PIXART-δ: Fast and Controllable Image Generation with Latent Consistency Models

Anonymous Authors[1]

Abstract

This paper introduces PIXART-δ, a text-to-image synthesis framework that integrates the Latent Consistency Model (LCM) and ControlNet into the advanced PIXART-α model. PIXART-α is recognized for its ability to generate high-quality images of 1024px resolution through a remarkably efficient training process. The integration of LCM in PIXART-δ significantly accelerates the inference speed, enabling the production of high-quality images in just 2-4 steps. Notably, PIXART-δ achieves a breakthrough 0.5 seconds for generating 1024 $\times$ 1024 pixel images, marking a 7$\times$ improvement over the PIXART-α. Additionally, PIXART-δ is designed to be efficiently trainable on 32GB V100 GPUs within a single day. With its 8-bit inference capability (von Platen et al., 2023), PIXART-δ can synthesize 1024px images within 8GB GPU memory constraints, greatly enhancing its usability and accessibility. Furthermore, incorporating a ControlNet-like module enables fine-grained control over text-to-image diffusion models. We introduce a novel ControlNet-Transformer architecture, specifically tailored for Transformers, achieving explicit controllability alongside high-quality image generation. As a state-of-the-art, open-source image generation model, PIXART-δ offers a promising alternative to the Stable Diffusion series, contributing significantly to text-to-image synthesis.

1. Introduction

In this paper, we propose PIXART-δ, which incorporates LCM (Luo et al., 2023a) and ControlNet (Zhang et al., 2023) into PIXART-α (Chen et al., 2023). Notably, PIXART-α is an advanced high-quality 1024px diffusion transformer text-to-image synthesis model, developed by our team, known for its superior image generation quality achieved through an exceptionally efficient training process.

We incorporate LCM into the PIXART-δ to accelerate the inference. LCM (Luo et al., 2023a) enables high-quality and fast inference with only 2$\sim$4 steps on pre-trained LDMs by viewing the reverse diffusion process as solving an augmented probability flow ODE (PF-ODE), which enables PIXART-δ to generate samples within ($\sim$4) steps while preserving high-quality generations. As a result, PIXART-δ takes 0.5 seconds per 1024 $\times$ 1024 image on an A100 GPU, improving the inference speed by 7$\times$ compared to PIXART-α. We also support LCM-LoRA (Luo et al., 2023b) for a better user experience and convenience.

In addition, we incorporate a ControlNet-like module into the PIXART-δ. ControlNet (Zhang et al., 2023) demonstrates superior control over text-to-image diffusion models' outputs under various conditions. However, it's important to note that the model architecture of ControlNet is intricately designed for UNet-based diffusion models, and we observe that a direct replication of it into a Transformer model proves less effective. Consequently, we propose a novel ControlNet-Transformer architecture customized for the Transformer model. Our ControlNet-Transformer achieves explicit controllability and obtains high-quality image generation.

2. LCM in PIXART-δ

In this section, we employ Latent Consistency Distillation (LCD) (Luo et al., 2023a) to train PIXART-δ on 120K internal image-text pairs. In Sec. 2.1, we first provide a detailed training algorithm and ablation study on specific modifications. In Sec. 2.2, we illustrate the training efficiency and the speedup of LCM of PIXART-δ. Lastly, in Sec. B.1, we present the training details of PIXART-δ.

2.1. Algorithm and modification

LCD Algorithm. Deriving from the original Consistency Distillation (CD) (Song et al., 2023) and LCD (Luo et al., 2023a) algorithm, we present the pseudo-code for **PIXART-δ** with classifier-free guidance (CFG) in Algo-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

rithm 1. Specifically, as illustrated in the training pipeline shown in Fig. 1, three models – Teacher, Student, and EMA Model – function as denoisers for the ODE solver $\Psi(\cdot,\cdot,\cdot,\cdot)$, $\boldsymbol{f_\theta}$, and $\boldsymbol{f_{\theta^-}}$, respectively. During the training process, we begin by sampling noise at timestep t_{n+k}, where the Teacher Model is used for denoising to obtain $\hat{z}_{T_{t_0}}$. We then utilize a ODE solver $\Psi(\cdot,\cdot,\cdot,\cdot)$ to calculate $\hat{z}_{t_n}^{\Psi,\omega}$ from $z_{t_{n+k}}$ and $\hat{z}_{T_{t_0}}$. EMA Model is then applied for further denoising, resulting in $\hat{z}_{E_{t_0}}$. In parallel, the Student Model denoises the sample $z_{t_{n+k}}$ at t_{n+k} to derive $\hat{z}_{S_{t_0}}$. The final step involves minimizing the distance between $\hat{z}_{S_{t_0}}$ and $\hat{z}_{E_{t_0}}$, also known as optimizing the consistency distillation objective.

Different from the original LCM, which selects variable guidance scale ω from a designated range $[\omega_{min}, \omega_{max}]$, in our implementation, we set the guidance scale as a constant ω_{fix}, removing the guidance scale embedding operation in LCM (Luo et al., 2023a) for convenience.

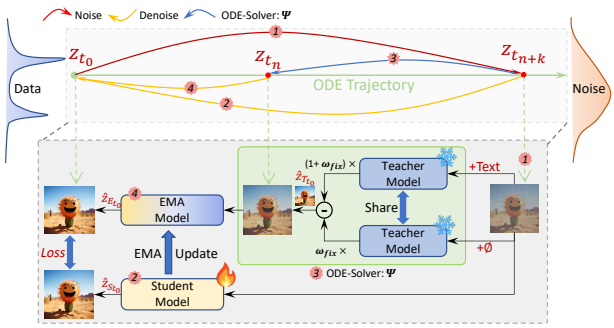


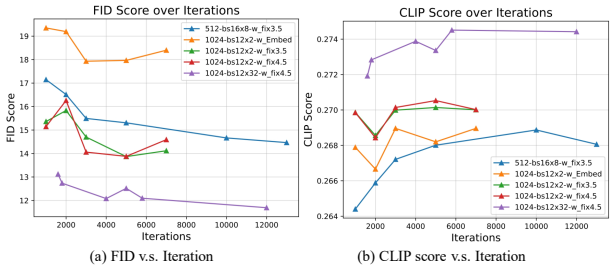
Figure 1: Training pipeline of PIXART-δ. The upper section of the diagram offers a high-level overview of the training process, depicting the sequential stages of noise sampling and denoising along a specific ODE trajectory. Sequence numbers are marked on the mapping lines to clearly indicate the order of these steps. The lower section delves into the intricate roles of the pre-trained (teacher) model and the student model, revealing their respective functions within the upper block's training process, with corresponding sequence numbers also marked for easy cross-referencing.

Effect of Hyper-parameters. Our study complements two key aspects of the LCM training process, CFG scale and batch size. These factors are evaluated using FID and CLIP scores as performance benchmarks. The terms 'bs', 'ω_fix', and 'ω_Embed' in the Fig. 2 represent training batch size, fixed guidance scale, and embedded guidance scale, respectively.

- **CFG Scale Analysis:** Referencing Fig. 2, we examine three distinct CFG scales: (1) 3.5, utilized in our ablation study; (2) 4.5, which yieldes optimal results in PIXART-α; and (3) a varied range of CFG scale embeddings (ω_Embed), the standard approach in LCM. Our research reveals that employing a constant guidance scale, instead of the more complex CFG embeddings improves performance in PIXART-δ and simplifies the implementation.

- **Batch Size Examination:** The impact of batch size on model performance is assessed using two configurations: 2 V100 GPUs and 32 V100 GPUs; each GPU loads 12 images. As illustrated in Fig. 2, our results indicate that larger batch size positively influences FID and CLIP scores. However, as shown in Fig. 8, PIXART-δ can also converge fast and get comparable image quality with smaller batch sizes.

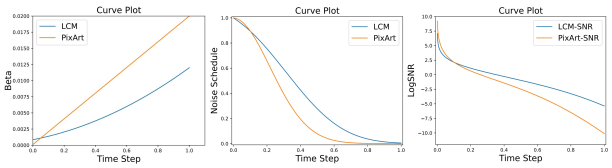
- **Convergence:** Finally, we observe that the training process tends to reach convergence after approximately 5,000 iterations. Beyond this phase, further improvements are minimal.



(a) FID v.s. Iteration (b) CLIP score v.s. Iteration

Figure 2: Ablation study of FID and CLIP Score on various strategies for classifier-free guidance scale (ω) and their impact on distillation convergence during training.

Noise Schedule Adjustment. Noise schedule is one of the most important parts of the diffusion process. Following (Hoogeboom et al., 2023; Chen, 2023), we adapt the noise schedule function in LCM to align with the PIXART-α noise schedule, which features a higher logSNR (signal-to-noise ratio) during the distillation training. Fig. 3 visualizes the noise schedule functions under different choices of PIXART-δ or LCM, along with their respective logSNR. Notably, PIXART-δ can parameterize a broader range of noise distributions, a feature that has been shown further to enhance image generation (Hoogeboom et al., 2023; Chen, 2023).



(a) Beta v.s. Time step (b) Noise schedule v.s. Time step (c) LogSNR v.s. Time step

Figure 3: Instantiations of β_t, noise schedule function and the corresponding logSNR between PIXART-δ and LCM. β_t is the coefficient in the diffusion process $z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \alpha_t = 1 - \beta_t$.

2.2. Training efficiency and inference speedup

For training, as illustrated in Tab. 1, we successfully conduct the distillation process within a 32GB GPU memory constraint, all while retaining the same batch size and supporting image resolution up to 1024 × 1024 with SDXL-LCM. Such training efficiency remarkably enables PIXART-δ to be trained on a wide array of consumer-grade GPU specifications. In light of the discussions in Sec.2.1, regarding the beneficial impact of larger batch size, our method notably makes it feasible to utilize larger batch size even on GPUs with limited memory capacity. Refer to B.1 for more training details.

For inference, as shown in Tab. 2 and Fig. 7, we present a comparative analysis of the generation speed achieved by our model, PIXART-δ, against other methods like SDXL LCM-LoRA, PIXART-α, and the SDXL standard across different hardware platforms. Consistently, PIXART-δ achieves **1024x1024 high resolution** image generation within **0.5 seconds** on an A100, and also completes the process in a mere 3.3 seconds on a T4, 0.8 seconds on a V100, all with a batch size of 1. This is a significant improvement over the other methods, where, for instance, the SDXL standard takes up to 26.5 seconds on a T4 and 3.8 seconds on an A100. The efficiency of PIXART-δ is evident as it maintains a consistent lead in generation speed with only 4 steps, compared to the 14 and 25 steps required by PIXART-α and SDXL standard, respectively. Notably, with the implementation of 8-bit inference technology, PIXART-δ requires less than **8GB of GPU VRAM**. This remarkable efficiency enables PIXART-δ to operate on a wide range of GPU cards, and it even opens up the possibility of running on a CPU.

Table 1: Illustration of the training setting between LCM on PIXART-δ and Stable Diffusion models. (* stands for Stable Diffusion Dreamshaper-v7 finetuned version)

Methods	**PIXART-δ**	SDXL LCM-LoRA	SD-V1.5-LCM*
Data Volume	120K	650K	650K
Resolution	1024px	1024px	768px
Batch Size	12 × 32	12 × 64	16 × 8
GPU Memory	∼32G	∼80G	∼80G

Table 2: Illustration of the generation speed we achieve on various devices. These tests are conducted on 1024 × 1024 resolution with a batch size of 1 in all cases. Corresponding image samples are shown in the Fig. 7

Hardware	**PIXART-δ**	SDXL LCM-LoRA	PIXART-α	SDXL standard
	4 steps	4 steps	14 steps	25 steps
T4	3.3s	8.4s	16.0s	26.5s
V100	0.8s	1.2s	5.5s	7.7s
A100	0.5s	1.2s	2.2s	3.8s

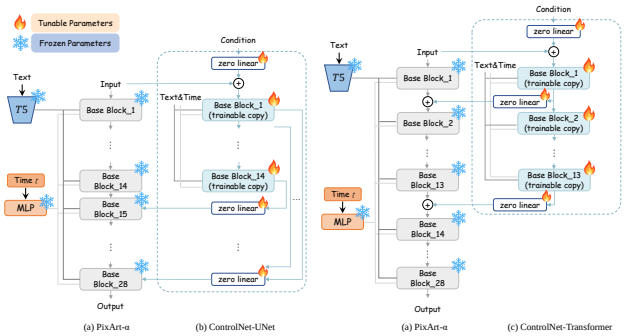


Figure 4: PIXART-δ integrated with ControlNet. (b): ControlNet-UNet. Base blocks are categorized into "encoder" and "decoder" stages. The controlnet structure is applied to each encoder level of PIXART-δ, and the output is connected to the decoder stage via skip-connections. (c): ControlNet-Transformer. The ControlNet is applied to the first several blocks. The output of each block is added to the output of the corresponding frozen block, serving as the input of the next frozen block.

3. ControlNet in PIXART-δ

3.1. Architecture

ControlNet, designed for UNet architecture, employed skip connections to enhance the integration of control signals. Incorporating ControlNet into Transformer-based models like PIXART-δ poses a unique challenge, as Transformers lack distinct "encoder" and "decoder" blocks, making conventional connections inappropriate. To address this, we propose ControlNet-Transformer for effective integration with Transformers.

PIXART-δ contains 28 Transformer blocks. We replace the original zero-convolution in ControlNet with a zero linear layer, that is, a linear layer with both weight and bias initialized to zero. We explore the following networks:

- **ControlNet-UNet** (Zhang et al., 2023). To follow the original ControlNet design, we treat the first 14 blocks as the "encoder" level of PIXART-δ, and the last 14 blocks as the "decoder" level of PIXART-δ. We use ControlNet to create a trainable copy of the 14 encoding blocks. Subsequently, the outputs from these blocks are integrated by addition into the 14 skip-connections, which link to the last 14 decoder blocks. The network design is shown in Fig. 4 (b).

  It is crucial to note that this adaptation, referred to as ControlNet-UNet, encounters challenges due to the absence of explicit "encoder" and "decoder" stages and skip-connections in the original Transformer design. This adaptation departs from the conventional architecture of the Transformer, which hampers the effectiveness and results in suboptimal outcomes.

- **ControlNet-Transformer**. To address these challenges, we propose a novel and specifically tailored design for Transformers, illustrated in Fig. 4 (c). This innovative approach aims to seamlessly integrate the ControlNet structure with the inherent characteristics of Transformer architectures. We selectively apply the ControlNet structure to the initial N base blocks. In this context, we generate N trainable copies of the first N base blocks. The output of ith trainable block is intricately connected to a zero linear layer, and the resulting output is then added to the output of the corresponding i^{th} frozen block. Subsequently, this combined output serves as the input for the subsequent $(i+1)^{th}$ frozen block. This design adheres to the original data flow of PixArt, and our observations underscore the significant enhancement in controllability and performance achieved by ControlNet-Transformer. This approach represents a crucial step toward harnessing the full potential of Transformer-based models in such applications. The ablation study of N is described in Sec. 3.3, and we use $N = 13$ as the final model.

3.2. Experiment Settings

We use a HED edge map in PixArt-δ as the condition and conduct an ablation study on 512px generation, focusing on network architecture variations. Specifically, we conduct ablations on both the ControlNet-UNet and ControlNet-Transformer. Other conditions, such as canny, will be a future work. For ControlNet-Transformer, we ablate the number of copied blocks, including 1, 4, 7, 13, and 27. We extract the HED on the internal data, and the gradient accumulation step is set as 4 following (Zhang et al., 2023) that recommends that larger gradient accumulation leads to improved results. The optimizer and learning rate are set as the same setting of PixArt-δ. All the experiments are conducted on 16 V100 GPUs with 32GB. The batch size per GPU for experiment ControlNet-Transformer ($N = 27$) is set as 2. For all other experiments, the batch size is set as 12. Our training set consists of 3M HED and image pairs.

3.3. Ablation Study

As shown in Fig. 5, ControlNet-Transformer generally outperforms by demonstrating faster convergence and improved performance. This superiority stems from its seamless alignment with the inherent data flow of Transformer architectures. In contrast, ControlNet-UNet introduces an artificial information flow between non-existent "encoder" and "decoder" stages, deviating from the Transformer's natural data processing pattern.

In our ablation study concerning the number of copied blocks, we observe that for the majority of scenarios, such as scenes and objects, satisfactory results can be achieved with merely $N = 1$. However, in challenging edge conditions, such as the outline edge of human faces and bodies, performance tends to improve as N increases. Considering a balance between computational burden and performance, we find that $N = 13$ is the optimal choice in our final design. Besides, we also analyze the training steps and observe the "sudden converge" phenomenon during the experiment. Refer to C.1 for more details.

Figure 5: The ablation study of ControlNet-UNet and ControlNet-Transformer. ControlNet-Transformer yields much better results than ControlNet-UNet. The controllability of ControlNet-Transformer increases as the number of copy blocks increases.

3.4. 1024px Results

Building upon the powerful text-to-image generation framework of PixArt, our proposed PixArt-ControlNet extends these capabilities to produce high-resolution images with a granular level of control. This is vividly demonstrated in the detailed visualizations presented in Fig. 9 and Fig. 10. Upon closer inspection of these figures, it is apparent that PixArt-ControlNet can exert precise control over the geometric composition of the resultant images, achieving fidelity down to individual strands of hair.

4. Conclusion

In this paper, we present PixArt-δ, a better text-to-image generation model integrating Latent Consistency Models (LCM)for 4-step sampling acceleration while maintaining high quality. We also propose Transformer-based ControlNet, designed specifically for Transformer architectures, enabling precise control over generated images. Through extensive experiments, we demonstrate PixArt-δ's faster sampling and ControlNet-Transformer's effectiveness in high-resolution and controlled image generation. Our model can generate high-quality 1024px and fine-grained controllable images in 1 second. PixArt-δ pushes the state-of-the-art in faster and more controlled image generation, unlocking new capabilities for real-time applications.

References

Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., et al. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023.

Chen, T. On the importance of noise scheduling for diffusion models. *arXiv preprint arXiv:2301.10972*, 2023.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Hoogeboom, E., Heek, J., and Salimans, T. simple diffusion: End-to-end diffusion for high resolution images. *arXiv preprint arXiv:2301.11093*, 2023.

Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *arXiv*, 2017.

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.

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. In *arXiv*, 2023.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

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

von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., and Wolf, T. Diffusers: State-of-the-art diffusion models, 2023. URL https://huggingface.co/docs/diffusers/main/en/api/pipelines/pixart#inference-with-under-8gb-gpu-vram?

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models, 2023.

A. Background

A.1. Consistency Model

Consistency Model (CM) and Latent Consistency Model (LCM) have made significant advancements in the field of generative model acceleration. CM, introduced by (Song et al., 2023) has demonstrated its potential to enable faster sampling while maintaining the quality of generated images on ImageNet dataset (Deng et al., 2009). A key ingredient of CM is trying to maintain the self-consistency property during training (consistency mapping technique), which allows for the mapping of any data point on a Probability Flow Ordinary Differential Equation (PF-ODE) trajectory back to its origin.

LCM, proposed by (Luo et al., 2023a), extends the success of CM to the current most challenging and popular LDMs, Stable Diffusion (Rombach et al., 2022) and SD-XL (Podell et al., 2023) on Text-to-Image generative task. LCM accelerates the reverse sampling process by directly predicting the solution of the augmented PF-ODE in latent space. LCM combines several effective techniques (e.g, One-stage guided distillation, Skipping-step technique) to achieve remarkable rapid inference speed on Stable Diffusion models and fast training convergence. LCM-LoRA (Luo et al., 2023b), training LCM with the LoRA method (Hu et al., 2021), demonstrates strong generalization, establishing it as a universal Stable Diffusion acceleration module. In summary, CM and LCM have revolutionized generative modeling by introducing faster sampling techniques while preserving the quality of generated outputs, paving the way for real-time generation applications.

A.2. ControlNet

ControlNet (Zhang et al., 2023) demonstrates superior control over text-to-image diffusion models' outputs under various conditions (e.g., canny edge, open-pose, sketch). It introduces a special structure, a trainable copy of UNet, that allows for the manipulation of input conditions, enabling control over the overall layout of the generated image. During training, ControlNet freezes the origin text-to-image diffusion model and only optimizes the trainable copy. It integrates the outputs of each layer of this copy by skip-connections into the original UNet using "zero convolution" layers to avoid harmful noise interference.

This innovative approach effectively prevents overfitting while preserving the quality of the pre-trained UNet models, initially trained on an extensive dataset comprising billions of images. ControlNet opens up possibilities for a wide range of conditioning controls, such as edges, depth, segmentation, and human pose, and facilitates many applications in controlling image diffusion models.

B. LCM in PixArt-δ

B.1. Training Details

As discussed in Sec. 2.1, we conduct our experiments in two resolution settings, 512×512 and 1024×1024, utilizing a high-quality internal dataset with 120K images. We smoothly train the models in both resolutions by leveraging the multi-scale image generation capabilities of PixArt-α, which supports 512px and 1024px resolutions. For both resolutions, PixArt-δ yields impressive results before reaching 5K iterations, with only minimal improvements observed thereafter. The training is executed on 2 V100 GPUs with a total batch size of 24, a learning rate of 2e-5, EMA rate $\mu = 0.95$, and using AdamW optimizer (Loshchilov & Hutter, 2017). We employ DDIM-Solver (Song et al., 2023) and a skipping step $k = 20$ (Luo et al., 2023b) for efficiency. As noted in Sec. 2.1 and illustrated in Fig. 3, modifications are made to the original LCM scheduler to accommodate differences between the pre-trained PixArt-α and Stable Diffusion models. Following the PixArt-α approach, we alter the β_t in the diffusion process from a scaled linear to a linear curve, adjusting β_{t_0} from 0.00085 to 0.0001, and β_{t_T} from 0.012 and to 0.02 at the same time. The guidance scale ω_{fix} is set to 4.5, identified as optimal in PixArt-α. While omitting the Fourier embedding of ω in LCM during training, both PixArt-α and PixArt-δ maintain identical structures and trainable parameters. This allows us to initialize the consistency function $\boldsymbol{f_\theta}(\hat{\boldsymbol{z}}, \omega_{fix}, \boldsymbol{c}, t_n)$ with the same parameters as the teacher diffusion model (PixArt-α) without compromising performance. Building on the success of LCM-LoRA (Luo et al., 2023b), PixArt-δ can further easily integrate LCM-LoRA, enhancing its adaptability for a more diverse range of applications.

Algorithm 1 PixArt - Latent Consistency Distillation (LCD)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, ODE solver $\Psi(\cdot,\cdot,\cdot,\cdot)$, distance metric $d(\cdot,\cdot)$, EMA rate μ, noise schedule $\alpha(t), \sigma(t)$, guidance scale ω_{fix}, skipping interval k, and encoder $E(\cdot)$
Encoding training data into latent space: $\mathcal{D}_z = \{(\boldsymbol{z}, \boldsymbol{c}) | \boldsymbol{z} = E(\boldsymbol{x}), (\boldsymbol{x}, \boldsymbol{c}) \in \mathcal{D}\}$
$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$
repeat
  Sample $(\boldsymbol{z}, \boldsymbol{c}) \sim \mathcal{D}_z$, $n \sim \mathcal{U}[1, N-k]$
  Sample $\boldsymbol{z}_{t_{n+k}} \sim \mathcal{N}(\alpha(t_{n+k})\boldsymbol{z}; \sigma^2(t_{n+k})\mathbf{I})$
  $\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega_{fix}} \leftarrow \boldsymbol{z}_{t_{n+k}} + (1+\omega_{fix})\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \boldsymbol{c}) - \omega_{fix}\Psi(\boldsymbol{z}_{t_{n+k}}, t_{n+k}, t_n, \varnothing)$
  $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \Psi) \leftarrow d(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z}_{t_{n+k}}, \omega_{fix}, \boldsymbol{c}, t_{n+k}), \boldsymbol{f}_{\boldsymbol{\theta}^-}(\hat{\boldsymbol{z}}_{t_n}^{\Psi,\omega_{fix}}, \omega_{fix}, \boldsymbol{c}, t_n))$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-)$
  $\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu\boldsymbol{\theta}^- + (1-\mu)\boldsymbol{\theta})$
until convergence

C. ControlNet in PixArt-δ

C.1. Convergence

As described in Fig. 12, we analyze the effect of training steps. The experiment is conducted on ControlNet-Transformer ($N = 13$). From our observation, the convergence is very fast, with most edges achieving satisfactory results at around 1,000 training steps. Moreover, we note a gradual improvement in results as the number of training steps increases, particularly noticeable in enhancing the quality of outline edges for human faces and bodies. This observation underscores the efficiency and effectiveness of ControlNet-Transformer.

We observe a similar "sudden converge" phenomenon in our model, as also observed in the original ControlNet work, where it "suddenly" adapts to the training conditions. Empirical observations indicate that this phenomenon typically occurs between 300 to 1,000 steps, with the convergence steps being influenced by the difficulty level of the specified conditions. Simpler edges tend to converge at earlier steps, while more challenging edges require additional steps for convergence. After "sudden converge", we observe an improvement in details as the number of steps increases.

Figure 6: Example of "Sudden Converge" during PixArt-ControlNet training. We empirically observe it happens before 1000 iterations.

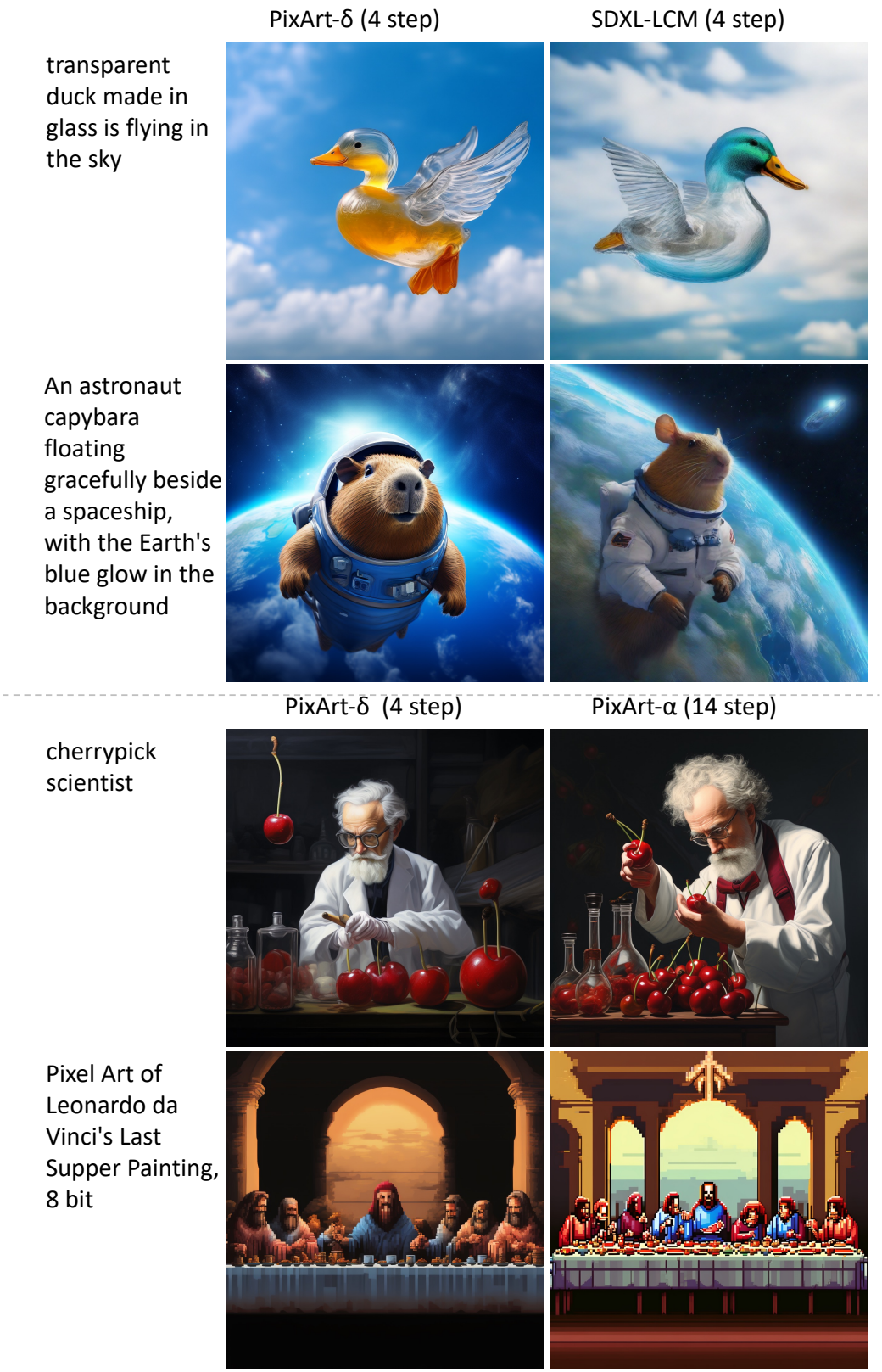


Figure 7: Examples of generated outputs. In the top half, the comparison is between PIXART-δ and SDXL-LCM, with 4 sampling steps. In the bottom half, the comparison involves PIXART-δ and PIXART-α (teacher model, using DPM-Solver with 14 steps).

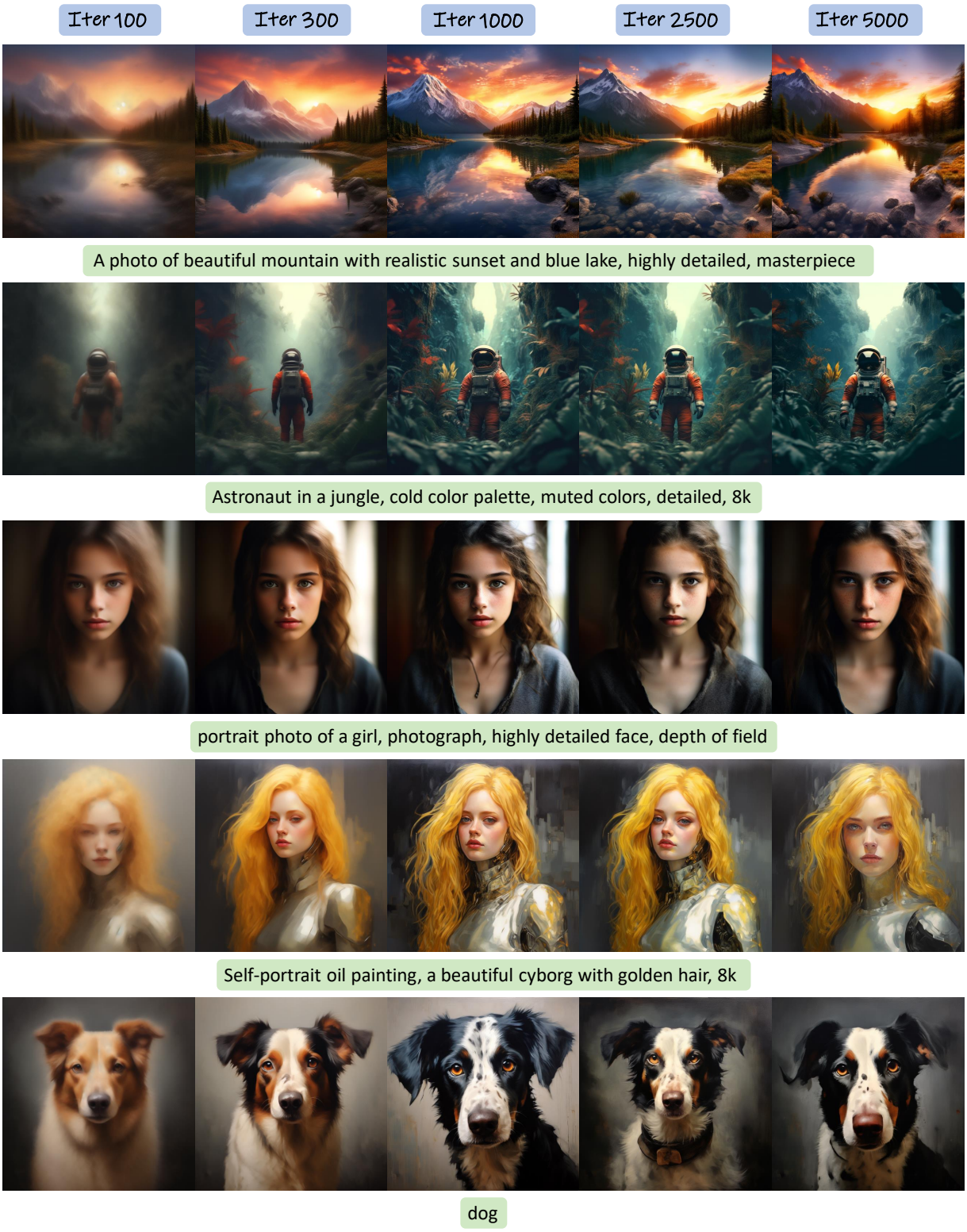


Figure 8: The 4-step inference samples generated by PixArt-δ demonstrate fast convergence in LCD training on 2 V100 GPUs with a total batch size of 24. Remarkably, the complete fine-tuning process requires less than 24GB of GPU memory, making it feasible on most contemporary consumer-grade GPUs.

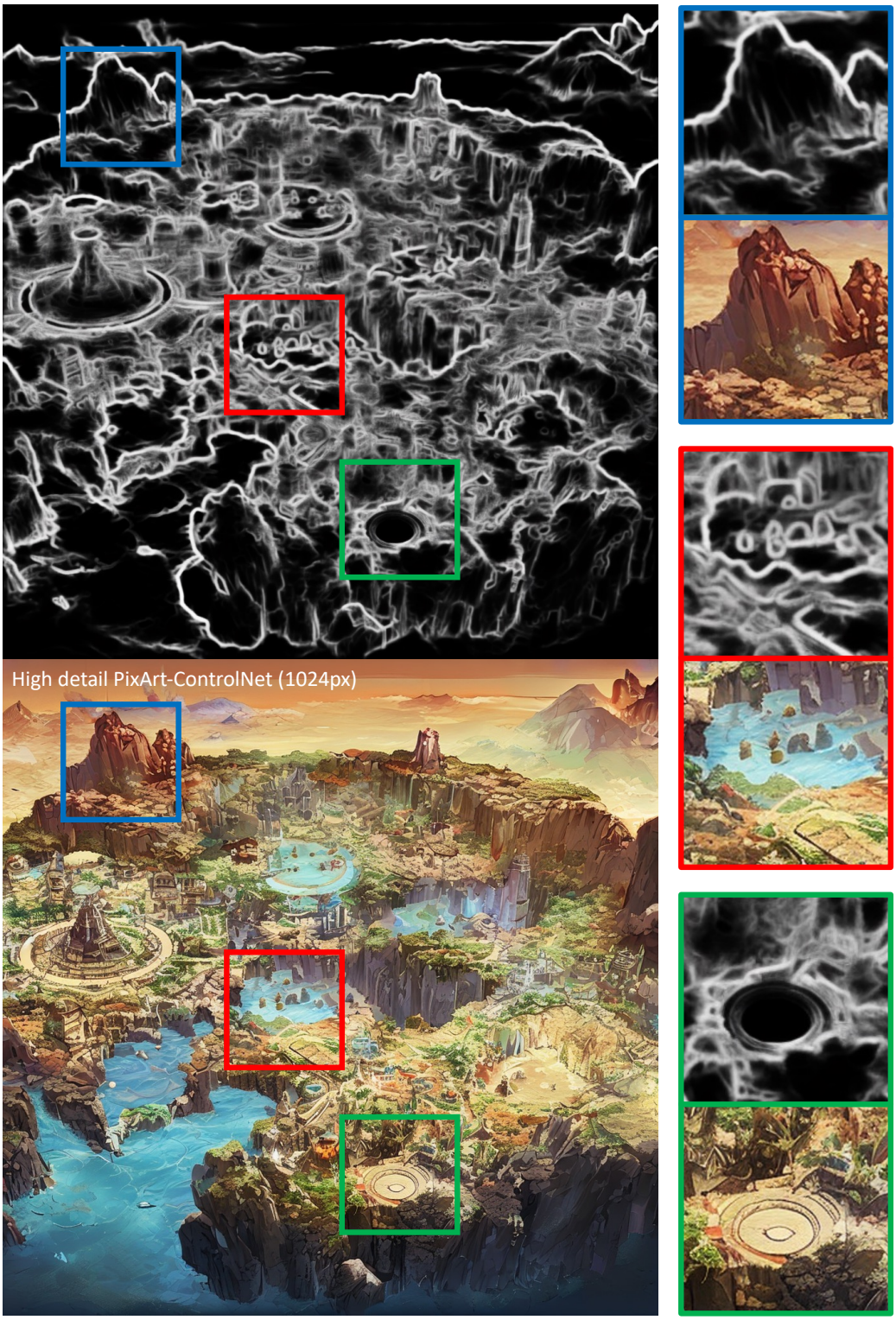


Figure 9: High-resolution and fine-grained controllable image generation. The output is generated with the prompt "the map of the final fantasy game's main island, in the style of hirohiko araki, raymond swanland, monumental murals, mosaics, naturalistic rendering, vorticism, use of earth tones."

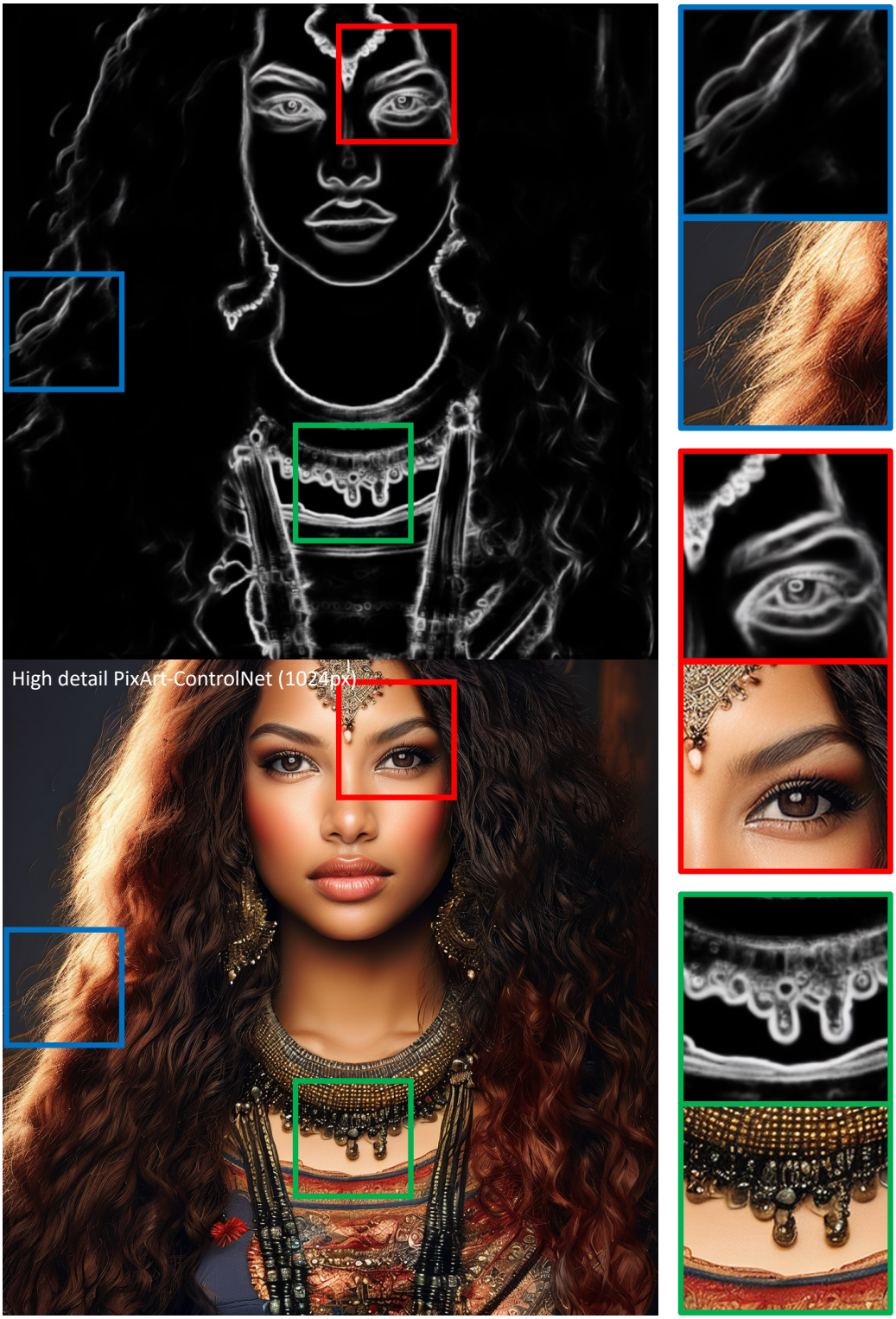


Figure 10: High-resolution and fine-grained controllable image generation. The output is generated with the prompt "Multicultural beauty. Women of different ethnicity - Caucasian, African, Asian and Indian."

Figure 11: More examples of our PixArt-ControlNet generated images.

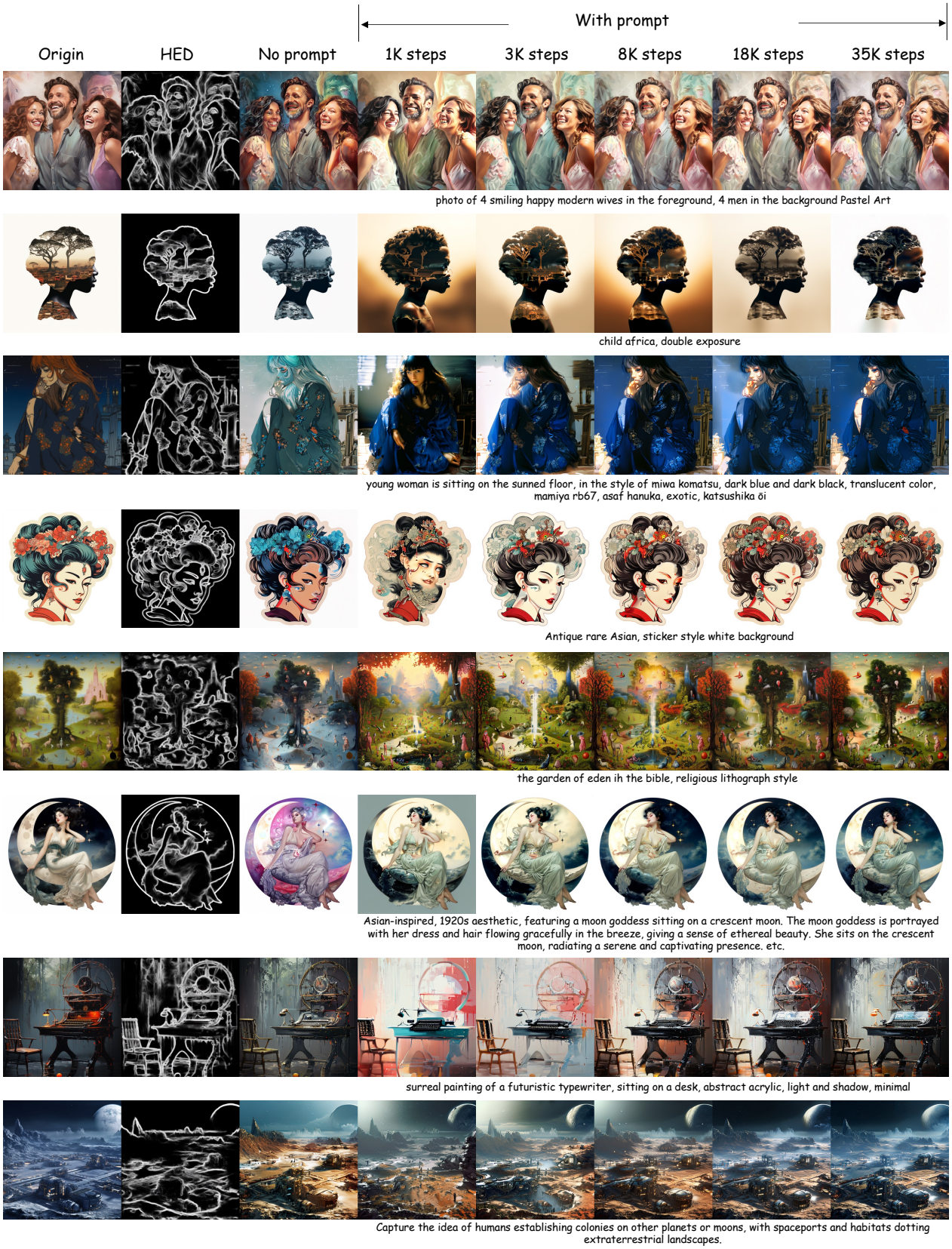


Figure 12: The influence of training steps. The convergence is fast, with details progressively improving and aligning more closely with the HED edge map as the training steps increase.