Appendix for HiDiffusion: Unlocking Higher-Resolution Creativity and Efficiency in Pretrained Diffusion Models

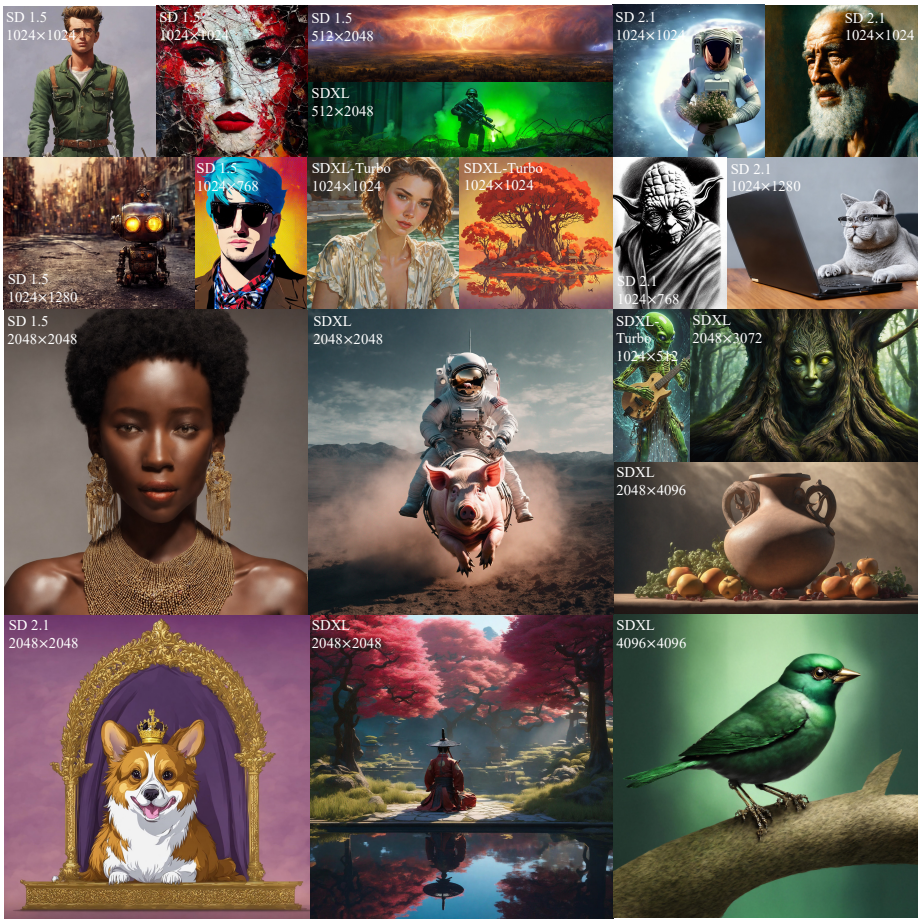


Fig. 1: Select HiDiffusion samples for various diffusion models, resolutions, and aspect ratios. HiDiffusion enables pretrained diffusion models to generate higher-resolution images surpassing the training image size without further training or fine-tuning and can effectively accelerate the inference. Best viewed when zoomed in.

In the appendix, we present the following details associated with HiDiffusion:
- Visualization of feature duplication across inference steps.
- More ablations about the components of HiDiffusion, including the effect of RAU-Net and MSW-MSA, the RAU operation, the position of RAD and RAU, the Switching Threshold, the window size of MSW-MSA.
- Details about SD 2.1, SDXL, SDXL-Turbo settings.
- Details about extreme resolutions (2048×2048 for SD 1.5, SD 2.1, 4096×4096 for SDXL).
- Comparison to training at higher resolution.
- Extensions to image-to-image task.
- More visualization results, including comparisons to diffusion acceleration and high-resolution synthesis methods.

A Feature Duplication across Inference Step

When directly inferring to generate higher-resolution images using pretrained diffusion models, we observed the feature duplication phenomenon at the 30th inference step. This section presents feature visualization across different inference steps to demonstrate that feature duplication arises at almost every inference step, as shown in Fig. 2. Even when the input latent is extremely noisy in the early denoising stages, such as the 1st inference step, direct inference still leads to conspicuous feature duplication, as shown in the UB 3 output of the 1st inference step. When the input latent is less noisy, for instance, at the 45th inference step, more severely pronounced feature duplication emerges. The feature duplication impacts the trajectory of image generation, ultimately causing object duplication in the final output image (the updated latent at the 50th inference step). Compared to direct inference, our HiDiffusion effectively alleviates feature duplication at each inference step and can generate reasonable higher-resolution images in a tuning-free way.

B More Ablation Results

B.1 The Effect of RAU-Net and MSW-MSA

We have analyzed the impact of RAU-Net and MSW-MSA in the main paper. Here we provide the qualitative comparison of all possible combinations, as shown in Fig. 3.

B.2 The RAD Operation

In the main paper, RAD is achieved by altering the stride, padding, and dilation of the original downsampler's convolution. Alternatively, We can add an extra adaptive pooling and keep the convolution unchanged, which can be written as:

$$\mathcal{R}(\mathcal{C}_{3,1,2,1}(x), \alpha) = ada\_pool(\mathcal{C}_{3,1,2,1}(x), \frac{\alpha}{2}). \tag{1}$$

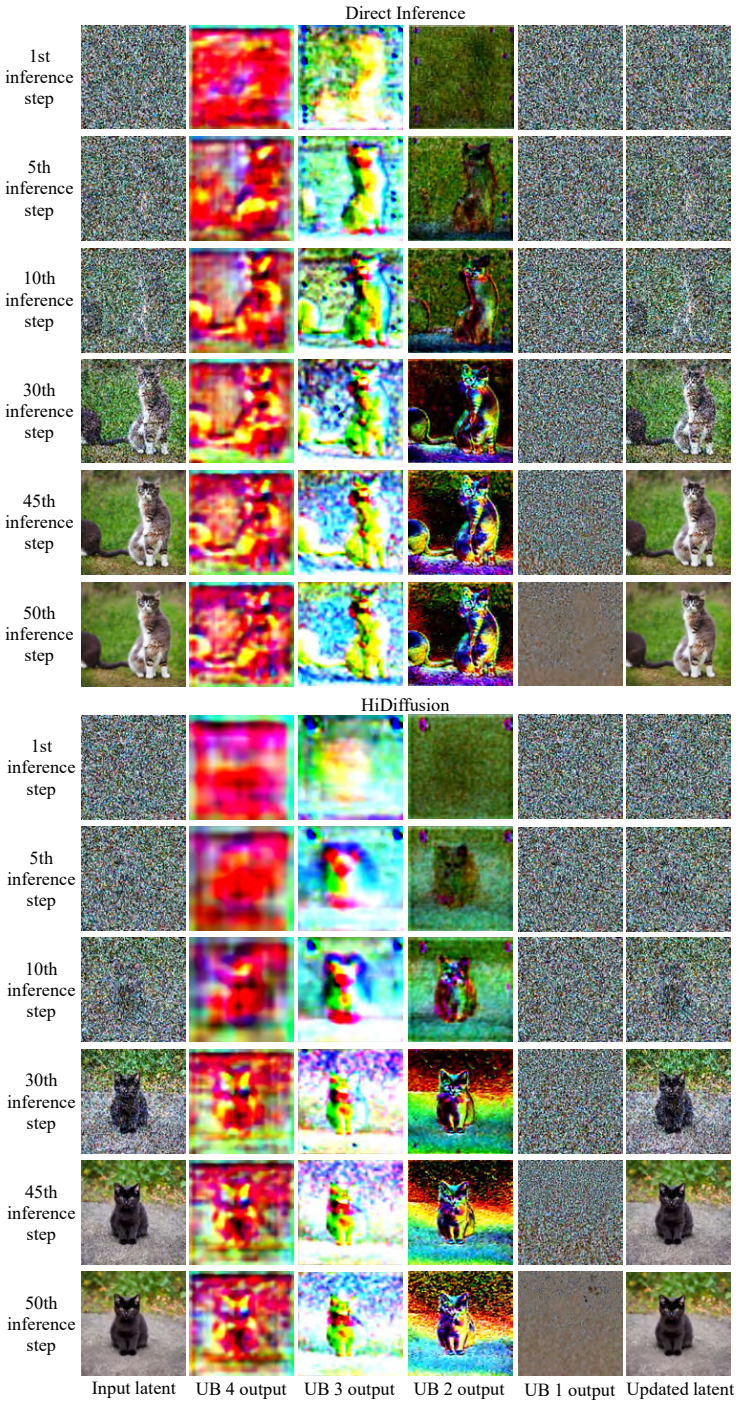


Fig. 2: The feature map visualization across different inference steps based on SD 1.5. The image resolution is 1024×1024 and we adopt 50 DDIM steps.

Fig. 3: The effect of RAU-Net and MSW-MSA based on SD 1.5. The resolution is 1024×1024.

This method can also achieve the goal of resolution-aware downsampling. for the 1024×1024 resolution generation, α is set as 4. In this section, we investigate which methods can generate higher-quality images. We present quantitative comparison in Tab. 1. Compared to the additional pooling operation, the method used in the main paper exhibits better performance in both FID and CLIP-Score.

Method	FID↓	CLIP-Score↑
Adaptive pooling	26.73	0.304
Larger stride	21.81	0.307

Table 1: Quantitative evaluation of two variants of resolution-aware operation in zero-shot text-guided image synthesis on ImageNet based on SD 1.5. The resolution is 1024×1024.

B.3 The impact of the position of RAD and RAU

Our main idea is to introduce RAD and RAU to dynamically downsample the feature map. We insert the RAD and RAU into Block 1, Block 2, and Block 3 respectively to examine the impact of the Resolution-aware sampler at different locations, as shown in Tab. 2. There is a minor quantitative metric difference between different locations. However, we visually observe that incorporating RAD and RAU in Block 1 can better mitigate object duplication, as illustrated in Fig. 4.

Fig. 4: Qualitative comparison between different positions of RAD and RAU based on SD 1.5. The resolution is 1024×1024.

Position	Block 1	Block 2	Block 3
FID	21.81	20.84	21.26
CLIP-Score	0.307	0.307	0.305

Table 2: Quantitative evaluation of the position of RAD and RAU in zero-shot text-guided image synthesis on ImageNet based on SD 1.5. The resolution is 1024×1024.

B.4 The Switching Threshold

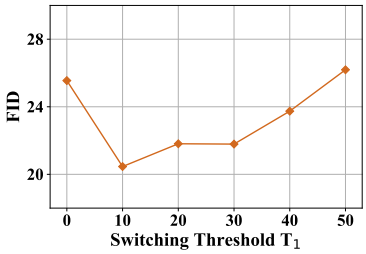


Fig. 5: Quantitative evaluation of different T_1 in zero-shot text-guided image synthesis on ImageNet based on SD 1.5. The resolution is 1024×1024.

The Switching Threshold determines when to switch from RAU-Net to vanilla U-Net. We explore the impact of different thresholds on the performance of HiDiffusion. The quantitative results are shown in Fig. 5. T_1=0 indicates that RAU-Net is not utilized, while T_1=50 indicates that RAU-Net is used in the entire denoising process. When T_1 is between 10 and 40, we improve the performance in metric evaluation. The qualitative comparison demonstrates that T_1 ranging from 10 to 50 can effectively alleviate object duplication, with $T_1 = 20$ yielding the optimal performance, as shown in Fig. 6. Therefore, we select $T_1 = 20$ as the default setting.

B.5 The Window Size of MSW-MSA

The window size determines the receptive field of self-attention. We compare the performance from the small window size proposed in Swin Transformer to our

Fig. 6: Qualitative comparison between different T_1 based on SD 1.5. The resolution is 1024×1024.

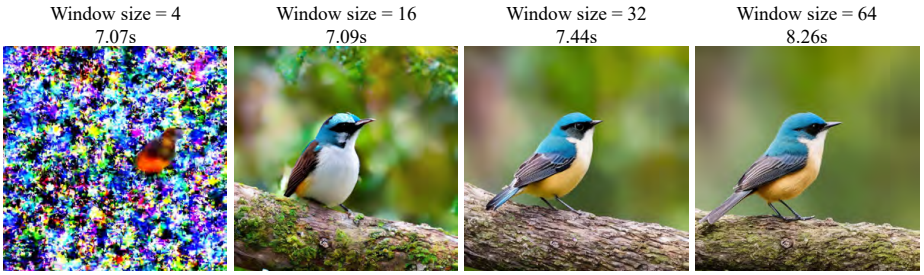


Fig. 7: Qualitative comparison between different window sizes based on SD 1.5. The resolution is 1024×1024.

proposed large window size based on SD 1.5, as shown in Tab. 3. As the window size gradually increases, the performance improves. We achieve the optimal balance between efficiency and image quality when the window size is half the height and width of the feature map. The qualitative results are shown in Fig. 7.

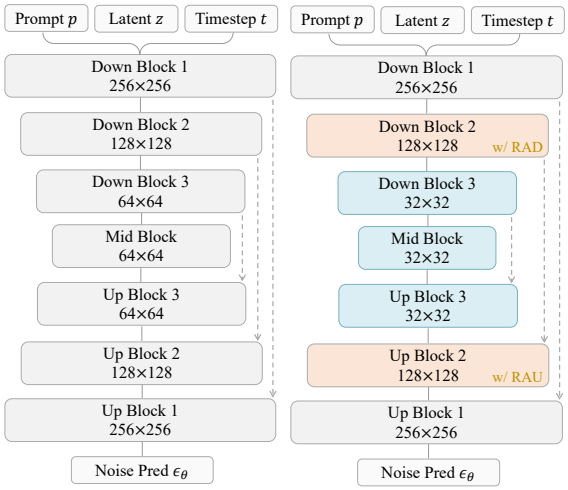
C Details about Other Models settings.

As SD 1.5 and SD 2.1 share the same U-Net architecture, the settings of SD 2.1 are consistent with those of SD 1.5.

An illustrative comparison of the vanilla SDXL U-Net and RAU-Net for SDXL in the context of generating 2048×2048 resolution images is presented in Fig. 8. We incorporate RAD and RAU in Block 2 and set $\alpha = \beta = 4$ to match the deep blocks of U-Net. In contrast to SD 1.5 and SD 2.1's U-Net, Down Block 1 and Up Block 1 of SDXL only consist of two and three ResNet blocks, respectively. If we choose to incorporate the RAD and RAU in Block 1,

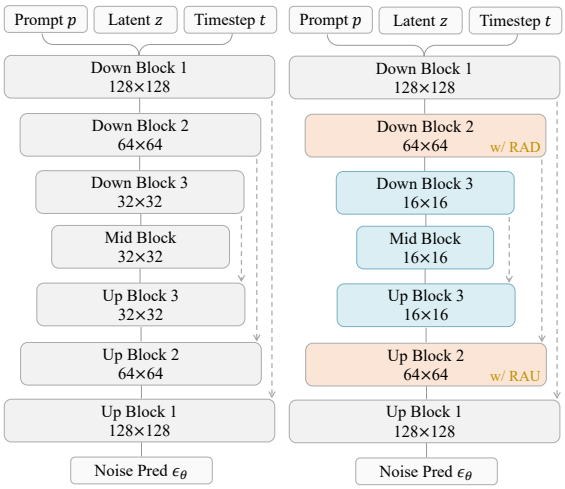
Window size	4	16	32	64
Latency (s)	7.07	7.09	7.44	8.26
FID	417.15	53.02	22.37	21.81
CLIP-Score	0.225	0.295	0.307	0.307

Table 3: Quantitative evaluation of the window size in zero-shot text-guided image synthesis on ImageNet based on SD 1.5. The resolution is 1024×1024.



(a) Vanilla U-Net. **(b)** HiDiffusion RAU-Net.

Fig. 8: Comparison between vanilla SDXL's U-Net and our proposed HiDiffusion RAU-Net for SDXL. Parameters in all blocks are frozen. The main difference lies in the blue Blocks (differ in the dimensions of feature map) and orange Blocks (Our proposed RAD and RAU modules are incorporated into Block 2).



(a) Vanilla U-Net. **(b)** HiDiffusion RAU-Net.

Fig. 9: Comparison between vanilla SDXL-Turbo's U-Net and our proposed HiDiffusion RAU-Net for SDXL-Turbo. Parameters in all blocks are frozen. The main difference lies in the blue Blocks (differ in the dimensions of feature map) and orange Blocks (Our proposed RAD and RAU modules are incorporated into Block 2.).

the ResNet Blocks in Block 1 are insufficient to effectively handle the resolution change caused by the interpolation function in RAD, resulting in the synthesis of blurry images. We present the qualitative comparison between inserting RAD

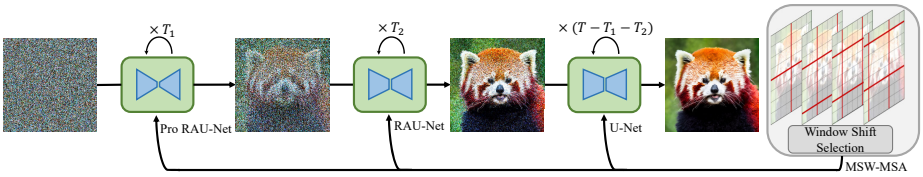


Fig. 10: The framework of image synthesis with extreme resolution (2048×2048 for SD 1.5 and SD 2.1, 4096×4096 for SDXL). Pro RAU-Net: progressive RAU-Net.

Fig. 11: 2048×2048 resolution comparison between inserting resolution-aware samplers into Block 1 and Block 2 based on SDXL.

and RAU in Block 1 and inserting RAD and RAU in Block 2 in Fig. 11. In the experiment of the main paper, We set $T_1 = 20$ for 50 DDIM steps. The classifier-free guidance scale is 7.5. Since Block 1 of SDXL U-Net does not contain self-attention, we incorporate MSW-MSA into Block 2. We set the window size as $(64, 64)$. The predefined set of shift strides is $\{(0, 0), (16, 16), (32, 32), (48, 48)\}$. For 4096×4096 resolution generation, please refer to Sec. D.

The U-Net architecture of SDXL-Turbo and SDXL are very similar, except for the differences in input and output dimensions, as shown in Fig. 9. We introduce the setting of SDXL-Turbo in brief. We incorporate the RAD and RAU into Block 2. The inference step is 4 and we set $T_1 = 2$. Classifier-free guidance is not used. We set the window size as $(32, 32)$. The predefined set of shift strides is $\{(0, 0), (8, 8), (16, 16), (24, 24)\}$.

D Details about extreme resolutions

For SD 1.5 and SD 2.1, generating images with 2048×2048 resolution is a significant challenge, considering that this resolution is already 16 times the training image resolution. RAU-Net can generate images with 2048×2048 resolution by simply setting $\alpha = \beta = 8$, as shown in Fig. 12b. However, $\beta = 8$ implies that RAU upsamples the feature map by a factor of 8 using an interpolation function. This abrupt resolution change brought by interpolation leads to the generation of blurry images, as illustrated in Fig. 14a. To tackle the issue, we adopt a progressive variant of RAU-Net, as shown in Fig. 12c. We incorporate RAU and RAD with $\alpha = \beta = 4$ into Block 1 and Block 2, respectively. This allows the feature map to gradually align with the deep blocks of U-Net, thus circumventing the blurriness issue caused by a large interpolation factor. For 4096×4096

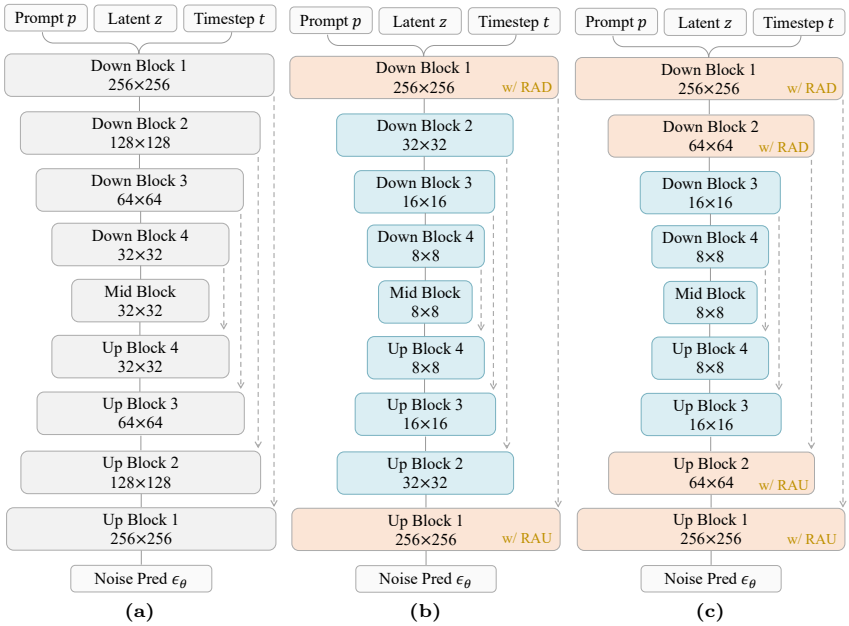


Fig. 12: U-Net variants of SD 1.5 and SD 2.1. (a) Vanilla U-Net. (b) RAU-Net. (c) Progressive RAU-Net.

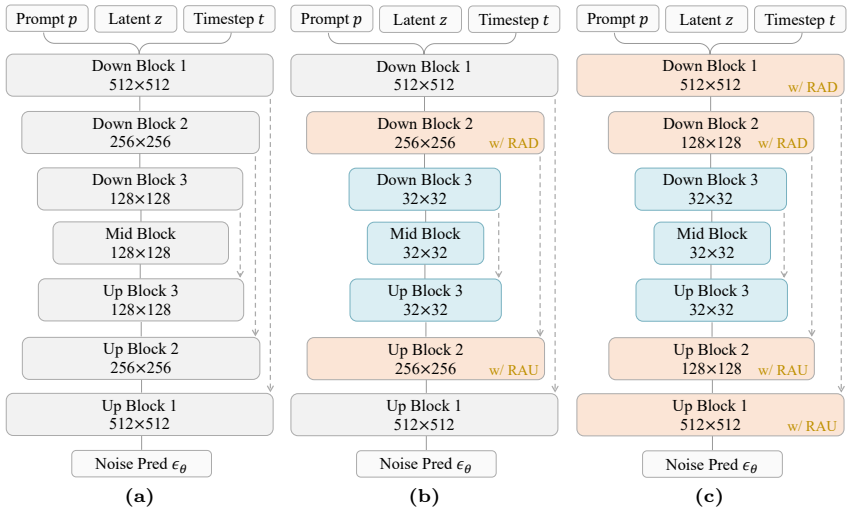


Fig. 13: U-Net variants of SDXL. (a) Vanilla U-Net. (b) RAU-Net. (c) Progressive RAU-Net. The parameter settings of (c) are same with Fig. 12.

(a) (b)

Fig. 14: 2048×2048 resolution samples generated by (a) Directly set $\alpha = 8$ in Block 1 of RAU-Net. (b) The final progressive method. The diffusion model version is SD 1.5.

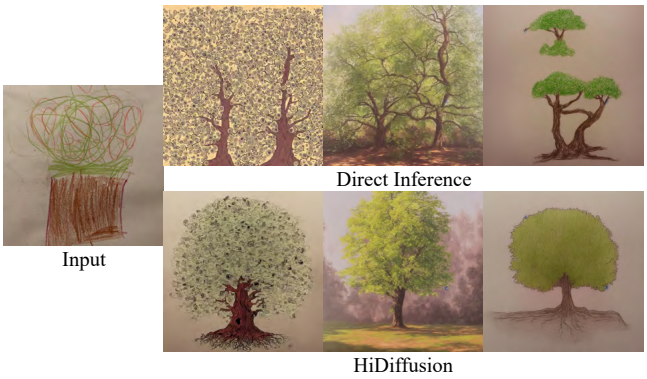


Fig. 15: SDEdit task of 1024×1024 resolution based on SD 1.5.

Fig. 16: ControlNet task of 1024×1024 resolution based on SD 1.5.

resolution generation of SDXL, we also adopt progressive RAU-Net, as shown in Fig. 13c. We incorporate RAU and RAD with $\alpha = \beta = 4$ into Block 1 and Block 2, respectively.

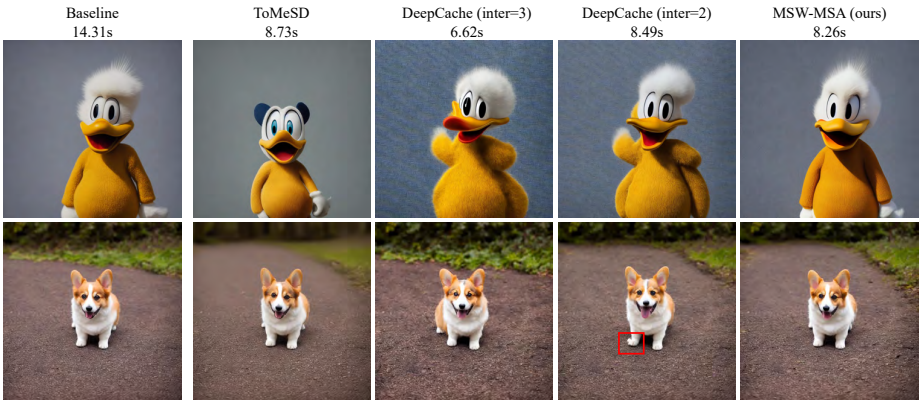


Fig. 17: The qualitative comparison between different diffusion acceleration methods based on SD 1.5. The resolution is 1024×1024. The baseline is SD 1.5 with RAU-Net.

As described in the main paper, matching the feature map size with the deep blocks of U-Net can generate coherent object structures while potentially affecting image details. Therefore, when generating images with extreme resolution, we gradually reduce the usage of resolution-aware samplers throughout the denoising process for finer image detail. Specifically, we employ Progressive RAU-Net in the early stage, followed by RAU-Net in the middle stage, and finally vanilla U-Net in the later stage. We establish two thresholds T_1 and T_2: when denoising steps $t < T_1$, We use progressive RAU-Net; when $T_1 \leq t \leq T_2$, We use RAU-Net ; when $t > T_2$, vanilla U-Net is used. We present the framework in Fig. 10 and generated samples in Fig. 14b. In the experiment of the main paper, We set $T_1 = 15$ and $T_2 = 35$ for 50 DDIM steps. We incorporate MSW-MSA into Block 1 for SD 1.5 and SD 2.1, and into Block 2 for SDXL. We set the window size as $(128, 128)$. The predefined set of shift strides is $\{(0, 0), (32, 32), (64, 64), (96, 96)\}$. The classifier-free guidance scales of SD 1.5, SD 2.1, and SDXL are all 7.5.

E Comparison to Training at Higher Resolution

Method	Resolution	Latency (s) ↓	FID ↓	pFID ↓	CLIP ↑	Params ↓
SDXL	1024 × 1024	15.29	**20.30**	31.20	**0.314**	3.5B
SD 1.5 + HiDiffusion		**8.26(1.85×)**	21.81	**30.86**	0.307	**1.1B**

Table 4: Comparison with training at the higher resolution.

Tab. 4 shows the comparison between HiDiffusion with SD 1.5 (trained with 512×512 images) and SDXL (trained with 1024×1024 images). HiDiffusion achieves comparable image quality and higher efficiency compared to models trained on datasets with 1024×1024 resolution.

F Image-to-image Task

HiDiffusion also works well in image-to-image tasks including SDEdit and ControlNet, as shown in Figs. 15 and 16.

G More Visualization Results

G.1 Comparison to Diffusion Acceleration Methods

In the main paper, We have demonstrated the effectiveness of our MSW-MSA compared to other diffusion acceleration methods. Here we provide the qualitative comparison between MSW-MSA and other methods, as shown in Fig. 17. Compared to other methods, the generated images by our method are more consistent with the baseline. Furthermore, we surpass other methods in terms of details and local features.

G.2 Comparison to High-Resolution Synthesis Method

We present more comparison results with LDM-SR, ScaleCrafter, and Demofusion based on SDXL to demonstrate the effectiveness of HiDiffusion, as shown in Figs. 18 and 19. HiDiffusion outperforms ScaleCrafter and DemoFusion in local details, while compared to LDM-SR, HiDiffusion achieves comparable or even better performance.

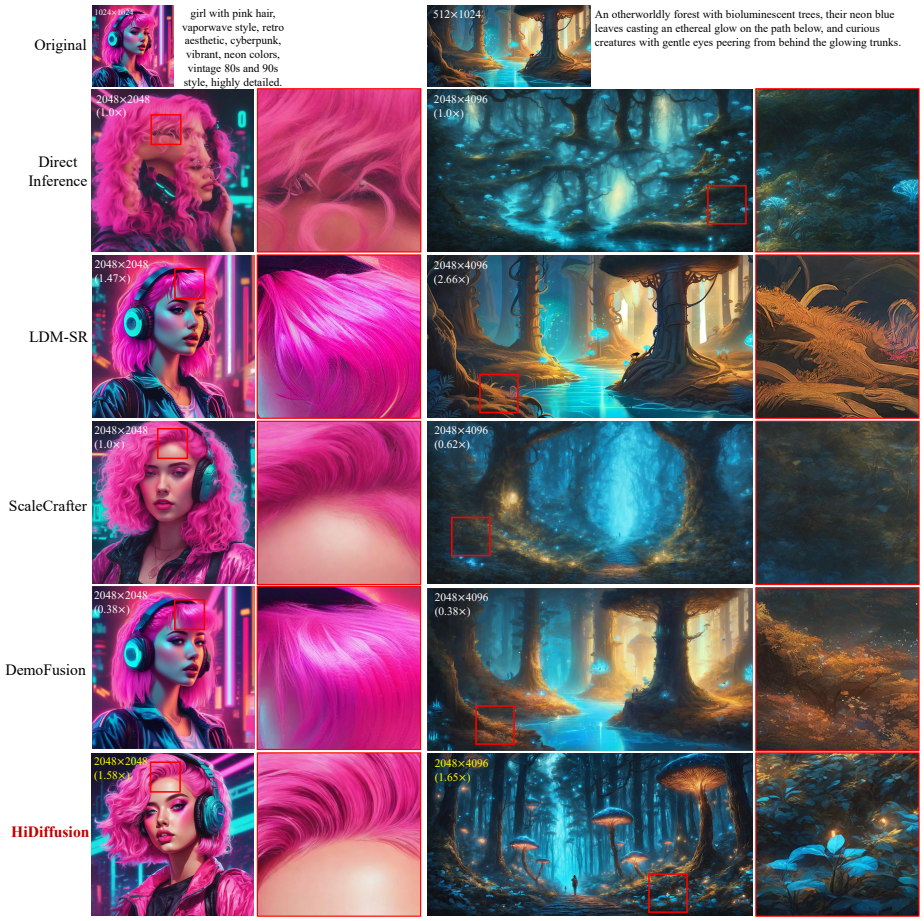


Fig. 18: More qualitative comparison with other methods based on SDXL. The input prompt is located to the right of the original image. The first line of text in the image indicates the image resolution, while the second line indicates the inference speed relative to direct inference. Best viewed when zoomed in.

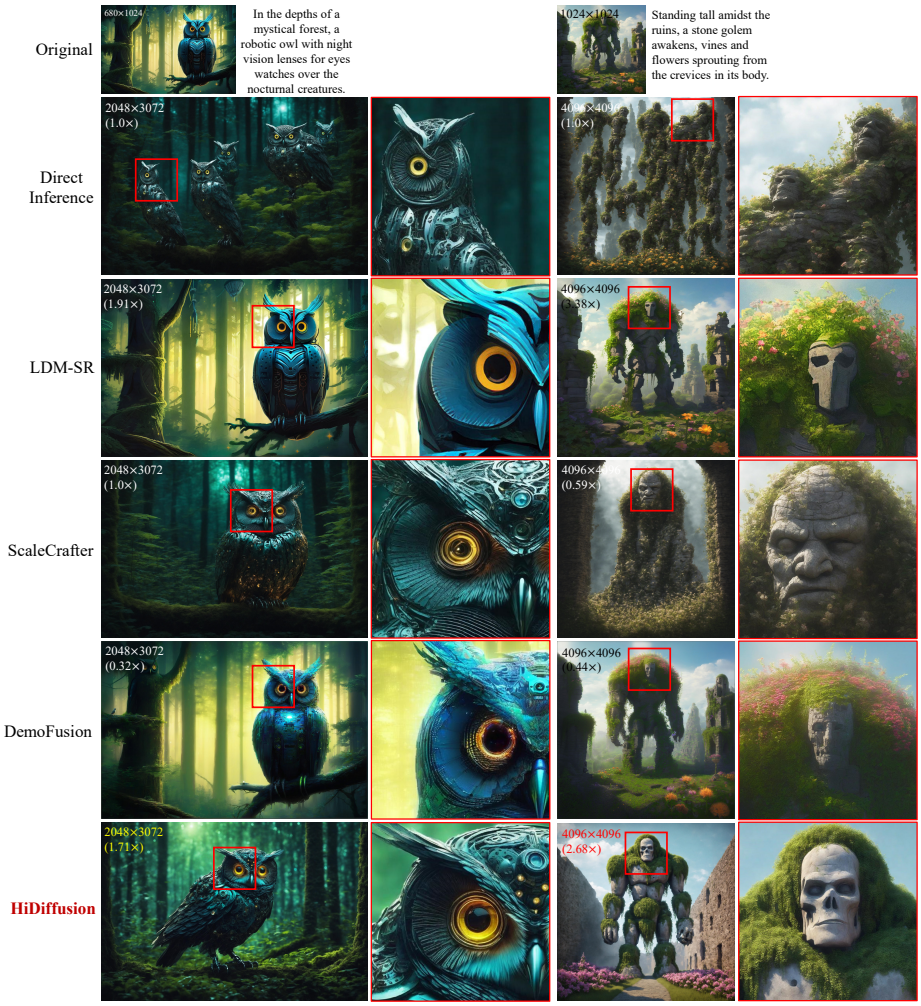


Fig. 19: More qualitative comparison with other methods based on SDXL. The input prompt is located to the right of the original image. The first line of text in the image indicates the image resolution, while the second line indicates the inference speed relative to direct inference. Best viewed when zoomed in.