A. Appendix

To complement the main content of the paper, we provide here additional details about the method in Sec. B as well as additional quantitative and qualitative results in Sec C.

B. Additional technical details

B.1. Frequency Modulation details

Time-varying high-pass filter definition. In our method, we rely on frequency domain and use a high pass filter to steer the denoising process as described in equation (4). In the following, we provide the formal definition of the time-varying high pass filter, $\mathcal{K}(t)$, that we used.

The high-pass filters $\mathcal{K}(t)$ have time-varying cut-off frequencies, defined as follows:

$$\rho(t) = \frac{t}{T} \tag{8}$$

$$\tau_h(t) = h \cdot c \cdot (1 - \rho(t)) \tag{9}$$

$$\tau_w(t) = w \cdot c \cdot (1 - \rho(t)) \tag{10}$$

where $\tau_h(t)$ and $\tau_w(t)$ are the horizontal and vertical cut-off frequencies at timestep t, respectively. Subsequently, the mask $\mathcal{K}(t)$, which is applied on the shifted frequency spectrum centered on (x_c, y_c), is defined as

$$\mathcal{K}(t) = \begin{cases} \rho(t), & \text{if } |x - x_c| < \frac{\tau_w(t)}{2} \\ & \quad \& \ |y - y_c| < \frac{\tau_h(t)}{2}, \\ 1, & \text{otherwise} \end{cases} \tag{11}$$

The cut-off frequency grows as the denoising process progresses, while the scaling factor of the low-frequency coefficients decreases. Our frequency modulation is designed such that the guidance from the denoised latent $\tilde{\mathbf{z}}_t$ becomes more significant as $t \to 0$. In our experiments, we set $c = 0.5$.

Derivation of the Frequency Modulation in time-domain. In the main paper, we mention that our frequency modulation introduced in Eq. (4) can be reformulated in time domain as Eq. (5) and discuss the corresponding benefits. Here, we provide a formal derivation to support the equivalence between the two formulations. For ease of presentation, we omit the timestep t and resolution m notations from operands.

Let $\mathbf{z} \in \mathbb{R}^{h \times w}$ be the 2D latent, and $\mathbf{Z} = DFT_{2D}(\mathbf{z}) \in \mathbb{C}^{h \times w}$ be the Fourier transform of $\mathbf{z}$. Written in matrix form,

$$\mathbf{Z} = (W_r \mathbf{z} W_c), \tag{12}$$

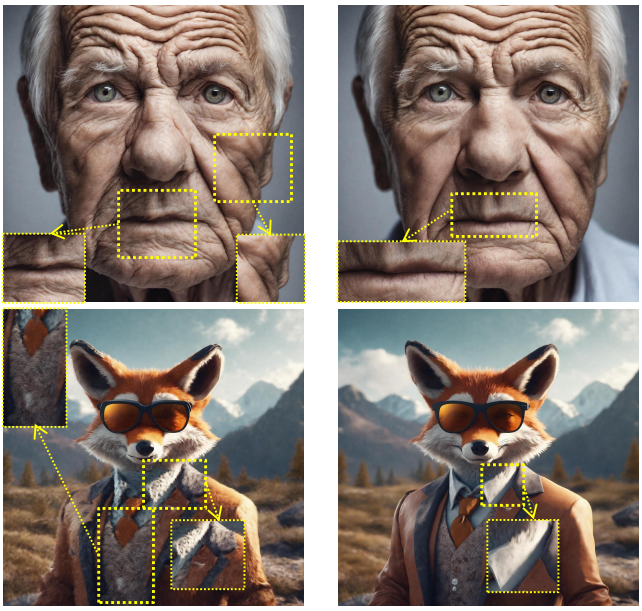
where $W_r \in \mathbb{C}^{h \times h}, W_c \in \mathbb{C}^{w \times w}$ are the row- and column-wise Fourier transform matrices, respectively. Let $\mathcal{K} \in \mathbb{R}^{h \times w}$ be the high-pass filter defined in the previous section, our proposed mixing operation in the frequency domain is formulated as below:

$$\begin{aligned} \hat{\mathbf{Z}} &= \mathcal{K} \odot DFT_{2D}(\mathbf{z}) + (1 - \mathcal{K}) \odot DFT_{2D}(\tilde{\mathbf{z}}) \\ &= \mathcal{K} \odot (W_r \mathbf{z} W_c) + (1 - \mathcal{K}) \odot (W_r \tilde{\mathbf{z}} W_c) \\ &= W_r \mathbf{z} W_c + (1 - \mathcal{K}) \odot (W_r(\tilde{\mathbf{z}} - \mathbf{z}) W_c) \end{aligned}$$

The inverse DFT of $\hat{\mathbf{Z}}$, which is the outcome of Eq. 4, is formulated as:

$$\begin{aligned} \hat{\mathbf{z}} &= IDFT_{2D}(\hat{\mathbf{Z}}) \\ &= W_r^{-1} \left(W_r \mathbf{z} W_c + (1 - \mathcal{K}) \odot (W_r(\tilde{\mathbf{z}} - \mathbf{z}) W_c) \right) W_c^{-1} \\ &= W_r^{-1} W_r \mathbf{z} W_c W_c^{-1} \\ &\quad + W_r^{-1} \left((1 - \mathcal{K}) \odot (W_r(\tilde{\mathbf{z}} - \mathbf{z}) W_c) \right) W_c^{-1} \\ &= \mathbf{z} + \left(W_r^{-1} (1 - \mathcal{K}) W_c^{-1} \right) \circledast \left(W_r^{-1} W_r (\tilde{\mathbf{z}} - \mathbf{z}) W_c W_c^{-1} \right) \\ &= \mathbf{z} + k \circledast (\tilde{\mathbf{z}} - \mathbf{z}), \end{aligned}$$

resulting in Eq. 5 in the main paper, where $k = W_r^{-1}(1 - K)W_c^{-1} = IDFT_{2D}(1 - \mathcal{K})$ is a convolutional kernel and $\circledast$ denotes a circular convolution operator.



(a) Swapping (b) Modulation

Figure 8. Comparison of Attention Swapping and Modulation

B.2. Attention Modulation details

In our method, Attention Modulation can be in practice implemented as:

$$\begin{aligned} z' &= (\lambda \cdot \mathcal{U}(M^n, s) + (1 - \lambda) \cdot M^m) \cdot V_m \\ &= \lambda \cdot \text{Att}(\mathcal{U}(Q_n, s), \mathcal{U}(K_n, s), V_m) \\ &\quad + (1 - \lambda) \cdot \text{Att}(Q_m, K_m, V_m) \end{aligned}$$

$\mathcal{U}$ denotes an s-times upsampling function. Both attention operations can utilize Flash Attention. We also note that Flash Attention is available as a Triton kernel, hence a custom kernel supporting AM could be implemented by scaling the raw block-wise scores directly.

B.3. Attention Modulation analysis

As mentioned in Sec. 3.3, we take inspiration from recent literature using attention swapping to control local texture. However, rather than swapping attention, we mix the two attention paths instead. In Figure 8 we compare attention swapping versus our proposed attention modulation. These results clearly show the benefit of including the attention from the high resolution path rather than directly swapping with the low res pass to avoid loss of information from the high res denoising path. We empirically set λ used in Eq (6) to 0.7.

C. Additional experimental results

C.1. Quantitative results for FM and AM

In Table 2 shows an ablation of the FAM Diffusion components, showing that: (1) Each component provides large improvements over the baseline (especially on the more meaningful FID_c and KID_c metrics), (2) FM and AM individual gains accumulate when used in combination.

C.2. FAM diffusion with different SD backbones

In Table 1 we show that our method outperforms several baselines when combined with SDXL. In addition to those main results, we further combine our FAM diffusion method with various SD backbones. The quantitative results in Table 3 demonstrate that our approach can seamless combine with different variants of SD and provides similarly large improvements in quality and image-text alignment across all experimental settings.

C.3. FAM diffusion with different aspect ratios

Thus far, we have used our method to generate high-resolution images by equally upscaling both the height and width. Here, we study the effect of using Fam diffusion targeting different aspect ratios. In particular, starting from the SDXL model, we use our approach targeting higher resolutions with different aspect ratios. The quantitative results in Table 4 and qualitative results shown in Figures 9 through 11, clearly highlight the versatility of our method that can seamlessly adapt to various settings without compromising quality.

C.4. FAM diffusion with different conditioning terms

Fam Diffusion enables seamless integration with various LDM-based applications, such as ControlNet [33]. As shown in Figure 12, Fam Diffusion combined with ControlNet [33] achieves controllable high-resolution generation, with examples showcasing the use of images and canny edges as conditions.

Method	FID↓	KID↓	FID_c ↓	KID_c ↓	CLIP ↑
SDXL	59.5	**0.0067**	50.5	0.0136	30.6
FM	59.4	0.0079	38.9	0.0112	31.1
AM	59.9	0.0075	41.3	0.0102	30.9
FAM	**58.9**	0.0072	**34.0**	**0.0080**	**32.3**

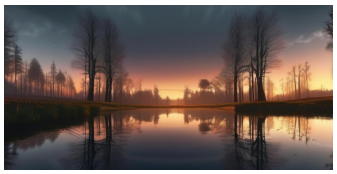
Table 2. Ablation of FAM components at 2K resolution.

Method	Resolution Scale Factor	$FID_r \downarrow$	$KID_r \downarrow$	$FID_c \downarrow$	$KID_c \downarrow$	CLIP Score ↑
SD 1.5	2×2	75.36	0.0122	43.99	0.0103	30.35
SD 1.5 + FAM diffusion		65.07	0.0087	34.06	0.0082	30.92
SD 2.1		86.62	0.0163	53.67	0.0137	29.66
SD 2.1 + FAM diffusion		64.77	0.0084	38.18	0.0091	31.13
SDXL		59.47	0.0067	50.54	0.0136	30.6
SDXL+ FAM diffusion		58.91	0.0072	33.96	0.0080	32.35
SD 1.5	3×3	106.50	0.0251	48.92	0.0133	28.89
SD 1.5 + FAM diffusion		38.19	0.0011	43.99	0.0082	30.44
SD 2.1		137.05	0.0384	63.91	0.01719	27.81
SD 2.1 + FAM diffusion		64.8	0.0089	40.49	0.0114	31.13
SDXL		78.41	0.0136	69.40	0.0210	28.44
SDXL + FAM diffusion		69.25	0.0007	36.40	0.0100	32.25
SD 1.5	4×4	150.84	0.0474	55.97	0.0155	27.40
SD 1.5 + FAM diffusion		67.77	0.0086	40.21	0.0012	30.36
SD 2.1		177.06	0.0645	69.43	0.019	26.36
SD 2.1+ FAM diffusion		66.32	0.0085	41.37	0.0018	31.10
SDXL		160.10	0.0602	74.37	0.0242	26.70
SDXL + FAM diffusion		58.91	0.0073	43.65	0.0130	32.33

Table 3. Comparison of vanilla Stable Diffusion and our FAM diffusion.

Method	Scaling Factor	FID↓	KID↓	$FID_c \downarrow$	$KID_c \downarrow$	CLIP ↑
DemoFusion [3]	2×4	81.69	0.0112	54.48	0.0165	29.3
AccDiffusion [15]		70.42	0.0119	55.73	0.0205	29.0
FouriScale* [12]		71.86	0.0302	63.28	0.0322	25.8
HiDiffusion [34]		118.56	0.038	65.46	0.021	26.3
SDXL [19]		80.62	0.0236	67.46	0.0302	25.5
SDXL [19] + FAM diffusion		**63.48**	**0.0090**	**41.44**	**0.0115**	**30.6**

Table 4. System-level comparisons with SDXL. * indicates inference with FreeU [26]



nature in the reflection of a mirror which is located in the middle of the caos, realistic, well done, detailed, 8k

A micro-tiny clay pot full of dirt with a beautiful daisie planted in it, shining in the autumn sun on a road in an abandoned city, fiction, wallpaper, character, cg artwork, art, flash photography

(a) Native Resolution Image

(b) DemoFusion

(c) FouriScale*

Figure 9. Qualitative comparison with other methods based on SDXL. Best viewed when zoomed in. * indicates inference with FreeU [26]. *(Continued in Fig. 10).*

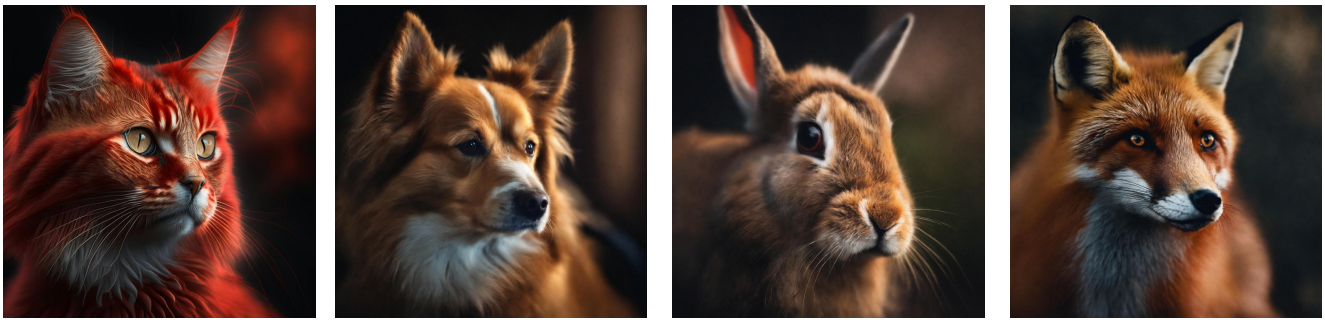
(a) HiDiffusion

(b) Our Method

Figure 10. Qualitative comparison with other methods based on SDXL *(continued from Fig. 9)*. Best viewed when zoomed in.

(a) FouriScale* (b) HiDiffusion (c) Our Method

Figure 11. Qualitative comparison with other methods based on SDXL with arbitrary resolutions. DemoFusion is unable to handle arbitrary resolutions, therefore not included. Best viewed when zoomed in.



(a) Image to Image

(b) Canny Edges to Image

Figure 12. Results of FAM Diffusion combining with ControlNet [33]. All images are generated at 2× (2048 × 2048).Best viewed when zoomed in.