# Supplementary Material for FouriScale

Linjiang Huang[1,3⋆], Rongyao Fang[1⋆], Aiping Zhang[4], Guanglu Song[5]
Si Liu[6], Yu Liu[5], and Hongsheng Li[1,2,3] ✉

[1] CUHK MMLab [2] Shanghai AI Laboratory [3] CPII under InnoHK
[4] Sun Yat-Sen University [5] SenseTime Research [6] Beihang University
ljhuang524@gmail.com; {rongyaofang@link, hsli@ee}.cuhk.edu.hk

**Abstract.** In this supplementary material, we provide the proof corresponding to Theorems and Lemmas in the main manuscript. Besides, we report the hyperparameter settings for various SD models and resolutions. The definition of low-pass filters is also included. Furthermore, we offer additional quantitative comparisons and qualitative results.

**Keywords:** Diffusion Model · Training Free · High-Resolution Synthesis

## 1 Proof

### 1.1 Proof of Theorem 1

Let's consider $f(x)$ as a one-dimensional signal. Its down-sampled counterpart is represented by $f'(x) = \text{Down}_s(f)$. To understand the connection between $f'(x)$ and $f(x)$, we base our analysis on their generated continuous signal $g(x)$, which is produced using a particular sampling function. It's important to note that the sampling function $sa_{\Delta T}(x)$ is characterized by a series of infinitely spaced impulse units, with each pair separated by intervals of $\Delta T$:

$$sa(x, \Delta T) = \sum_{n=-\infty}^{\infty} \delta(x - n\Delta T). \tag{1}$$

Based on Eq. (1), $f(x)$ and $f'(x)$ can be formulated as

$$\begin{aligned} f(x) &= g(x)sa(x, \Delta T), \\ f'(x) &= g(x)sa(x, s\Delta T). \end{aligned} \tag{2}$$

Based on the Fourier transform and the convolution theorem, the spatial sampling described above can be represented in the Fourier domain as follows:

⋆ Equal contribution. ✉ Corresponding author.

$$
\begin{aligned}
F(u) &= G(u) \circledast SA(u, \Delta T) \\
&= \int_{-\infty}^{\infty} G(\tau) SA(u-\tau, \Delta T) d\tau \\
&= \frac{1}{\Delta T} \sum_{n} \int_{-\infty}^{\infty} G(\tau) \delta\left(u-\tau-\frac{n}{\Delta T}\right) d\tau \\
&= \frac{1}{\Delta T} \sum_{n} G\left(u-\frac{n}{\Delta T}\right),
\end{aligned} \tag{3}
$$

where $G(u)$ and $SA(u, \Delta T)$ are the Fourier transform of $g(x)$ and $sa(x, \Delta T)$. From the above Equation, it can be observed that the spatial sampling introduces the periodicity to the spectrum and the period is $\frac{1}{\Delta T}$.

Note that the sampling rates of $f(x)$ and $f'(x)$ are $\Omega_x$ and $\Omega'_x$, the relationship between them can be written as

$$
\Omega_x = \frac{1}{\Delta T}, \quad \Omega'_x = \frac{1}{s\Delta T} = \frac{1}{s}\Omega_x. \tag{4}
$$

With the down-sampling process in consideration, we presume that $f(x)$ complies with the Nyquist sampling theorem, suggesting that $u_{max} < \frac{\Omega_x}{2}$.

Following down-sampling, as per the Nyquist sampling theorem, the entire sub-frequency range is confined to $(0, \frac{\Omega_x}{s})$. The resulting frequency band is a composite of $s$ initial bands, expressed as:

$$
F'(u) = \mathbb{S}(F(u), F(\tilde{u}_1), \ldots, F(\tilde{u}_{s-1})), \tag{5}
$$

where $\tilde{u}_i$ represents the frequencies higher than the sampling rate, while $u$ denotes the frequencies that are lower than the sampling rate. The symbol $\mathbb{S}$ stands for the superposition operator. To simplify the discussion, $\tilde{u}$ will be used to denote $\tilde{u}_i$ in subsequent sections.

(1) In the sub-band, where $u \in (0, \frac{\Omega_x}{2s})$, $\tilde{u}$ should satisfy

$$
\tilde{u} \in \left(\frac{\Omega_x}{2s}, u_{max}\right). \tag{6}
$$

According to the aliasing theorem, the high frequency $\tilde{u}$ is folded back to the low frequency:

$$
\hat{u} = \left|\tilde{u} - (k+1)\frac{\Omega'_x}{2}\right|, \quad k\frac{\Omega'_x}{2} \leq \tilde{u} \leq (k+2)\frac{\Omega'_x}{2} \tag{7}
$$

where $k = 1, 3, 5, \ldots$ and $\hat{u}$ is folded results by $\tilde{u}$.

According to Eq. 6 and Eq. 7, we have

$$
\hat{u} = \frac{a\Omega_x}{s} - \tilde{u} \quad \text{and} \quad \hat{u} \in \left(\frac{\Omega_x}{s} - u_{max}, \frac{\Omega_x}{2s}\right), \tag{8}
$$

where $a = (k+1)/2 = 1, 2, \ldots$. According to Eq. (5) and Eq. (8), we can attain

$$F'(u) = \begin{cases} F(u) & \text{if } u \in (0, \frac{\Omega_x}{s} - u_{max}), \\ \mathbb{S}(F(u), F(\frac{a\Omega_x}{s} - u)) & \text{if } u \in (\frac{\Omega_x}{s} - u_{max}, \frac{\Omega_x}{2s}). \end{cases} \tag{9}$$

According to Eq. (3), $F(u)$ is symmetric with respect to $u = \frac{\Omega_x}{2}$:

$$F(\frac{\Omega_x}{2} - u) = F(u + \frac{\Omega_x}{2}). \tag{10}$$

Therefore, we can rewrite $F(\frac{a\Omega_x}{s} - u)$ as:

$$\begin{aligned} & F(\frac{\Omega_x}{2} - (\frac{\Omega_x}{2} + u - \frac{a\Omega_x}{s})) \\ =& F(\frac{\Omega_x}{2} + (\frac{\Omega_x}{2} + u - \frac{a\Omega_x}{s})) \\ =& F(u + \Omega_x - \frac{a\Omega_x}{s}) \\ =& F(u + \frac{a\Omega_x}{s}) \end{aligned} \tag{11}$$

since $a = 1, 2, \ldots, s-1$. Additionally, for $s = 2$, the condition $u \in (0, \frac{\Omega_x}{s} - u_{max})$ results in $F(u + \frac{\Omega_x}{s}) = 0$. When $s > 2$, the range $u \in (0, \frac{\Omega_x}{s} - u_{max})$ typically becomes non-existent. Thus, in light of Eq. (11) and the preceding analysis, Eq. (9) can be reformulated as

$$F'(u) = \mathbb{S}(F(u), F(u + \frac{a\Omega_x}{s})) \mid u \in (0, \frac{\Omega_x}{2s}). \tag{12}$$

(2) In the sub-band, where $u \in (\frac{\Omega_x}{2s}, \frac{\Omega_x}{s})$, different from (1), $\tilde{u}$ should satisfy

$$\tilde{u} \in (\frac{\Omega_x}{s} - u_{max}, \frac{\Omega_x}{2s}). \tag{13}$$

Similarly, we can obtain:

$$F'(u) = \mathbb{S}(F(\tilde{u}), F(u + \frac{a\Omega_x}{s})) \mid u \in (\frac{\Omega_x}{2s}, \frac{\Omega_x}{s}). \tag{14}$$

Combining Eq. (12) and Eq. (14), we obtain

$$F'(u) = \mathbb{S}(F(u), F(u + \frac{a\Omega_x}{s})) \mid u \in (0, \frac{\Omega_x}{s}), \tag{15}$$

where $a = 1, 2, \ldots, s-1$.

### 1.2 Proof of Lemma 1

Based on Eq. (3), it can be determined that the amplitude of $F'$ is $\frac{1}{s}$ times that of $F$. Hence, $F'(u)$ can be expressed as:

$$F'(u) = \frac{1}{s}F(u) + \sum_a \frac{1}{s}F\left(u + \frac{a\Omega_x}{s}\right) \mid u \in \left(0, \frac{\Omega_x}{s}\right). \tag{16}$$

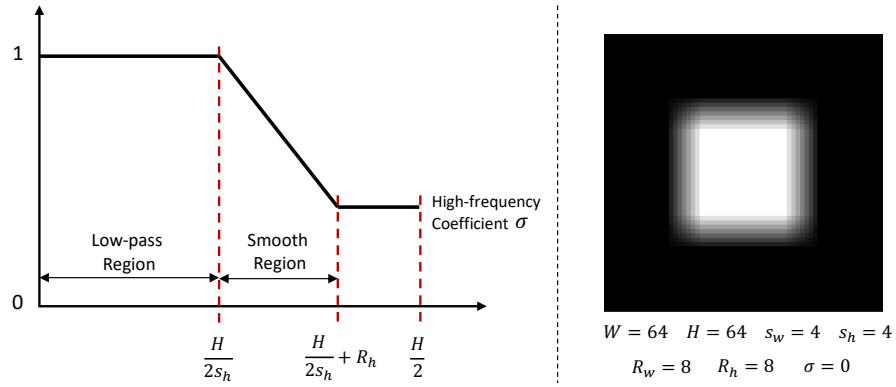


**Fig. 1:** Visualization of the design of a low-pass filter. (a) 1D filter for the positive axis. (2) 2D low-pass filter, which is constructed by mirroring the 1D filters and performing an outer product between two 1D filters, in accordance with the settings of the 1D filter.

Based on the dual principle, we can prove $F'(u, v)$ in the whole sub-band

$$F'(u,v) = \frac{1}{s^2}\left(\sum_{a,b=0}^{s-1} F\left(u + \frac{a\Omega_s}{s}, v + \frac{b\Omega_y}{s}\right)\right), \tag{17}$$

where $u \in \left(0, \frac{\Omega_x}{s}\right)$, $v \in \left(0, \frac{\Omega_y}{s}\right)$.

## 2 Implementation Details

### 2.1 Low-pass Filter Definition

In Fig. 1, we show the design of a low-pass filter used in FouriScale. Inspired by [7,8], we define the low-pass filter as the outer product between two 1D filters (depicted in the left of Fig. 1), one along the height dimension and one along the width dimension. We define the function of the 1D filter for the height dimension as follows, filters for the width dimension can be obtained in the same way:

$$\text{mask}^h_{(s_h,R_h,\sigma)} = \min\left(\max\left(\frac{1-\sigma}{R_h}\left(\frac{H}{s_h} + 1 - i\right) + 1, \sigma\right), 1\right), i \in [0, \frac{H}{2}], \tag{18}$$

where $s_h$ denotes the down-sampling factor between the target and original resolutions along the height dimension. $R_h$ controls the smoothness of the filter and $\sigma$ is the modulation coefficient for high frequencies. Exploiting the characteristic of conjugate symmetry of the spectrum, we only consider the positive axis, the whole 1D filter can be obtained by mirroring the 1D filter. We build the 2D low-pass filter as the outer product between the two 1D filters:

$$\text{mask}(s_h, s_w, R_h, R_w, \sigma) = \text{mask}^h_{(s_h,R_h,\sigma)} \otimes \text{mask}^w_{(s_w,R_w,\sigma)}, \tag{19}$$

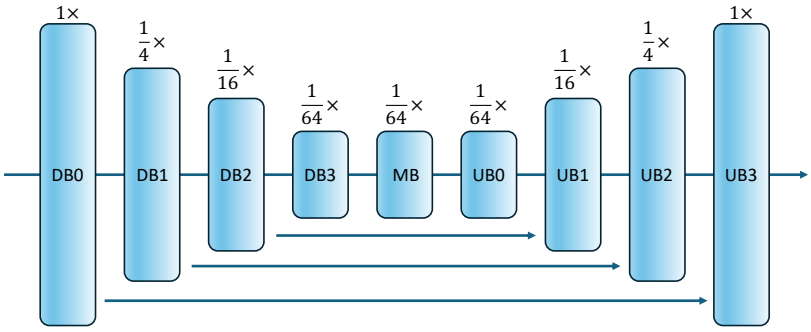


**Fig. 2:** Reference block names of stable diffusion in the following experiment details.

where $\otimes$ denotes the outer product operation. Likewise, the whole 2D filter can be obtained by mirroring along the height and width axes. A toy example of a 2D low-pass filter is shown in the right of Fig. 1.

### 2.2 Hyper-parameter Settings

In this section, we detail our choice of hyperparameters. The evaluative parameters are detailed in Tab. 1. Additionally, Fig. 2 provides a visual guide of the precise positioning of various blocks within the U-Net architecture employed in our model.

The dilation factor used in each FouriScale layer is determined by the maximum value of the height and width scale relative to the original resolution. As stated in our main manuscript, we employ an annealing strategy. For the first $S_{init}$ steps, we employ the ideal dilation convolution and low-pass filtering. During the span from $S_{init}$ to $S_{stop}$, we progressively decrease the dilation factor and $r$ (as detailed in Algorithm 1 of our main manuscript) down to 1. After $S_{stop}$ steps, the original UNet is utilized to refine image details further. The settings for $S_{init}$ and $S_{stop}$ are shown in Tab. 1.

**Table 1:** Experiment settings for SD 1.5, SD 2.1, and SDXL 1.0.

| Params | SD 1.5 & SD 2.1 | SDXL 1.0 |
|---|---|---|
| FouriScale blocks | [DB2,DB3,MB,UB0,UB1,UB2] | [DB2,MB,UB0,UB1] |
| inference timesteps | 50 | 50 |
| [$S_{init}$, $S_{stop}$] | [10,30] (4×1:1 and 6.25×1:1)<br>[20,35] (8×1:2 and 16×1:1) | [20,35] |

**Table 2:** Quantitative comparisons with SD + super-resolution method [9].

| **Method** | $FID_r$ | $KID_r$ | $FID_b$ | $KID_b$ | $FID_p$ | $KID_p$ |
|---|---|---|---|---|---|---|
| SD + Super Resolution | 25.94 | 0.91 | 20.10 | 0.56 | 84.93 | 3.31 |
| Ours | 39.49 | 1.27 | 28.14 | 0.73 | 70.15 | 2.20 |

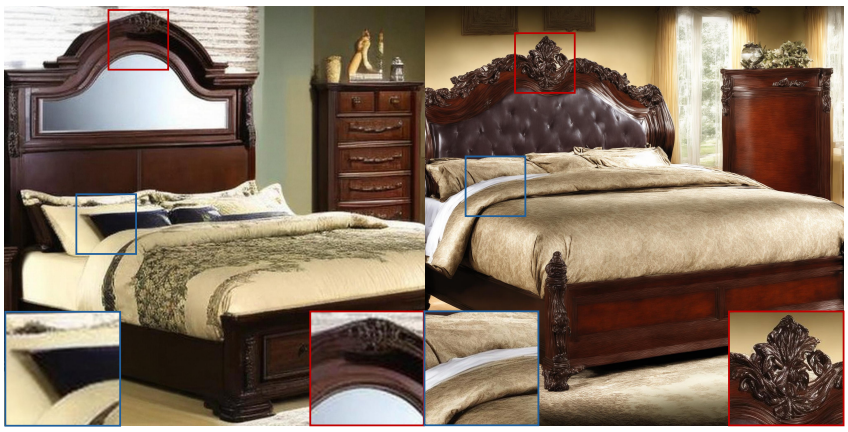

**Fig. 3:** Visual comparison with SD+SR. **Left:** 2048×2048 image upscaled by SD+SR from 512×512 SD 2.1 generated image. **Right:** 2048×2048 image generated by our FouriScale with SD 2.1.

## 3 More Experiments

### 3.1 Comparison with Diffusion Super-Resolution Method

In this section, we compare the performance of our proposed method with a cascaded pipeline, which uses SD 2.1 to generate images at the default resolution of 512×512, and upscale them to 2048×2048 by a pre-trained diffusion super-resolution model, specifically the Stable Diffusion Upscaler-4× [9]. We apply this super-resolution model to a set of 10,000 images generated by SD 2.1. We then evaluate the $FID_r$ and $KID_r$ scores of these upscaled images and compare them with images generated at 2048×2048 resolution using SD 2.1 equipped with our FouriScale. The results of this comparison are presented in Tab. 2. We also report FID-patch [2] ($FID_p$) and KID-patch ($KID_p$) for a more reasonable measure at high resolutions. As we can see, our method obtains somewhat worse results than the cascaded method. However, on the metrics of $FID_p$ and $KID_p$, our method achieves better performance, indicating our method can generate much better details than the cascaded pipeline, which is also proved by Fig. 3. Due to a lack of prior knowledge in generation, the super-resolution method can only utilize existing knowledge within a single image for upscaling the image, resulting in an over-smooth appearance. However, our method can effectively upscale images

**Table 3:** Quantitative comparisons among training-free methods. We generate $2048^2$ images using SDXL on a single NVIDIA A100 GPU.

| **Method** | $FID_r$ | $KID_r$ | $FID_b$ | $KID_b$ | $FID_p$ | $KID_p$ | Latency |
|---|---|---|---|---|---|---|---|
| ScaleCrafter (ICLR'24) | 49.46 | 1.73 | 36.22 | 1.07 | 65.06 | 2.17 | 58s |
| ElasticDiff (CVPR'24) | 52.02 | 3.03 | 40.46 | 2.22 | 76.77 | 3.45 | 212s |
| DemoFusion (CVPR'24) | 30.51 | 1.06 | 18.34 | 0.42 | 51.12 | 1.42 | 107s |
| Ours | 33.89 | 1.21 | 20.10 | 0.47 | 56.44 | 1.59 | 76s |

and fill in details using generative priors with a pre-trained diffusion model. Furthermore, our method is capable of generating high-resolution images in only one stage, without the need for a multi-stage process. Besides, our method does not need model re-training, while the SR model demands extensive data and computational resources for training.

### 3.2 Comparison with More SOTAs

We observe that the recent approach, ElasticDiffusion [5], has established a technique to equip pre-trained diffusion models with the capability to generate images of arbitrary sizes, both smaller and larger than the resolution used during training. Besides, DemoFusion [4] demonstrates promising results in high-resolution generation by employing a series of strategies, such as progressive upscaling, skip residual, and dilated sampling. Here, we provide a comparison with those state-of-the-art methods on the SDXL $2048\times2048$ setting.

The results are shown in Tab 3. First, it's important to note that the inference times for ElasticDiffusion are approximately 4 to 5 times longer than ours. DemoFusion also takes nearly $1.5\times$ inference time of ours due to its cascaded architecture. Our method demonstrates performance on par with the leading approach, DemoFusion, while significantly reducing inference costs. When compared to ScaleCrafter and ElasticDiffusion, we outperform them across all evaluation metrics, achieving lower FID and KID scores. This indicates that our method produces images of higher quality and greater diversity.

### 3.3 More Ablation Studies

We present additional ablation studies in Tab. 4, with experiments conducted using SD 2.1 to generate $2048^2$ images.

In Tab. 4(a), we examine the effect of modifying the dilation rate. To isolate this factor, we omit low-pass filtering and FouriScale guidance. Given that SD 2.1 is trained on $512\times512$ images, our main manuscript's conclusion on structural consistency suggests an optimal dilation rate of 4. When the rate is changed to 2 or 6, we observe a significant performance decrease, highlighting the importance of maintaining structural consistency.

**Table 4:** Quantitative results of generating $2048^2$ images using SD 2.1. $\text{FID}_p$ and $\text{KID}_p$ are patched FID/KID. d-x denotes the dilation rate is x, m-x denotes the mask size for low-pass filtering is (H/x, W/x).

| **Method** | | $\text{FID}_r$ | $\text{KID}_r$ | $\text{FID}_p$ | $\text{KID}_p$ |
|---|---|---|---|---|---|
| (a) Dilation | d-2 | 52.70 | 1.71 | 74.50 | 2.51 |
| | d-4 (ours) | 47.66 | 1.59 | 71.52 | 2.34 |
| | d-6 | 58.44 | 2.09 | 75.83 | 2.28 |
| (b) Low-pass filter | m-2 | 44.00 | 1.49 | 76.66 | 2.46 |
| | m-4 (ours) | 41.16 | 1.36 | 71.94 | 2.33 |
| | m-6 | 56.27 | 2.00 | 85.50 | 2.87 |

**Fig. 4:** Visualization of the high-resolution images generated by SD 2.1 integrated with customized LoRAs (images in red rectangle) and images generated by a personalized diffusion model, AnimeArtXL [1], which is based on SDXL.

Tab. 4(b) evaluates the impact of the low-pass filtering mask size, without employing FouriScale guidance. Consistent with our expectations, the optimal mask size of (H/4, W/4) yields the best performance, further validating the importance of scale consistency.

## 4 More Visualizations

### 4.1 LoRAs

In Fig. 4, we present the high-resolution images produced by SD 2.1, which has been integrated with customized LoRAs [6] from Civitai [3]. We can see that our method can be effectively applied to diffusion models equipped with LoRAs.

### 4.2 Diffusion-based Applications

Our approach is applicable to any method that employs a diffusion model as a generator, which incorporates a specific number of convolutional layers. In Fig.

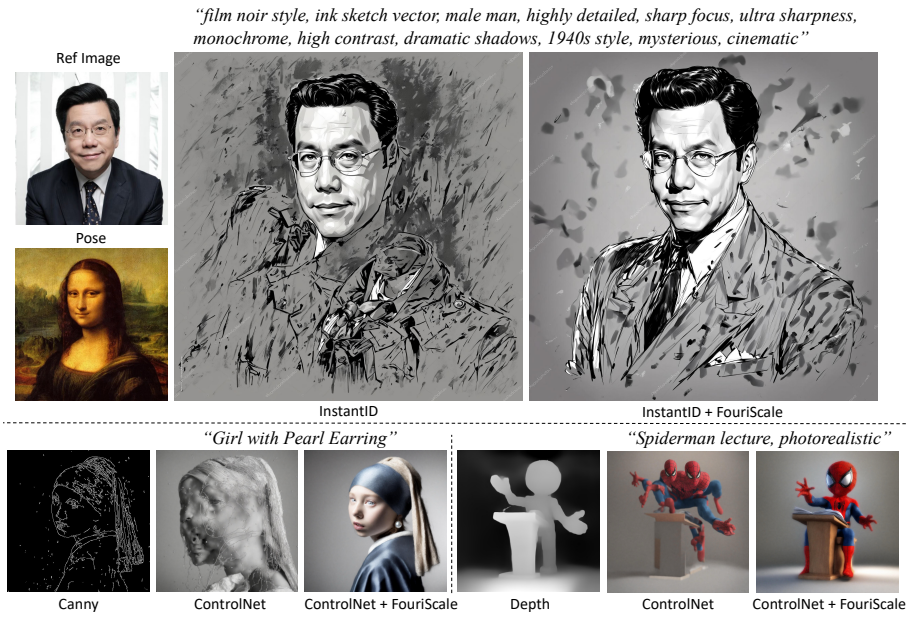


**Fig. 5:** Visualization of $2048^2$ images generated by InstantID and ControlNet (with SDXL), and them equipped with FouriScale.

5, we apply our method to InstantID [10] and ControlNet [11], our method works well on these methods, demonstrating the scalability of our method.

### 4.3 Other Resolutions

In Fig. 6, we present more images generated at different resolutions by SD 2.1, aside from the 4×, 6.25×, 8×, and 16× settings. Our approach is capable of generating high-quality images of arbitrary aspect ratios and sizes.

## References

1. AnimeArtXL: (2024), `https://civitai.com/models/117259/anime-art-diffusion-xl`, accessed: 17, 01, 2024
2. Chai, L., Gharbi, M., Shechtman, E., Isola, P., Zhang, R.: Any-resolution training for high-resolution image synthesis. In: ECCV. pp. 170–188. Springer (2022)
3. Civitai: (2024), `https://civitai.com/`, accessed: 17, 01, 2024
4. Du, R., Chang, D., Hospedales, T., Song, Y.Z., Ma, Z.: Demofusion: Democratising high-resolution image generation with no $$$. In: CVPR. pp. 6159–6168 (2024)
5. Haji-Ali, M., Balakrishnan, G., Ordonez, V.: Elasticdiffusion: Training-free arbitrary size image generation through global-local content separation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6603–6612 (2024)

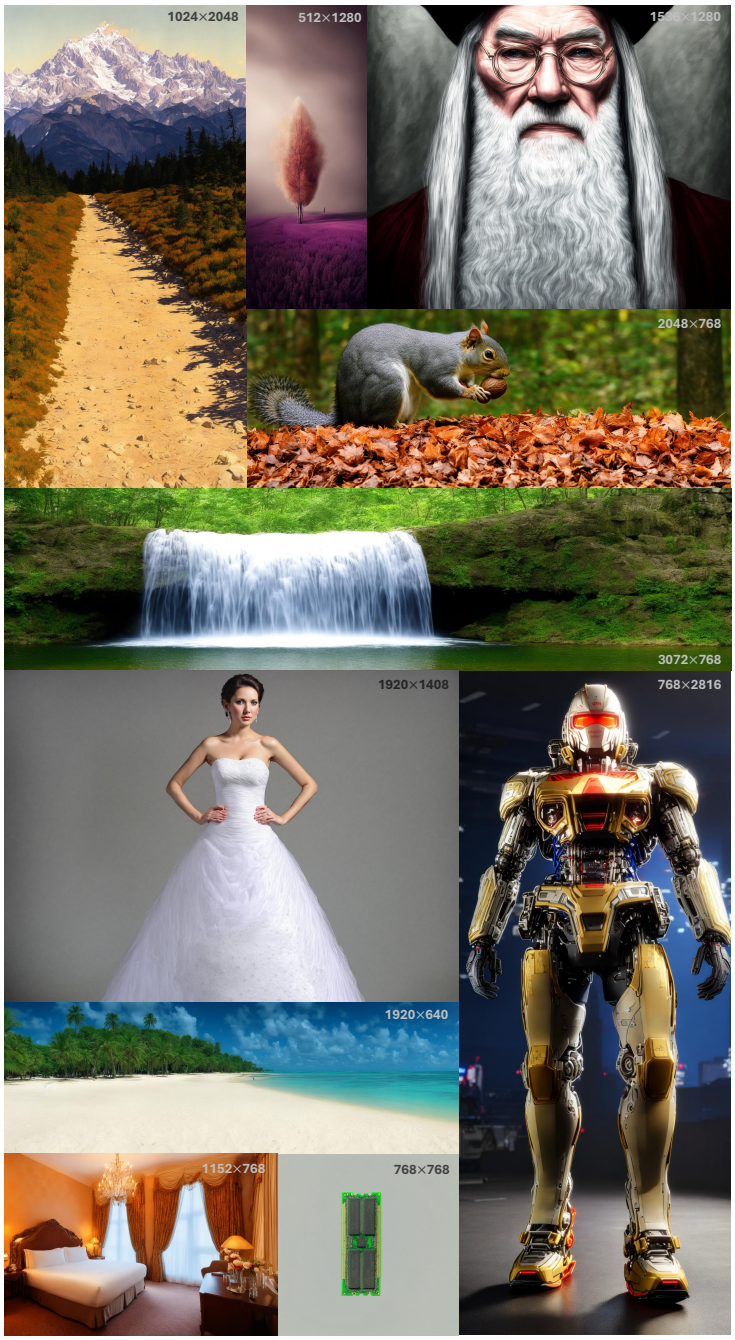


**Fig. 6:** More generated images using FouriScale and SD 2.1 with arbitrary resolutions.

6. 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: International Conference on Learning Representations (2021)
7. Riad, R., Teboul, O., Grangier, D., Zeghidour, N.: Learning strides in convolutional neural networks. In: ICLR (2021)
8. Sukhbaatar, S., Grave, É., Bojanowski, P., Joulin, A.: Adaptive attention span in transformers. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 331–335 (2019)
9. Upscaler, S.D.: (2024), `https://huggingface.co/stabilityai/stable-diffusion-x4-upscaler`, accessed: 17, 01, 2024
10. Wang, Q., Bai, X., Wang, H., Qin, Z., Chen, A.: Instantid: Zero-shot identity-preserving generation in seconds. arXiv preprint arXiv:2401.07519 (2024)
11. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV. pp. 3836–3847 (2023)