BlockDance: Reuse Structurally Similar Spatio-Temporal Features to Accelerate Diffusion Transformers

Supplementary Material

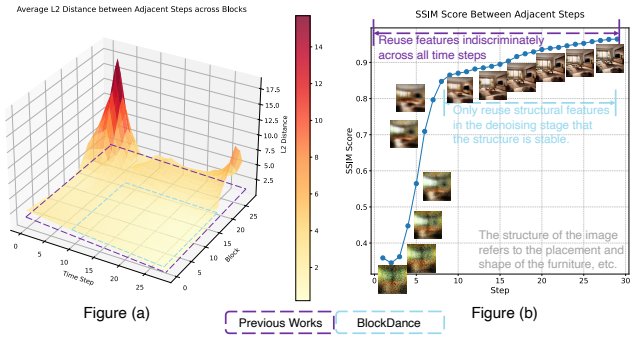

Figure 11. On the left: the 3D surface plot of (step, block, L2 distance). On the right: SSIM score of the images x_0^t at adjacent time steps. BlockDance only reuses the computations on structural features after the structure stabilizes. Zoom in for details.

5.1. More justification for the design in BlockDance

We supplement the (step, block, L2 distance) 3D surface plot, as shown in (a) of Figure 11. BlockDance focuses on reusing high-similarity spatio-temporal features within the blue box. In contrast, other methods reuse features indiscriminately across all spatio-temporal zones framed by the purple box, which leads to quality degradation caused by reusing low-similarity features. To further demonstrate that the features within the blue box are primarily structurally similar, we directly calculate $x_0^t = VAE_{decoder}(\frac{z_t - \sqrt{1-\bar{\alpha}_t}\cdot\epsilon_\theta(z_t,t)}{\sqrt{\bar{\alpha}_t}})$ based on the noise predicted at each step and then compute the SSIM score of the images x_0^t at adjacent time steps to measure structural similarity. As shown in (b) of Figure 11, compared to other methods that reuse features indiscriminately, we reuse the computations focusing on structural features (*i.e.* shallow and middle blocks in DiT) after the structure stabilizes, thereby maintaining high consistency with the original content.

5.2. Additional Experiments

Accelerate at any number of steps. The acceleration paradigm we proposed is complementary to other acceleration techniques and can be used on top of them for further enhancement. Here, we validate the performance of BlockDance across different sampling steps for each model. As demonstrated in Tables 5, 6, and 7, BlockDance effectively accelerates the process across various step counts while maintaining the quality of the generated content.

Accelerate SD3 for text-to-image generation. To validate the effectiveness of our proposed paradigm across different DiT architecture variants, we apply BlockDance to MMDiT-based DiT models [8, 18], such as Stable Diffusion 3 [8]. The results are conducted on the 25k COCO2017 validation set, as shown in Table 8. The experimental results indicate that with $N = 2$, BlockDance accelerates SD3 by 25.3% while maintaining comparable image quality, both in terms of visual aesthetics and prompt following. Different speed-quality trade-offs can be modulated by N.

COCO2017	PixArt-α		BlockDance (N=2)	
	Latency ↓	FID ↓	Latency ↓	FID ↓
step=20	2.02	30.79	1.51 (↑ 24.8%)	30.87
step=30	3.10	30.41	2.31 (↑ 25.4%)	30.69
step=40	4.15	30.19	3.08 (↑ 25.8%)	30.35

Table 5. Accelerate PixArt-α at any number of steps. All the methods adopt the DPM-Solver sampler.

ImageNet	DIT-XL/2		BlockDance (N=2)	
	Latency ↓	FID ↓	Latency ↓	FID ↓
step=30	1.07	16.15	0.67 (↑ 37.3%)	16.06
step=40	1.43	16.04	0.90 (↑ 37.1%)	15.91
step=50	1.79	15.89	1.12 (↑ 37.4%)	15.70

Table 6. Accelerate DiT-XL/2 at any number of steps. All the methods here adopt the DDIM sampler.

MSR-VTT	Open-Sora		BlockDance (N=2)	
	Latency ↓	FVD ↓	Latency ↓	FVD ↓
step=50	27.72	582.91	18.16 (↑ 34.5%)	585.21
step=75	36.53	561.72	23.78 (↑ 34.9%)	562.83
step=100	44.99	548.72	29.32 (↑ 34.7%)	550.22

Table 7. Accelerate Open-Sora at any number of steps. All the methods here adopt the DDIM sampler.

Model	Latency (s) ↓	IR ↑	Pick ↑	IS ↑	CLIP ↑	FID ↓	SSIM ↑
SD3	4.35	1.01	22.49	41.52	0.334	26.95	-
BlockDance(N=2)	3.25 (↑25.3%)	**1.00**	**22.45**	**40.89**	**0.334**	**27.52**	**0.96**
BlockDance(N=3)	2.99 (↑31.3%)	0.99	22.42	40.52	0.334	27.74	0.95
BlockDance(N=4)	2.74 (↑37.0%)	0.98	22.34	39.53	0.334	28.42	0.92

Table 8. Text-to-image generation on SD3.

More qualitative results. To comprehensively verify the method we proposed, we present additional qualitative results for each DiT model, as indicated in Figures 12, 13, and 14. Our method maintains high-quality content with a high degree of consistency with the content generated by the original models, while achieving significant acceleration.

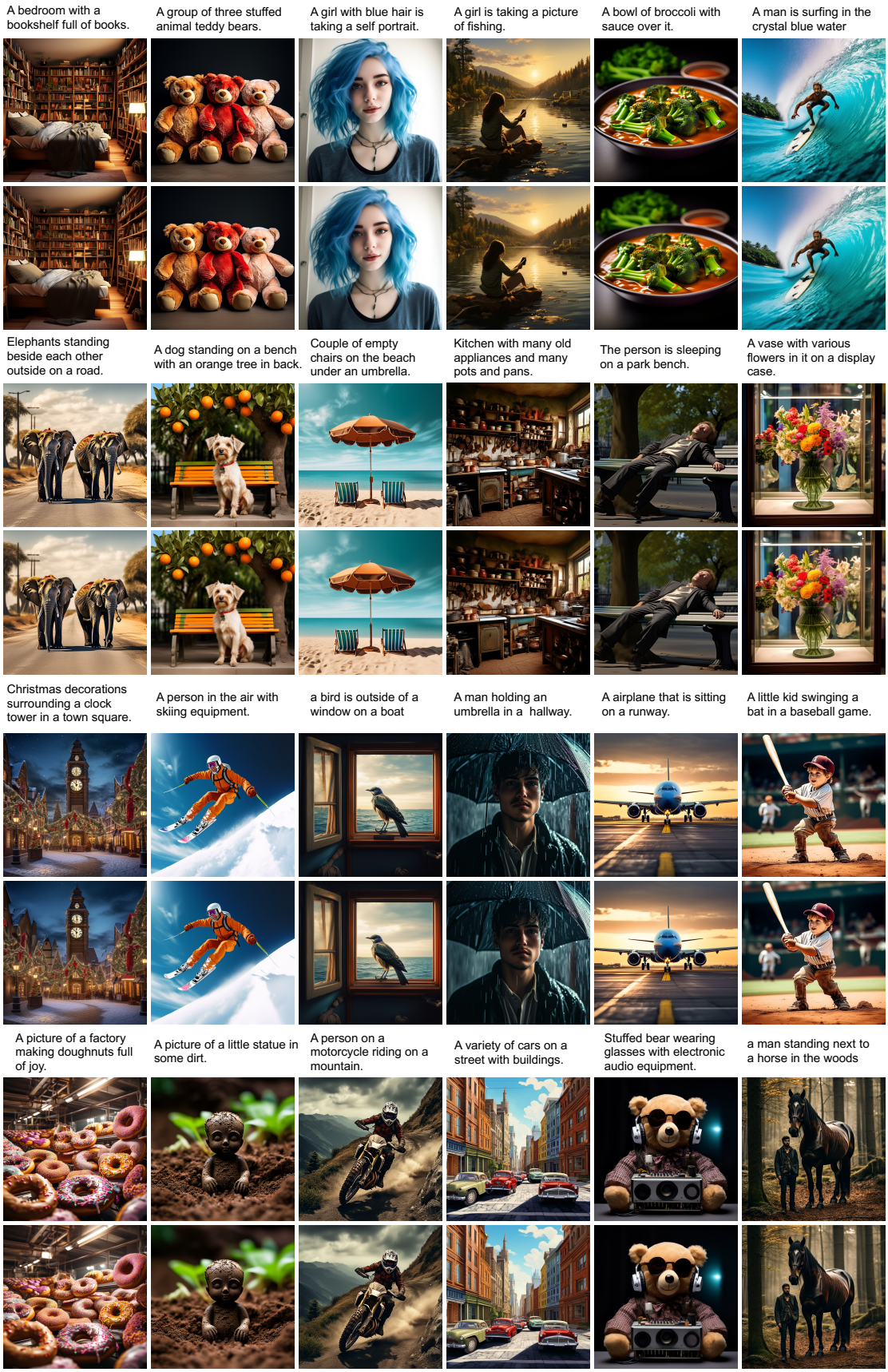

Figure 12. PixaArt-α: Samples with 30 DPM-Solver steps (upper row) and 30 DPM-Solver steps + BlockDance with $N = 2$ (lower row). Our method speeds up 25.4% while maintaining the visual aesthetics and prompt following. Here, prompts are selected from the COCO2017 validation set.

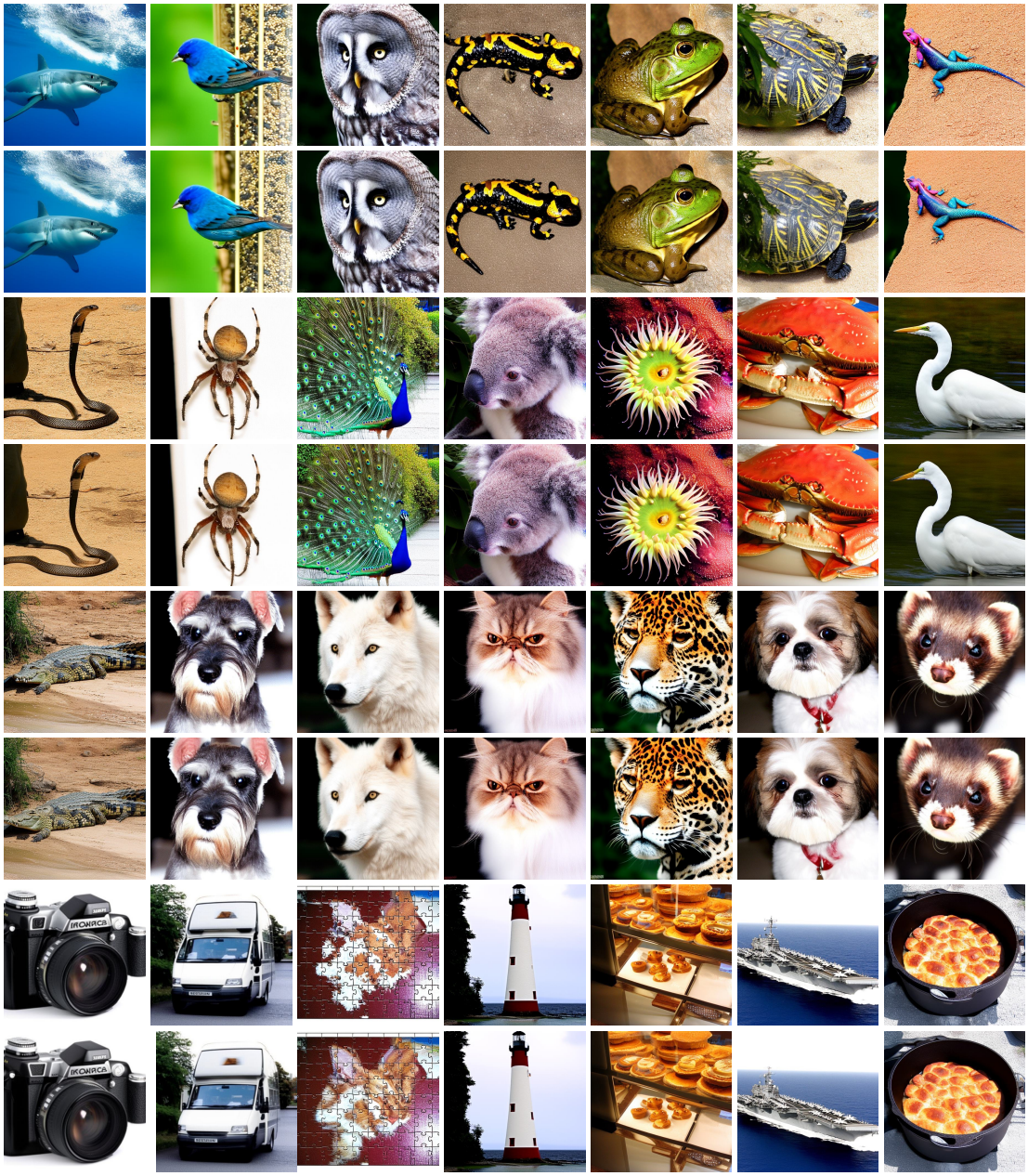

Figure 13. DiT-XL/2 for ImageNet: Samples with 50 DDIM steps (upper row) and 50 DDIM steps + BlockDance with $N = 2$ (lower row). Our method achieves a 37.4% acceleration while maintaining image quality.

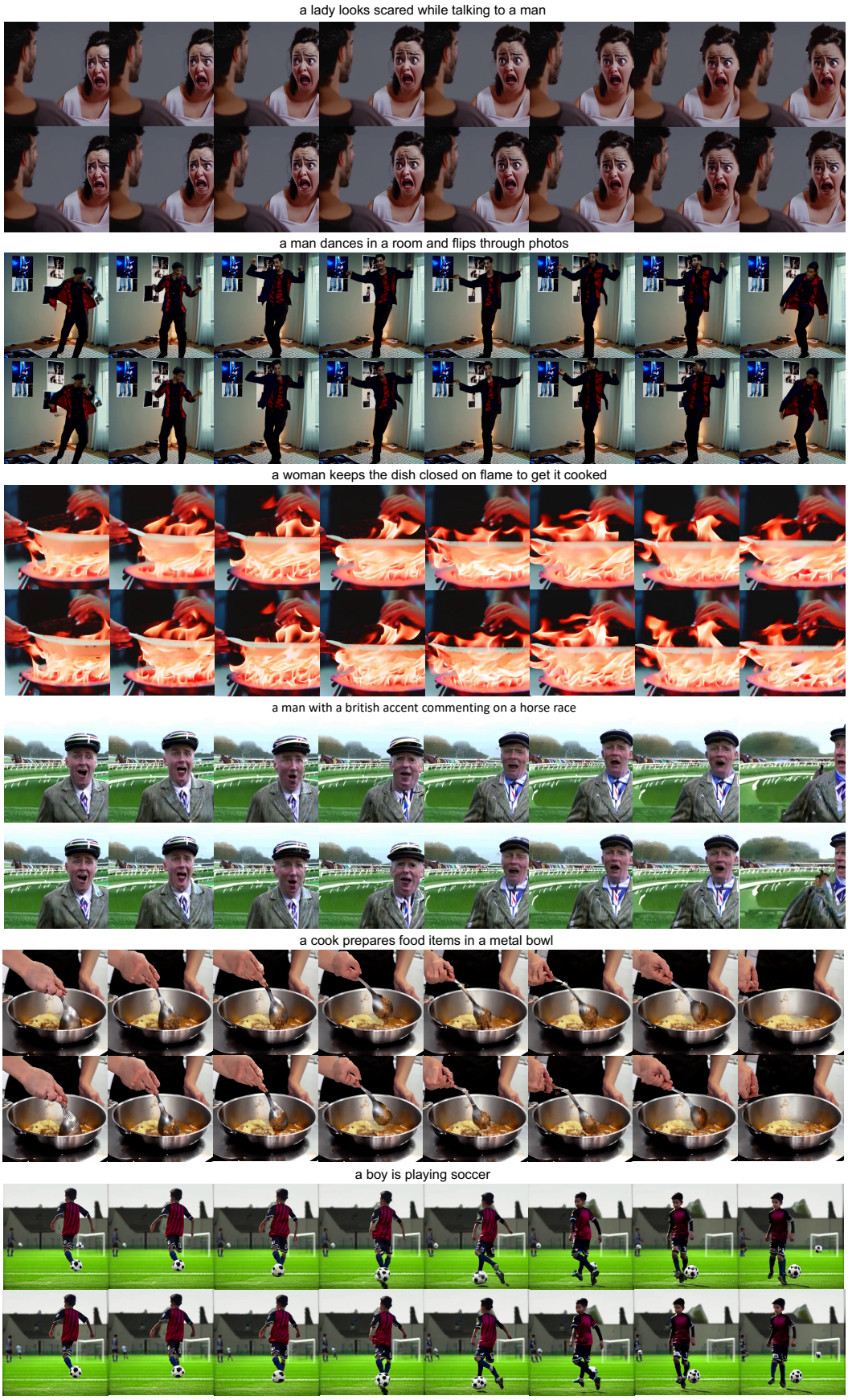

Figure 14. Open-Sora: Samples with 100 DDIM steps (upper row) and 100 DDIM steps + BlockDance with $N = 2$ (lower row). Our method achieves a 34.8% acceleration while maintaining visual quality and high motion consistency with the original video.