girl,” with the ‘pose’ condition, enabling high-quality generation without detailed textual descriptions.



Figure 10. ControlNeXt serving as a plugin-unit to ensure a stable generation with minimal costs.



Figure 11. Controllable generation under multiple conditions.

Method	Inference Time (s)			Δ
	SD1.5	SDXL	SVD	
ControlNet	0.31	1.01	1.73	+ 41.9%
ControlNeXt_(Ours)	0.24	0.82	1.29	+ 10.4%
Base model	0.22	0.70	1.23	-

Table 6. Comparison of inference time with various backbones.

	FID↓/Clip↑	CrossNorm	Concat	CrossAttn	ControlNet
HED	20.4 / 29.4	27.2 / 28.9	271 / 22.7	26.6 / -	
MLSD	21.1 / 29.2	24.5 / 28.8	381 / 20.6	31.4 / -	

Table 7. Comparative analysis of different methods for integrating conditional controls.

Multiple conditions. We fine-tune lightweight control modules for ‘depth’ and ‘pose’ conditions and integrate them into the main branch without other operations. Learnable parameters in the main branch are assigned to non-overlapping blocks for each condition. Results in Fig. 11 demonstrate our method’s support for multiple conditions.

Information intergration. We conduct ablation studies to validate our method’s effectiveness, with results in Tab. 7. Beyond qualitative improvements, our approach allows adjustable control impact on the backbone, including turning off the control signal—unlike direct concatenation or cross-attention. This is especially beneficial when combined with classifier-free-guidance for optimal results.

5. Conclusion

This paper presents ControlNeXt, an advanced and efficient method for controllable image and video generation. ControlNeXt employs a compact architecture, eliminating heavy auxiliary components to minimize latency overhead and reduce trainable parameters. We propose *Cross Normalization* for finetuning pre-trained large models, improving training convergence in both speed and stability. Extensive experiments across various image and video generation backbones demonstrate the effectiveness and generality.

References

- [1] Civitai 2024. Civitai. <https://civitai.com>, 2024. 8
- [2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4
- [3] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. *arXiv preprint arXiv:2209.12152*, 2022. 3
- [4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2, 6
- [5] Pu Cao, Feng Zhou, Qing Song, and Lu Yang. Controllable generation with text-to-image diffusion models: A survey. *arXiv preprint arXiv:2403.04279*, 2024. 2
- [6] Di Chang, Yichun Shi, Quankai Gao, Hongyi Xu, Jessica Fu, Guoxian Song, Qing Yan, Yizhe Zhu, Xiao Yang, and Mhammad Soleymani. Magicpose: Realistic human poses and facial expressions retargeting with identity-aware diffusion. In *Forty-first International Conference on Machine Learning*, 2023. 7
- [7] Zhe Chen, Yuchen Duan, Wenhui Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 2
- [8] Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022. 3
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. 3
- [10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2
- [11] Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Capsulevos: Semi-supervised video object segmentation using capsule routing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8480–8489, 2019. 7
- [12] Patrick Esser et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, 2024. 3, 6
- [13] Fanda Fan, Chaoxu Guo, Litong Gong, Biao Wang, Tiezheng Ge, Yuning Jiang, Chunjie Luo, and Jianfeng Zhan. Hierarchical masked 3d diffusion model for video inpainting. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7890–7900, 2023. 7
- [14] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2
- [15] Zeyu Han, Chao Gao, Jinyang Liu, Sai Qian Zhang, et al. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024. 2
- [16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4
- [17] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4918–4927, 2019. 4
- [18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NIPS*, pages 6840–6851, 2020. 2, 6
- [19] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 7
- [20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2, 4
- [21] Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8153–8163, 2024. 2
- [22] Lianghua Huang, Di Chen, Yu Liu, Yujun Shen, Deli Zhao, and Jingren Zhou. Composer: Creative and controllable image synthesis with composable conditions. *arXiv preprint arXiv:2302.09778*, 2023. 3
- [23] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 4
- [24] Yasamin Jafarian and Hyun Soo Park. Learning high fidelity depths of dressed humans by watching social media dance videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12753–12762, 2021. 7
- [25] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion image-to-video synthesis via stable diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22623–22633. IEEE, 2023. 2
- [26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 2
- [27] Siddharth Krishna Kumar. On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863*, 2017. 4
- [28] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 4
- [29] Ming Li, Taojiaann Yang, Huafeng Kuang, Jie Wu, Zhaoning Wang, Xuefeng Xiao, and Chen Chen. Controlnet

++

- : Improving conditional controls with efficient consistency feedback. In *European Conference on Computer Vision*, pages 129–147. Springer, 2025. 7
- [30] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 7
- [31] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shanghai Yuan, Lihuan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. 7
- [32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 6
- [33] Zuzeng Lin, Ailin Huang, and Zhewei Huang. Collaborative neural rendering using anime character sheets. *arXiv preprint arXiv:2207.05378*, 2022. 2
- [34] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965, 2022. 4
- [35] Liying Lu, Wenbo Li, Xin Tao, Jiangbo Lu, and Jiaya Jia. Masa-sr: Matching acceleration and spatial adaptation for reference-based image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6368–6377, 2021. 3, 4
- [36] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022. 3
- [37] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023. 3
- [38] Lykon. Dreamshaper. <https://huggingface.co/Lykon/DreamShaper>, 2022. 8
- [39] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. Peft: State-of-the-art parameter-efficient fine-tuning methods. <https://github.com/huggingface/peft>, 2022. 2
- [40] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023. 3
- [41] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024. 2, 3, 7
- [42] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2021. 2
- [43] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2
- [44] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 3
- [45] Pablo Pernas, Dominic Rampas, Mats L. Richter, Christopher J. Pal, and Marc Aubreville. Wuerstchen: An efficient architecture for large-scale text-to-image diffusion models, 2023. 3
- [46] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2, 3, 6
- [47] Can Qin, Shu Zhang, Ning Yu, Yihao Feng, Xinyi Yang, Yingbo Zhou, Huan Wang, Juan Carlos Niebles, Caiming Xiong, Silvio Savarese, et al. Unicontrol: A unified diffusion model for controllable visual generation in the wild. *arXiv preprint arXiv:2305.11147*, 2023. 7
- [48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3
- [49] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020. 3
- [50] Sucheng Ren, Chu Han, Xin Yang, Guoqiang Han, and Shengfeng He. Tenet: Triple excitation network for video salient object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 212–228. Springer, 2020. 7
- [51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3
- [52] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 2
- [53] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training

- next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 4
- [54] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NIPS*, 2019. 2
- [55] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023. 3
- [56] Stability. Stable diffusion v1.5. <https://huggingface.co/runwayml/stable-diffusion-v1-5>, 2022. 2, 3, 6, 8
- [57] Furqanil Taqwa. Anything v3. <https://huggingface.co/Linaqruf/anything-v3.0>, 2022. 8
- [58] Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*, 2020. 4
- [59] Tan Wang, Linjie Li, Kevin Lin, Yuanhao Zhai, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. Disco: Disentangled control for realistic human dance generation. *arXiv preprint arXiv:2307.00040*, 2023. 2, 3
- [60] Yufei Wang, Wenhan Yang, Xinyuan Chen, Yaohui Wang, Lanqing Guo, Lap-Pui Chau, Ziwei Liu, Yu Qiao, Alex C Kot, and Bihan Wen. Sinsr: diffusion-based image super-resolution in a single step. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 25796–25805, 2024. 7
- [61] Pengxu Wei, Ziwei Xie, Hannan Lu, Zongyuan Zhan, Qixiang Ye, Wangmeng Zuo, and Liang Lin. Component divide-and-conquer for real-world image super-resolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 101–117. Springer, 2020. 7
- [62] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4
- [63] Zhiqiang Xia, Zhaokang Chen, Bin Wu, Chao Li, Kwok-Wai Hung, Chao Zhan, Yingjie He, and Wenjiang Zhou. Musev: Infinite-length and high fidelity virtual human video generation with visual conditioned parallel denoising. *arxiv*, 2024. 7
- [64] Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. Parameter-efficient fine-tuning for pre-trained vision models: A survey. *arXiv preprint arXiv:2402.02242*, 2024. 2
- [65] Jinbo Xing, Menghan Xia, Yuxin Liu, Yuechen Zhang, Yong Zhang, Yingqing He, Hanyuan Liu, Haoxin Chen, Xiaodong Cun, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Make-your-video: Customized video generation using textual and structural guidance. *arXiv preprint arXiv:2306.00943*, 2023. 3
- [66] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pages 399–417. Springer, 2025. 2
- [67] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 7
- [68] Fanghua Yu, Jinjin Gu, Zheyuan Li, Jinfan Hu, Xiangtao Kong, Xintao Wang, Jingwen He, Yu Qiao, and Chao Dong. Scaling up to excellence: Practicing model scaling for photo-realistic image restoration in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25669–25680, 2024. 7
- [69] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4791–4800, 2021. 7
- [70] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 2, 3, 4, 5, 7
- [71] Pengze Zhang et al. Tackling the singularities at the endpoints of time intervals in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 3, 4
- [72] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Ao-jun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 2, 4
- [73] Yuang Zhang, Jiaxi Gu, Li-Wen Wang, Han Wang, Junqi Cheng, Yuefeng Zhu, and Fangyuan Zou. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance. *arXiv preprint arXiv:2406.19680*, 2024. 3, 7
- [74] Yuechen Zhang, Shengju Qian, Bohao Peng, Shu Liu, and Jiaya Jia. Prompt highlighter: Interactive control for multi-modal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13215–13224, 2024. 2
- [75] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 6

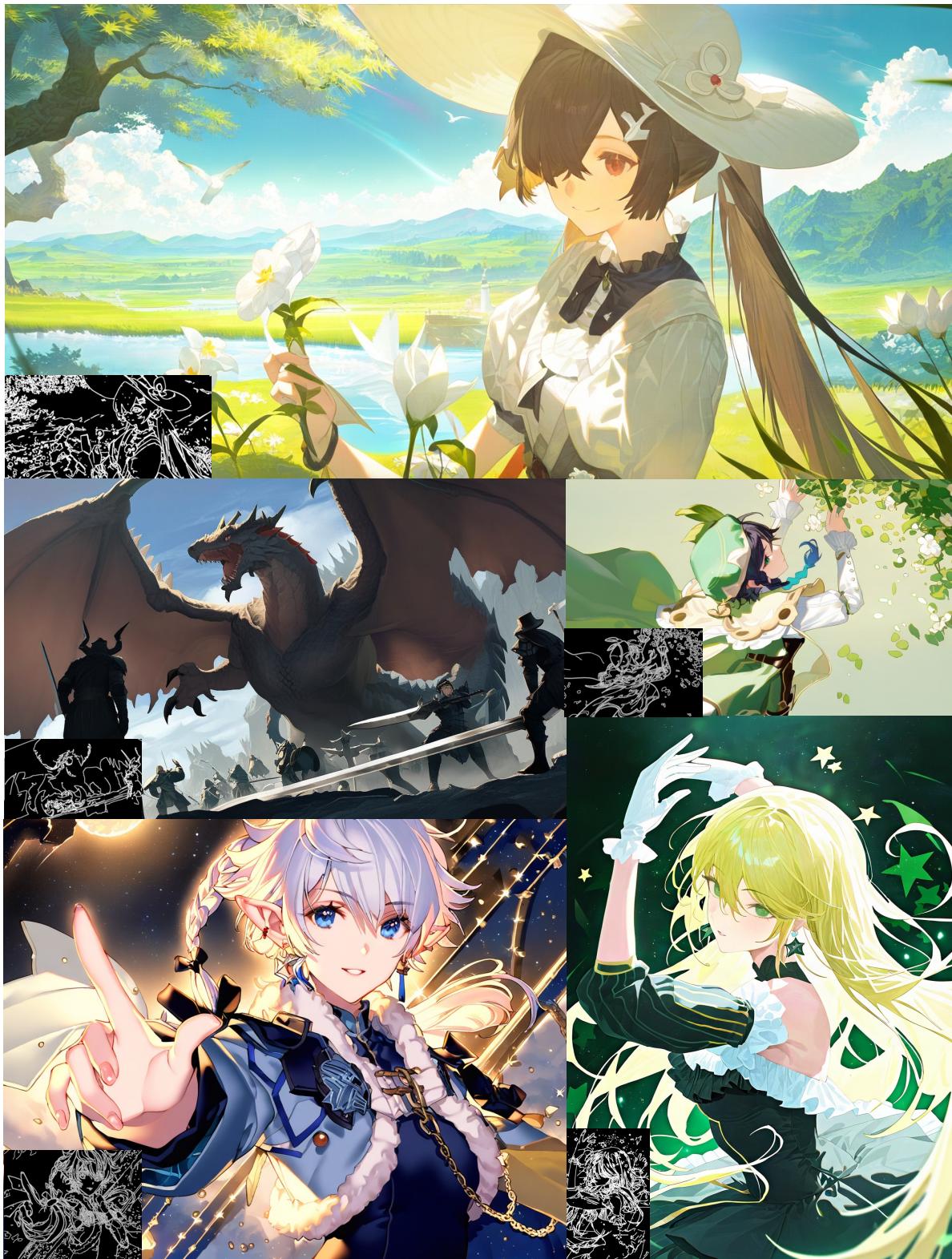


Figure 12. Stable Diffusion XL.



Figure 13. Stable video diffusion.

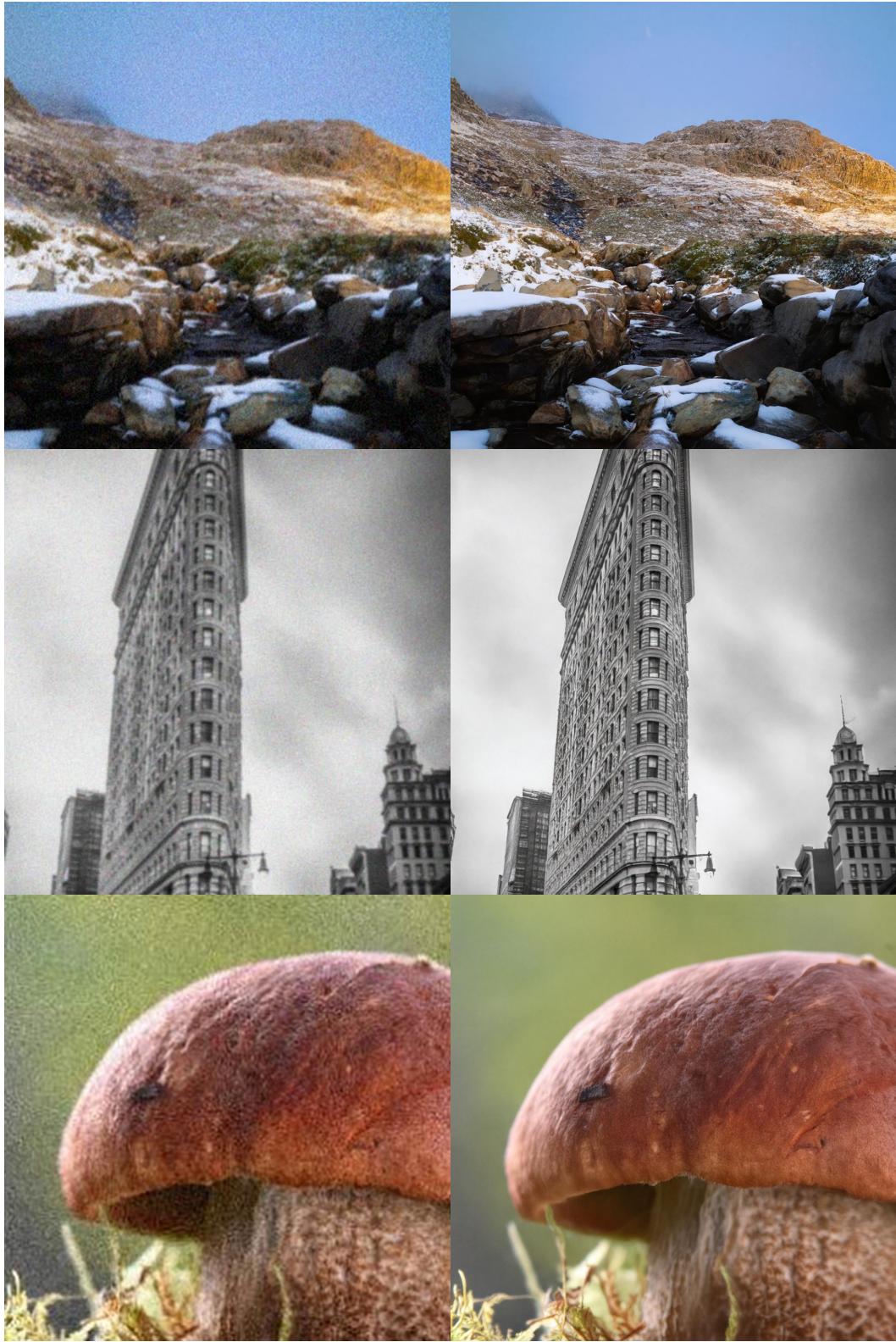


Figure 14. Stable Diffusion 3.

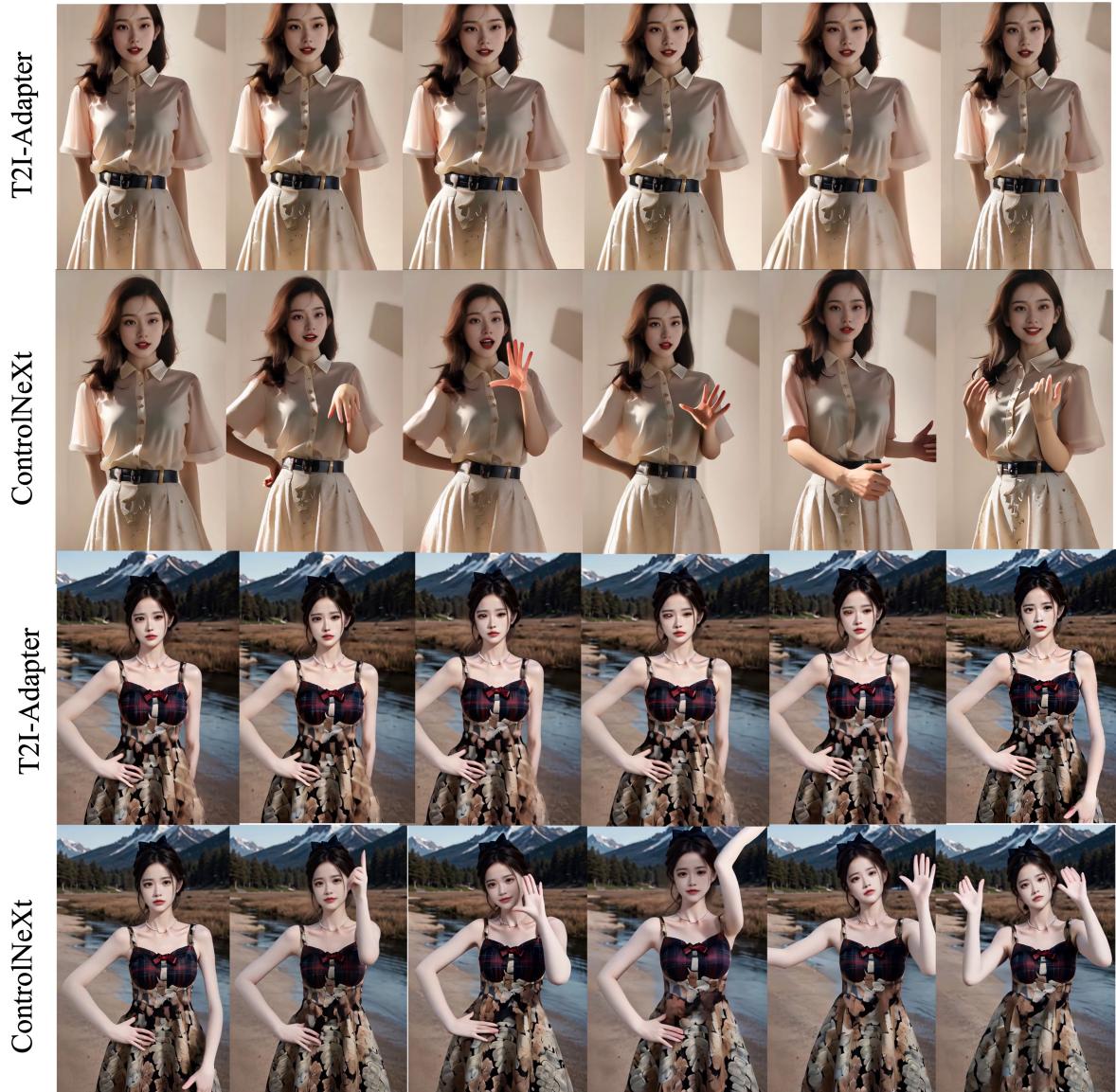


Figure 15. Comparison with the T2I-adapter. The T2I-adapter is specifically designed for image generation and is challenging to adapt for more complex tasks, such as video generation.