

## Appendix

<b>A How Does TFSA/Attention Guidance Work?</b>	<b>14</b>
A.1 TFSA Clusters Semantically Related Tokens . . . . .	14
A.2 TFSA Adjusts the Amplitude of High- and Low-frequency Components . . . . .	15
A.3 Visualization of Attention Maps in TFSA . . . . .	17
<b>B Supplementary Qualitative Comparison of §4.3</b>	<b>17</b>
<b>C Supplementary Ablation Experiments of §5</b>	<b>17</b>
C.1 Further Qualitative Analysis of Attention Guidance . . . . .	17
C.2 Ablation on the hyper-parameters of Attention Guidance . . . . .	17
C.3 Ablation on Progressive Scheduler Value . . . . .	18
<b>D Ablation on the Attention Guidance Components</b>	<b>19</b>
D.1 Ablation on the Guidance Scale Decay Strategy . . . . .	19
D.2 Ablation on the Attention Calculation Paradigm . . . . .	20
<b>E Further Model Efficiency Analysis</b>	<b>21</b>
<b>F RepLDM Algorithm</b>	<b>21</b>
<b>G Robustness Analysis</b>	<b>22</b>

### A How Does TFSA/Attention Guidance Work?

In this section, we further elaborate on the working mechanism of attention guidance. Our attention guidance enhances the structural consistency of the latent representation by integrating the output of TFSA. Therefore, we conduct a detailed analysis of TFSA. Specifically, the functionality of TFSA can be described in two aspects: (i) *clustering the related tokens* in the latent representations; (ii) *adjusting the amplitude of the high-frequency and low-frequency components* in the latent representations.

#### A.1 TFSA Clusters Semantically Related Tokens

**Visualization of the clustering effect of TFSA.** TFSA reorganizes tokens based on their similarities. Intuitively, this enables TFSA to perform token clustering, which enhances the structural consistency of latent representations. To demonstrate the clustering effect of TFSA, we calculated the deviation of the tokens’ mean (DTM) of the latent representations  $\tilde{z}_t$  and  $z_t$ . Concretely, assuming  $z_t \in \mathbb{R}^{h \times w \times c}$ , and  $Z_t = \text{Flatten}(z_t) = [y_{t1}, \dots, y_{tN}] \in \mathbb{R}^{N \times c}$ , where  $N = h \times w$ , we calculate DTM as:

$$\text{DTM} = [\text{mean}(y_{ti}) - \text{mean}(Z_t) \text{ for } i = 1, \dots, N] \quad (5)$$

To provide an intuitive illustration of the clustering effect of TFSA, we visualize the DTM based on token indices (*i.e.*,  $i = 1, \dots, N$ ) when  $t$  is relatively large. As shown in columns (A) and (B) of Fig. 13, compared to the DTM of  $z_t$  (blue points), the DTM of  $\tilde{z}_t$  (red points) becomes more dispersed and exhibits distinct stripe patterns, indicating that TFSA indeed clusters the tokens of the latent representations. This clustering effect can be more directly demonstrated when  $t$  is smaller. As shown in the heatmaps in columns (C) and (D) of Fig. 13, it is evident that TFSA clusters semantically related tokens.

**The clustering effect of TFSA leads to accelerated structural denoising.** Fig. 13 shows that the clustering effect of TFSA clarifies the semantic structures of objects, enabling the model to complete the denoising of low-frequency structures earlier. This early revelation of the overall image layout

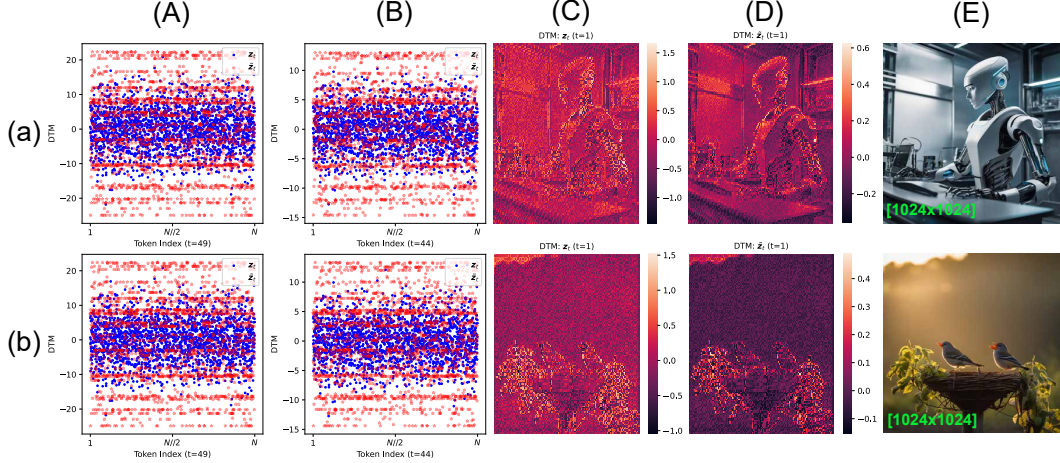


Figure 13: **The clustering effect of TFSA.** Columns (A), (B), (C), and (D) show the DTM of latent representations, while column (E) presents the corresponding generated RGB images.

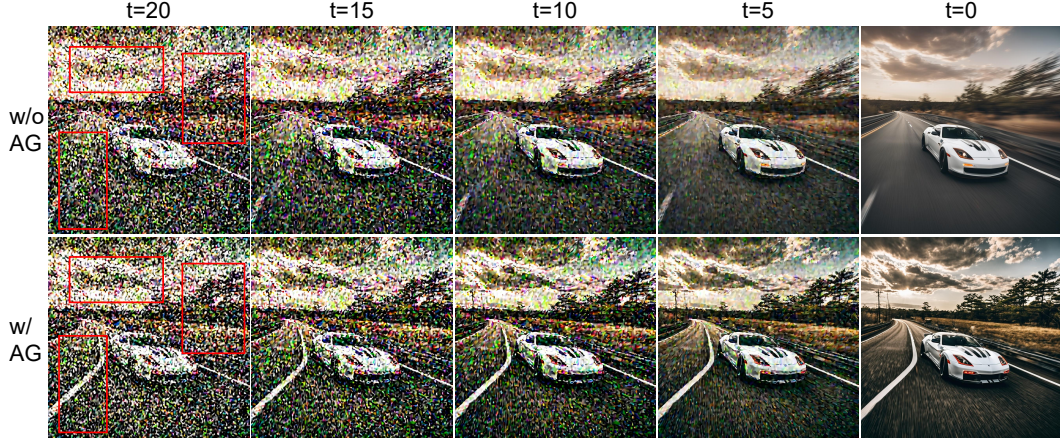


Figure 14: **Denoising visualization for the ablation of attention guidance.** As indicated by the red boxes, the clustering effect of TFSA prompts earlier structural emergence, delivering better prior for subsequent fine-detail generation. Resolution:  $1024 \times 1024$ .

provides a stronger prior for subsequent fine-detail generation. To illustrate this, Fig. 14 presents the denoising process for the ablation of attention guidance. Note the regions highlighted by red boxes. With the incorporation of attention guidance, these areas exhibit clearer structures, which facilitates the generation of more affluent details and more vivid colors in subsequent steps.

To quantitatively demonstrate that TFSA accelerates structural emergence, we calculate the SSIM between  $z_t$  and  $z_0$ , where  $t \in 1, 2, \dots, T-1$ , and  $T = 50$ . As shown in Fig. 15, compared to the naive denoising process, attention guidance consistently drives the latent representations closer to their final states at each step, indicating the structural foreseeability of TFSA.

## A.2 TFSA Adjusts the Amplitude of High- and Low-frequency Components

The aim of this experiment is to explain: (i) why appropriately delaying attention guidance can resolve structural deformation issues (as shown in Fig. 9); (ii) why attention guidance enhances the details and colors of the image (as shown in Fig. 6

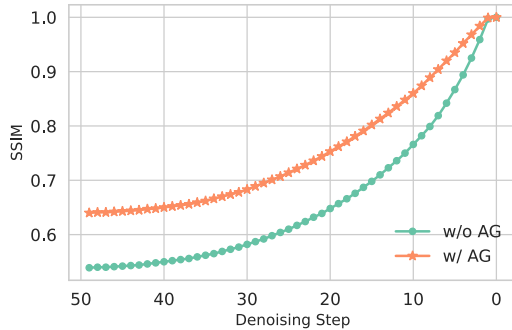


Figure 15: **Quantitatively analysis on the clustering effect of TFSA.** We calculate the SSIM between noised latents  $z_t$  ( $1 \leq t \leq 49$ ) and their corresponding clean latent  $z_0$ .

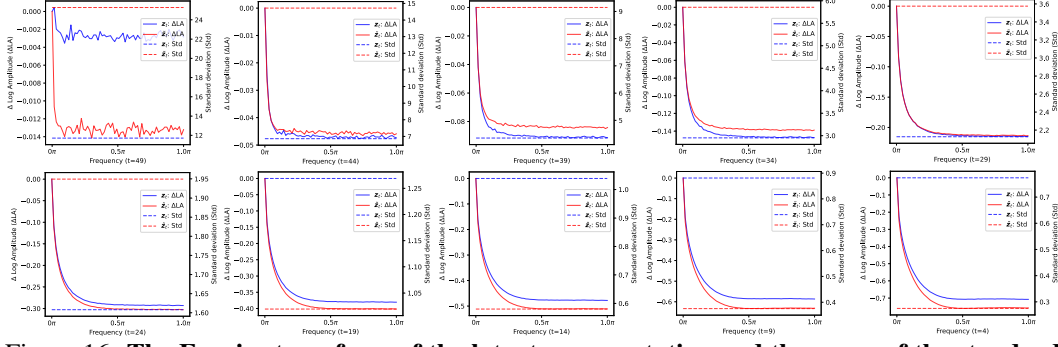


Figure 16: **The Fourier transform of the latent representation and the mean of the standard deviations across all channels.**  $z_t$  is represented in blue, while  $\tilde{z}_t$  is represented in red; the Fourier transforms are shown as solid lines, and the standard deviations are shown as dashed lines. The results are based on the generation process of 5k images.

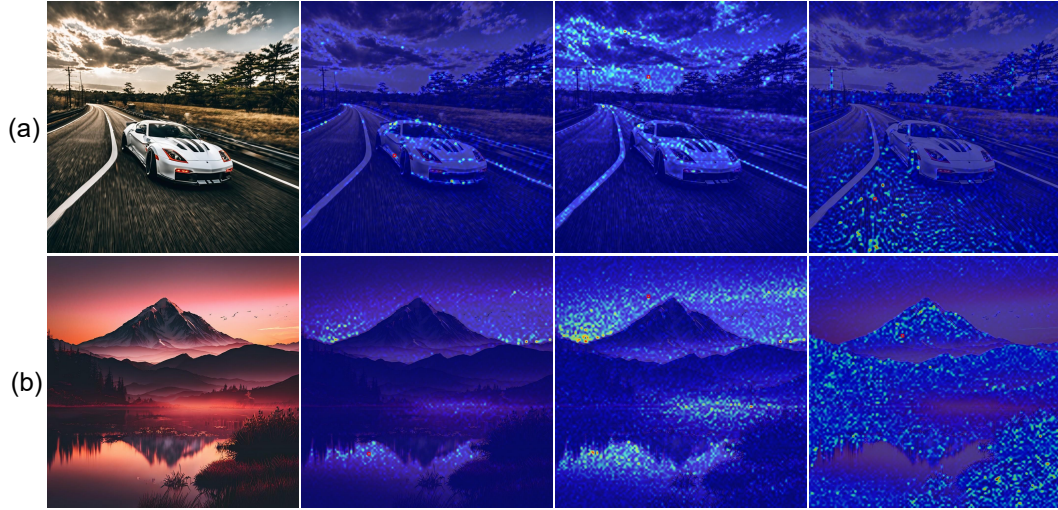


Figure 17: **Visualization of attention maps in TFSA.** The query tokens are highlighted with red boxes, and the heatmap color ranges from blue to red, indicating increasing correlation strength between the key tokens and the query tokens. Resolution:  $1024 \times 1024$ . Zoom-in for a better view.

and 8); and (iii) why applying attention guidance in the later stages of denoising does not enhance the image details and colors (as shown in Fig. 10).

To explain the aforementioned three points, as shown in Fig. 16, we calculate the Fourier transforms of  $z_t$  (blue solid line) and  $\tilde{z}_t$  (red solid line), along with the mean of the standard deviations for all their channels (dashed line). It can be observed that TFSA significantly alters the relative amplitudes of the high- and low-frequency components in the latent representations during the initial denoising steps (from  $t = 49$  to  $t = 47$ ), particularly affecting the low-frequency components, which results in structural deformation. During the early and middle stages of denoising (from  $t = 44$  to  $t = 29$ ), TFSA increases the amplitudes of high-frequency components in the latent representations, which explains why attention guidance leads to richer details and colors. In the later stages of denoising (from  $t = 28$  to  $t = 0$ ), TFSA slightly suppresses the high-frequency components of the latent representations while almost leaving the low-frequency components unchanged. This explains why applying attention guidance in the later stages of denoising cannot enrich details and colors of the generated images.

Additionally, Fig. 16 shows that TFSA increases the standard deviation of  $\tilde{z}_t$  during the early and middle stages of denoising, while decreasing it in the later stages. The trend of the standard deviation changes is closely consistent with the variation in the amplitude of the high-frequency components. We conjecture that this is because the amount of information in the latent representations is positively correlated with the standard deviation, where a larger standard deviation corresponds to more image details and larger high-frequency components.



### A.3 Visualization of Attention Maps in TFSA

To further demonstrate the clustering effect of TFSA on related tokens, we visualize its attention maps. As shown in Fig. 17, without using projection matrices, the correlations between tokens are determined jointly by their represented colors and semantics. For example, in Fig. 17(a), the key tokens correlated with the query token at the selected car location are related not only to the car itself (*i.e.*, the concept of the car) but also to its color. TFSA leverages such correlations to fuse token information, thereby accelerating the formation of the overall image layout.

## B Supplementary Qualitative Comparison of §4.3

Fig. 18 presents additional qualitative comparison results. MultiDiffusion continues to struggle with maintaining global consistency; as indicated by the red boxes, DemoFusion tends to produce repetitive content, a problem somewhat alleviated in AccDiffusion but not fully resolved. As highlighted by the black boxes, another issue with AccDiffusion is the presence of noticeable streak artifacts in the images.

## C Supplementary Ablation Experiments of §5

### C.1 Further Qualitative Analysis of Attention Guidance

Fig. 19 provides additional qualitative ablation results on attention guidance. Individual preferences for contrast, color vividness, and detail richness may vary. attention guidance allows users to adjust parameters such as the guidance scale to synthesize images according to their preferences.

### C.2 Ablation on the hyper-parameters of Attention Guidance

**Quantitative analysis of guidance scale.** We sampled 1k prompts, fixed  $\eta_1 = 0.06$ ,  $\eta_2 = [0.2]$  and performed ablation studies for guidance scale  $\gamma$ . The quantitative results are shown in Table 5. Considering all metrics, we find that  $\gamma = 0.004$  achieved better quantitative results.

Table 5: **Quantitative ablation experiments on the guidance scale  $\gamma$ .** The best results are marked in **bold**, and the second best results are marked by underline.

Method	1024 × 1024					1600 × 1600					2048 × 2048				
	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
$\gamma = 0.000$	90.85	58.18	21.21	<b>17.69</b>	25.09	90.91	54.74	<b>21.45</b>	15.41	24.93	91.78	59.08	<u>21.57</u>	<u>17.36</u>	24.86
$\gamma = 0.001$	90.50	58.04	<b>21.34</b>	16.76	25.08	91.17	54.31	21.19	<u>15.47</u>	24.93	91.40	58.75	<b>21.87</b>	15.85	24.86
$\gamma = 0.002$	89.82	57.54	<u>21.28</u>	<u>17.04</u>	25.08	90.39	<b>53.71</b>	21.26	15.00	24.97	90.81	<u>58.34</u>	21.45	17.16	24.90
$\gamma = 0.003$	90.10	<u>57.08</u>	20.80	16.61	25.08	90.56	<u>53.95</u>	<u>21.35</u>	15.46	24.98	90.87	58.40	21.47	<b>17.60</b>	24.92
$\gamma = 0.004$	<b>89.40</b>	<b>56.64</b>	20.96	16.63	25.09	<b>89.91</b>	54.23	20.91	<b>15.54</b>	25.01	<b>90.11</b>	<b>58.11</b>	21.18	16.78	24.94s
$\gamma = 0.005$	90.17	57.50	20.89	16.34	25.12	<u>90.24</u>	55.19	20.67	15.21	25.02	90.46	58.91	20.79	16.87	24.97
$\gamma = 0.006$	<u>89.79</u>	58.18	20.33	15.93	25.16	90.36	56.71	20.33	14.59	25.06	<u>90.32</u>	59.86	20.37	16.12	25.00
$\gamma = 0.007$	90.42	60.29	20.07	16.20	25.21	90.91	59.35	20.36	14.16	25.12	90.86	61.81	20.14	15.70	25.06
$\gamma = 0.008$	91.64	63.63	19.66	14.25	<b>25.25</b>	91.98	63.93	19.13	13.71	<u>25.13</u>	92.16	64.82	19.59	14.24	<u>25.08</u>
$\gamma = 0.009$	94.29	67.87	19.15	13.00	<u>25.25</u>	94.38	70.21	19.45	12.12	<b>25.16</b>	94.39	68.84	19.22	13.63	<b>25.12</b>

**Quantitative analysis of delay rate.** We sampled 1k prompts, fixed  $\gamma = 0.004$ ,  $\eta_2 = [0.2]$  and performed ablation studies for delay rate  $\eta_1$ . Table 6 presents the experimental results, indicating that better results can be achieved when  $\eta_1 = 0.06$ . This means that appropriately delaying the effect of attention guidance can further enhance the quality of the generated images.

Table 6: **Quantitative ablation experiments on the delay rate  $\eta_1$ .** The best results are marked in **bold**, and the second best results are marked by underline.

Method	1024 × 1024					1600 × 1600					2048 × 2048				
	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
$\eta_1 = 0.00$	89.98	58.29	20.74	16.48	25.06	90.89	55.54	21.00	14.42	24.98	90.75	59.41	20.54	16.99	24.91
$\eta_1 = 0.02$	89.96	57.67	20.99	<b>16.87</b>	25.05	90.76	54.77	21.08	15.35	24.95	91.78	59.08	<b>21.57</b>	<b>18.16</b>	24.86
$\eta_1 = 0.04$	89.47	57.28	20.98	16.63	25.07	90.22	54.14	20.86	15.43	24.98	90.52	58.47	20.76	17.02	24.91
$\eta_1 = 0.06$	<u>89.44</u>	<b>56.64</b>	20.92	16.58	<b>25.11</b>	<u>89.91</u>	54.23	20.91	15.54	25.01	<b>90.11</b>	<b>58.11</b>	21.18	16.78	24.94
$\eta_1 = 0.08$	89.95	56.97	21.05	16.76	25.09	<b>89.87</b>	54.10	21.22	<u>15.65</u>	24.98	90.74	58.45	20.99	17.06	<b>24.92</b>
$\eta_1 = 0.10$	<b>89.29</b>	<u>56.88</u>	<b>21.11</b>	<u>16.84</u>	25.09	89.97	53.99	21.04	15.37	<u>24.99</u>	90.41	58.45	20.99	17.12	<u>24.92</u>
$\eta_1 = 0.12$	89.84	57.32	21.05	16.58	25.08	90.00	53.85	21.24	<b>15.81</b>	24.93	<u>90.24</u>	58.45	<u>21.24</u>	<u>17.36</u>	24.90
$\eta_1 = 0.14$	89.85	57.12	20.91	16.40	<u>25.09</u>	90.06	<u>53.83</u>	<u>21.33</u>	15.62	<b>24.99</b>	90.69	<u>58.25</u>	21.17	16.74	24.91
$\eta_1 = 0.16$	90.06	57.28	<u>21.10</u>	16.53	25.09	90.91	54.74	<b>21.45</b>	15.41	24.93	90.76	58.37	20.97	16.87	24.91
$\eta_1 = 0.18$	90.16	57.29	20.88	15.10	25.08	90.26	<b>53.79</b>	21.06	15.07	24.97	90.78	58.33	21.05	17.21	24.90



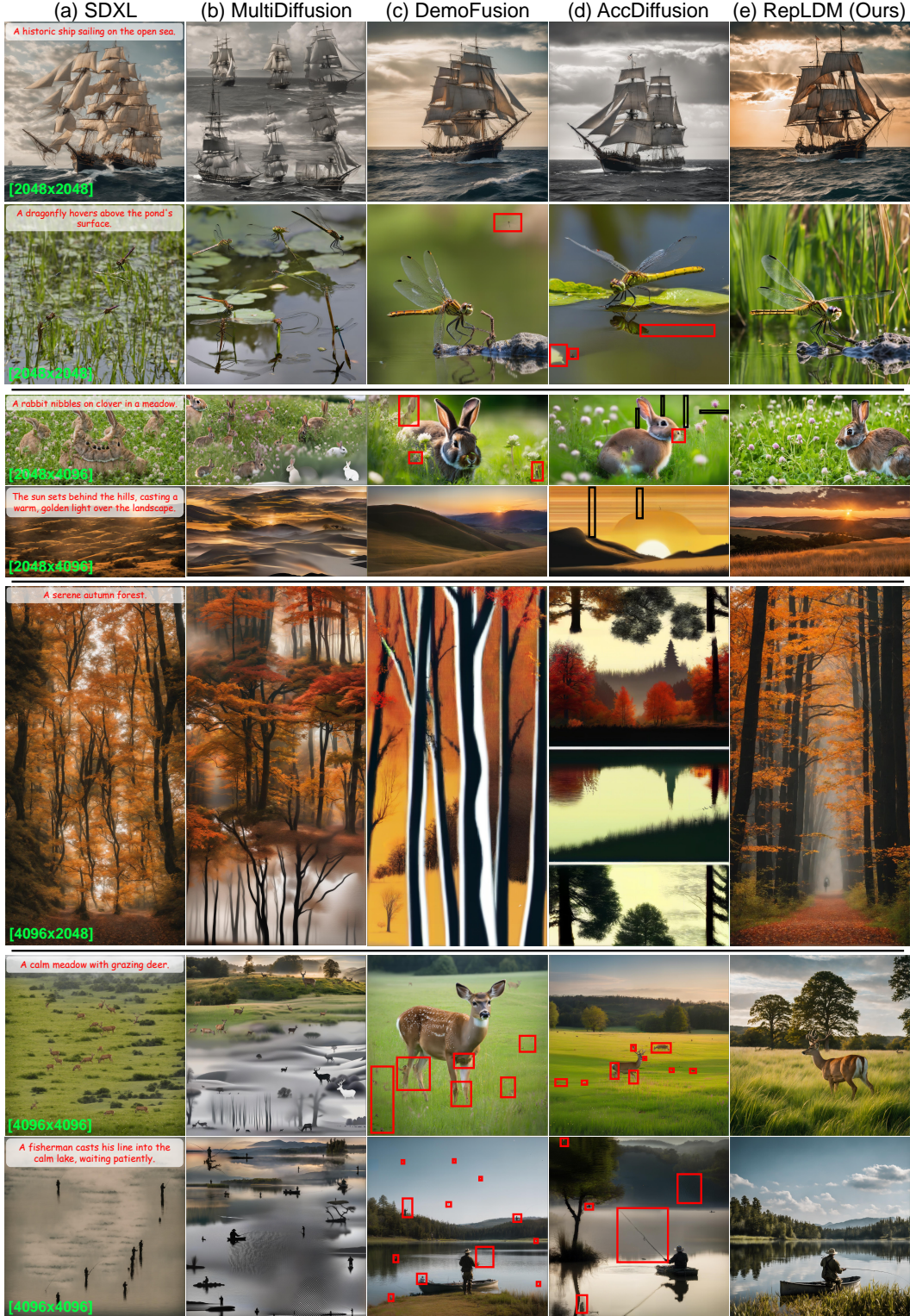


Figure 18: **Qualitative comparison with other baselines.** Zoom-in for a better view.

### C.3 Ablation on Progressive Scheduler Value

This section presents the results of quantitative ablation analysis on the progressive scheduler  $\eta_2$  in the second stage of RepLDM. We fixed  $\gamma = 0$ ,  $\eta_1 = 0$ , sampled 500 prompts, and generated 1k



Figure 19: **Further qualitative analysis of attention guidance (AG).** Using attention guidance significantly enhances image quality. The details were enriched, for example: the clouds in the sky, ripples on the water, reflections on the lake, and even the expressions in a person’s eyes. Resolution:  $2048 \times 2048$ . Best viewed **ZOOMED-IN**.

images to investigate the optimal value of the progressive scheduler. Table 7 presents the quantitative results, indicating that using an excessively large progressive scheduler may lead to a decline in image quality.

Table 7: **Quantitative ablation study of the progressive scheduler Value.** The best results are marked in **bold**, and the second best results are marked by underline.

Method	1600 × 1600					2048 × 2048				
	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
SDXL	101.56	25.78	73.67	21.23	26.87	112.64	18.44	79.03	20.61	26.55
$\eta_2 = [0.9]$	94.59	27.04	67.60	23.01	26.97	97.14	24.48	64.34	22.14	26.59
$\eta_2 = [0.8]$	93.13	28.80	65.67	24.83	26.99	93.93	26.75	60.84	23.27	26.77
$\eta_2 = [0.7]$	<b>92.05</b>	29.44	65.35	24.97	27.07	92.50	28.17	57.34	24.05	26.93
$\eta_2 = [0.6]$	92.94	30.79	64.57	24.29	27.11	91.86	30.45	55.38	24.96	26.98
$\eta_2 = [0.5]$	<u>92.73</u>	30.65	63.43	24.26	27.13	<u>91.80</u>	<u>31.18</u>	54.32	24.48	27.02
$\eta_2 = [0.4]$	93.04	<u>30.96</u>	63.33	24.77	27.14	<b>91.71</b>	<b>32.47</b>	53.72	25.16	27.03
$\eta_2 = [0.3]$	92.93	30.91	<b>63.09</b>	24.84	27.15	92.39	30.72	<u>53.32</u>	<b>26.63</b>	27.07
$\eta_2 = [0.2]$	93.09	<b>31.17</b>	<u>63.23</u>	<b>25.71</b>	<u>27.17</u>	92.71	30.45	<b>53.19</b>	<u>26.19</u>	<u>27.12</u>
$\eta_2 = [0.1]$	93.44	30.69	63.75	<u>25.18</u>	<b>27.22</b>	92.94	30.69	53.77	24.71	<b>27.18</b>

## D Ablation on the Attention Guidance Components

### D.1 Ablation on the Guidance Scale Decay Strategy

To investigate the impact of different guidance scale decay strategies, we conduct ablation studies using two additional schemes—linear decay and exponential decay—and analyze their quantitative and qualitative performance. For quantitative ablation, we generate 2k samples at a resolution of  $2048 \times 2048$  using each strategy and calculate the criteria on the SAM benchmark. Table 8 shows that different strategies yield similar results, indicating that RepLDM is not sensitive to a specific decay strategy. Fig. 20 illustrates the qualitative results. Qualitatively, these decay strategies also produce similar visual experience.

Table 8: **Ablation on the guidance scale decay strategies.** The best results are marked in **bold**, and the second best results are marked by underline.

Strategies	FID ↓	IS <sub>c</sub> ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
Linear	<u>66.2</u>	21.5	47.2	<b>20.3</b>	<b>25.4</b>
Exponential	66.8	<b>21.8</b>	<b>47.0</b>	16.3	<u>25.3</u>
Cosine (default)	<b>66.0</b>	21.0	47.4	<u>17.5</u>	25.1





Figure 20: **Qualitative ablation on guidance scale decay strategies.**

## D.2 Ablation on the Attention Calculation Paradigm

For TFSA, our objective is to remove the learnable parameters from the Self-Attention mechanism, while maintaining its computational paradigm as unchanged as possible. In TFSA,  $Q$ ,  $K$ , and  $V$  are identical. Therefore, TFSA is a totally symmetric formula. As analyzed before, this paradigm encourages the clustering of semantically related tokens, and finally leads to finer details and richer colors. An interesting question arises: if we spatially downsample  $Q$ ,  $K$ , or  $V$  before applying TFSA and reformulate it into an asymmetric paradigm (denoted as TFSA-A), would TFSA-A encourage the model to attend more explicitly from fine details to coarse structures?

To answer this question, we design an asymmetric variants, TFSA-A. Specifically, TFSA-A performs a  $2 \times 2$  pooling operation to downsample the  $K$  and  $V$  matrices before the attention calculation operation, ensuring that the output of  $\text{Softmax}(QK^T/\sqrt{d})V$  remains the of shape  $(hw) \times c$ . Table 9 shows that TFSA-A produces comparable quantitative results. In Fig. 21, we observe that although TFSA-A achieves quantitative results comparable to those of TFSA, its visual quality is significantly inferior. In fact, TFSA-A tends to reduce image details. This aligns with our hypothesis: the  $2 \times 2$  pooling acts as a low-pass filter, causing the loss of fine-grained information in the latent representations and leading the model to focus more on low-frequency structures.

Table 9: **Ablation on the attention calculation paradigm.** The best results are marked in **bold**, and the second best results are marked by underline.

Paradigm	FID ↓	IS <sub>c</sub> ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
w/o guidance	66.8	21.6	47.5	17.4	<b>25.3</b>
w/ TFSA-A	67.4	<b>22.6</b>	47.9	<b>20.4</b>	<b>25.3</b>
w/ TFSA	<b>66.0</b>	21.0	<b>47.4</b>	<u>17.5</u>	<u>25.1</u>

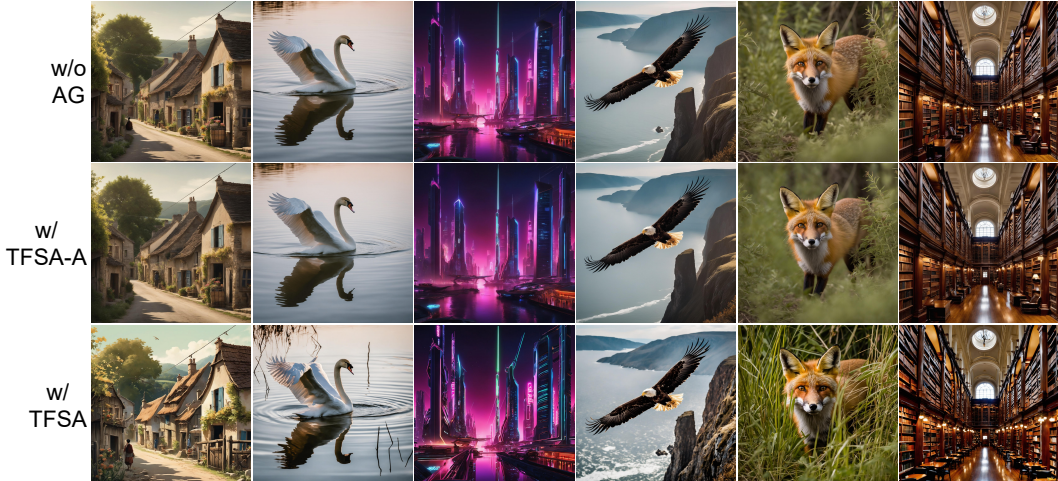


Figure 21: **Ablation on the attention calculation paradigm.** Resolution:  $2048 \times 2048$ .



## E Further Model Efficiency Analysis

**Computational complexity analysis of TFSA.** Note that attention guidance is only applied during the first stage of generation. Assume we have a HR image  $x_0$  with a resolution of  $H \times W \times C$ . we encode the image  $x_0$  into latent space and obtain latent representation  $z_0 \in \mathbb{R}^{h \times w \times c}$ . Before feeding  $z_0$  into TFSA, we reshape it to a  $(hw) \times c$  matrix. The computation of TFSA follows a formulation similar to that of self-attention:  $\text{Softmax}(z_0 z_0^T / \sqrt{c}) z$ . Thus, the computational complexity of TFSA is  $O((hw)^2 c)$ . Taking SDXL as an example, the training resolution is  $H = 1024, W = 1024$ . After VAE encoding,  $c = 4, h = H/8 = 128, w = W/8 = 128$ . For each denoising step, the FLOPs of TFSA is approximately  $2 \times (h \times w)^2 \times c$ , which is around 2.15 GFLOPs—negligible compared to the FLOPs of the denoising network (several TFLOPs per step).

**How does pixel space upsampling accelerate generation?** To answer this question, we analyze the time consumption of each component in DemoFusion and RepLDM when generating images at the resolution of  $4096 \times 4096$ .

Table 10: The time consumption of DemoFusion when generating  $4096 \times 4096$  resolution images.

Metric	Denoise 1024	Denoise 2048	Denoise 3072	Denoise 4096	Decode 4096	Total
number of steps	50	50	50	50	-	200
Time (s)	12	185	480	901	106	1684

Table 11: The time consumption of RepLDM when generating  $4096 \times 4096$  resolution images. The intermediate encoding/decoding operations are highlighted in underline.

Metric	Denoise 1024	Decode 1024	Encode 3304	Denoise 3304	Decode 3304	Encode 4096	Denoise 4096	Decode 4096	Total
number of steps	50	-	-	5	-	-	10	-	65
Time (s)	12	<u>0</u>	<u>12</u>	20	<u>64</u>	<u>11</u>	118	106	343

Table 10 shows that denoising at high resolutions is a time-consuming process. DemoFusion requires substantial generation time because it performs the full denoising process at high resolutions. Note that, compared with the cost of the denoising process at high resolutions, the costs of encoding and decoding are negligible. Table 11 shows that RepLDM significantly accelerates generation by substantially reducing the number of denoising steps at high resolutions. This is because RepLDM performs pixel space upsampling through multiple rounds of encoding and decoding, producing high-quality low-resolution images that serve as better initialization. As a result, RepLDM can significantly reduce the number of sampling steps required for HR generation, thereby accelerating the process. Moreover, Table 11 shows that the additional overhead from multiple intermediate encoding and decoding operations is also relatively minor compared to the total generation cost.

**Further efficiency comparison across different models.** To provide a more comprehensive assessment of model efficiency, we further report the NFE and FLOPs of different models when generating a single image at resolutions of  $2048 \times 2048$  and  $4096 \times 4096$ . Tables 12 and 13 show that RepLDM significantly reduces the NFE and FLOPs required for inference by decreasing the number of denoising steps at high resolutions, thereby substantially reducing the time needed to generate HR images.

Table 12: Inference cost of generating a  $2048 \times 2048$  Image for different models.

Model	SDXL [32]	MultiDiff. [1]	ScaleCrafter [11]	HiDiff. [51]	UG [18]	DemoFusion [5]	AccDiff. [25]	RepLDM
NFE	50	50	50	50	80	100	100	60
TFLOPs	3010	5420	2437	1857	3608	9015	8597	1140
Time (min)	1.0	3.0	1.0	0.8	1.8	3.0	3.0	0.6

Table 13: Inference cost of generating a  $4096 \times 4096$  Image for different models.

Model	SDXL [32]	MultiDiff. [1]	ScaleCrafter [11]	HiDiff. [51]	UG [18]	DemoFusion [5]	AccDiff. [25]	RepLDM
NFE	50	50	50	50	80	200	200	65
TFLOPs	12026	29566	9759	5211	12624	72167	74225	7140
Time (min)	8.0	15.0	19.0	3.4	11.1	25.0	26.0	5.7

## F RepLDM Algorithm

The implementation details of RepLDM can be found in Algorithm 1, and further information is available in our code repository.

---

**Algorithm 1** RepLDM Inference Pipeline

---

**Require:** The number of inference time steps of the first stage  $T_0$ ; progressive scheduler  $\eta_2$ ; attention guidance scale  $\gamma$ ; attention guidance delay rate  $\eta_1$ ; the decay factor  $\beta$ ; target image size tuple  $(H', W')$ ; the denoising model  $\mathcal{F}$ ; denoising model's training resolution tuple  $(H, W)$ ; VAE encoder  $\mathcal{E}$ ; VAE decoder  $\mathcal{D}$ ; noise scheduler's hyper-parameter list  $\bar{\alpha}_{1:T_0}$ .

```
1: Initialization:
2:  $\mathbf{z}_{T_0}^{(0)} = \epsilon \sim \mathcal{N}(0, \mathbf{I})$  {Sampling from Standard Gaussian Distribution}
3:  $n_{\text{stages}} = \text{length}(\eta_2) + 1$  {Get the total number of denoising stages}
4:  $r' = \frac{H'}{W'}$  {Keep the aspect ratio and number of pixels unchanged}
5:  $H^{(0)} = \text{ceil}(\sqrt{H \times W \times r'})$ 
6:  $W^{(0)} = \text{ceil}(\sqrt{\frac{H \times W}{r'}})$ 
7:  $H^{(n)} = H'$ 
8:  $W^{(n)} = W'$ 
9:  $\text{area}_{\text{list}} = \text{linspace}(H^{(0)} \times W^{(0)}, H^{(n)} \times W^{(n)}, n_{\text{stages}})$  {Upsampling according to the number of pixels}
10:  $H_{\text{list}} = [\text{ceil}(\sqrt{i \times r'}) \text{ for } i \text{ in } \text{area}_{\text{list}}]$  {Get the height and width of each stage}
11:  $W_{\text{list}} = [\text{ceil}(\sqrt{i/r'}) \text{ for } i \text{ in } \text{area}_{\text{list}}]$ 
12:  $k_{\text{denoising}} = [T_0]$  {Get the number of denoising steps for each stage}
13:  $k_{\text{denoising}}.\text{extend}([i \times T_0 \text{ for } i \text{ in } \eta_2])$ 
14:  $k = T_0 \times \eta_1$  {Obtain the number of delay steps}
15:  $\gamma_{\text{list}} = [\gamma(\frac{\cos(\frac{T-k-i}{T-k}\pi)+1}{2})^\beta \text{ for } i = 1, \dots, T-k]$  {Obtain the guidance scale for each step}
16: Denoising:
17: for  $s = 0, \dots, n_{\text{stages}} - 1$  do
18:    $n_{\text{steps}} \leftarrow k_{\text{denoising}}[s]$ 
19:   if  $s \geq 1$  then
20:      $\mathbf{x}^{(s)} \leftarrow \text{upsample}(\mathbf{x}^{(s-1)}, H_{\text{list}}[s], W_{\text{list}}[s])$  {Upsampling in pixel space}
21:      $\mathbf{z}_0^{(s)} \leftarrow \mathcal{E}(\mathbf{x}^{(s)})$ 
22:      $\mathbf{z}_{n_{\text{steps}}}^{(s)} \sim \mathcal{N}(\sqrt{\bar{\alpha}[n_{\text{steps}}]}\mathbf{z}_0^{(s)}, (1 - \bar{\alpha}[n_{\text{steps}}])\mathbf{I})$ 
23:   end if
24:   for  $t = n_{\text{steps}} - 1, \dots, 0$  do
25:      $\mathbf{z}_t^{(s)} \leftarrow \mathcal{F}(\mathbf{z}_{t+1}^{(s)}, t+1)$  {Denoising}
26:     if  $s == 0$  and  $t \leq T - 1 - k$  then
27:        $\mathbf{z}_t^{(s)} \leftarrow \gamma_{\text{list}}[t]\text{PFSA}(\mathbf{z}_t^{(s)}) + (1 - \gamma_{\text{list}}[t])\mathbf{z}_t^{(s)}$  {Attention Guidance}
28:     end if
29:   end for
30:    $\mathbf{x}^{(s)} \leftarrow \mathcal{D}(\mathbf{z}_0^{(s)})$  {Obtain the pixel space image}
31: end for
```

---

## G Robustness Analysis

In this section, we conduct a robustness analysis to complement the experiments in §4.2, providing a more comprehensive evaluation of the models' performance. Our robustness analysis is conducted from two perspectives: (i) we vary the random seeds and repeat each experiment three times to compute the mean and standard deviation of all results; (ii) we randomly sample 20k HR images from the HR subset of LAION-5B dataset [36] to construct a new benchmark for evaluating the models' generalization performance. Since HR generation requires substantial computational resources, we analyze the four best-performing models from Table 1, *i.e.*, HiDiffusion, DemoFusion, AccDiffusion, and RepLDM.

**Analysis on the SAM benchmark.** We maintain the exact experimental settings as in §4.2 and conduct the analysis at resolutions of  $2048 \times 2048$  and  $4096 \times 4096$ . Table 14 shows that RepLDM continues to exhibit superior performance across the repeated experiments.

Table 14: **Robustness analysis on the SAM benchmark.** The best results are marked in **bold**.

Method	2048 × 2048					4096 × 4096				
	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑	FID ↓	IS ↑	FID <sub>c</sub> ↓	IS <sub>c</sub> ↑	CLIP ↑
HiDiff. [51]	80.29±0.57	17.18±0.40	63.55±0.63	15.26±0.76	24.95±0.04	144.24±0.84	12.71±0.14	146.62±0.32	7.48±0.28	21.18±0.05
DemoF. [5]	71.89±0.60	22.10±0.37	53.58±0.22	19.21±0.27	25.21±0.01	101.83±0.49	20.81±0.11	63.60±0.46	14.92±1.24	<b>24.75±0.03</b>
AccDiff. [25]	71.37±0.48	21.21±0.32	53.04±0.33	19.24±1.72	25.13±0.01	102.41±1.40	19.88±0.24	65.86±0.17	12.73±0.71	24.65±0.02
RepLDM	<b>66.08±0.02</b>	<b>22.13±0.74</b>	<b>47.31±0.11</b>	<b>20.38±2.03</b>	<b>25.30±0.12</b>	<b>91.46±0.61</b>	<b>21.63±0.46</b>	<b>58.93±0.20</b>	<b>15.02±0.16</b>	24.62±0.02

**Analysis on the LAION-5B benchmark.** Considering that only 1K samples were used for the  $4096 \times 4096$  resolution in §4.2, which may lead to unstable metric evaluations, we double the number of samples to 2k for this resolution in the current experiment. Regarding evaluation metrics, since IS may lead to high variances beyond ImageNet, we follow some recent studies and adopt Kernel Inception distance (KID) for more accurate evaluation [17, 33]. Table 15 shows that on the LAION benchmark, RepLDM still demonstrates superior performance, surpassing previous SOTA models across all metrics.

Table 15: **Robustness analysis on the LAION-5B benchmark.** The best results are marked in **bold**. Since the magnitude of KID is relatively small, we multiply its mean and standard deviation by  $10^3$ .

Method	2048 $\times$ 2048					4096 $\times$ 4096				
	FID $\downarrow$	KID $\downarrow$	FID <sub>c</sub> $\downarrow$	KID <sub>c</sub> $\downarrow$	CLIP $\uparrow$	FID $\downarrow$	KID $\downarrow$	FID <sub>c</sub> $\downarrow$	KID <sub>c</sub> $\downarrow$	CLIP $\uparrow$
HiDiff. [51]	48.17 $\pm$ 0.41	8.06 $\pm$ 0.20	36.26 $\pm$ 0.37	10.93 $\pm$ 0.11	23.16 $\pm$ 0.03	92.81 $\pm$ 0.78	35.36 $\pm$ 0.60	120.26 $\pm$ 0.91	103.45 $\pm$ 0.27	18.55 $\pm$ 0.06
DemoF. [5]	34.15 $\pm$ 0.31	4.50 $\pm$ 0.05	21.38 $\pm$ 0.17	6.80 $\pm$ 0.06	25.44 $\pm$ 0.02	37.03 $\pm$ 0.27	5.71 $\pm$ 0.14	30.77 $\pm$ 0.36	16.12 $\pm$ 0.22	25.12 $\pm$ 0.04
AccDiff. [25]	34.49 $\pm$ 0.31	4.92 $\pm$ 0.08	22.71 $\pm$ 0.17	8.57 $\pm$ 0.11	24.90 $\pm$ 0.02	38.56 $\pm$ 0.23	7.21 $\pm$ 0.20	38.85 $\pm$ 0.29	20.87 $\pm$ 0.20	24.46 $\pm$ 0.01
RepLDM	<b>34.08</b> $\pm$ 0.25	<b>4.18</b> $\pm$ 0.04	<b>20.30</b> $\pm$ 0.30	<b>4.87</b> $\pm$ 0.13	<b>25.78</b> $\pm$ 0.03	<b>34.01</b> $\pm$ 0.26	<b>4.13</b> $\pm$ 0.05	<b>23.08</b> $\pm$ 0.26	<b>12.08</b> $\pm$ 0.13	<b>25.88</b> $\pm$ 0.04