Decouple and Track: Benchmarking and Improving Video Diffusion Transformers for Motion Transfer

Supplementary Material

Overview.

1. More Qualitative Results

We show more qualitative results in Fig. 1, Fig. 2, and Fig. 3. Each source video is combined with two newly generated videos.

2. More Qualitative Comparison Results

We present additional qualitative comparisons in Fig. 4, Fig. 5, and Fig. 6. Our method accurately transfers complex motions while enabling flexible foreground and background control. These results demonstrate the generality of our motion transfer approach.

3. More quantitative Results

V2VBench. We further evaluate our approach on V2VBench. As shown in Tab. 1, DeT demonstrates a clear advantage in both motion and video-text alignment—key aspects of the motion transfer task.

User Study. We conduct a user preference evaluation. As shown in Tab. 5, DeT demonstrates a clear advantage across all assessed aspects.

4. Implementation Details

HunyuanVideo implementation details. We train DeT on HunyuanVideo for 500 steps using the AdamW optimizer with a learning rate of 1e-5 and a weight decay of 1e-2. The learning rate is linearly warmed up over the first 100 steps. The loss weight parameters λ_{DL} and λ_{TL} are set to 1.0 and 0.1, respectively. For the motion module, the mid-dimension is set to 128, and the kernel size is configured to 5. During training, we integrate our shared temporal kernel into all DiT blocks. During inference, we remove it from the last 40 blocks (66%) - all of the single-stream DiT blocks. We perform 30 steps denoising using the Flow Matching scheduler. The generated videos have a resolution of $49 \times 512 \times 768$. The training process takes approximately 1.5 hours on a single NVIDIA A100 80GB GPU.

Table 1. V2VBench results (higher is better).

Method	Motion Align ↑	Video Txt ↑	Frames Txt ↑	Object Cons. ↑	Semantic Cons. ↑	Video Qual. ↑	Frames Qual. ↑	Frames Pick ↑
MotionDirector	−3.09	20.92	**27.85**	0.94	0.95	0.62	4.98	0.26
TokenFlow	−1.57	20.76	27.52	0.95	0.94	**0.72**	**5.07**	0.25
DeT (Ours)	**-0.83**	**31.50**	27.84	**0.96**	**0.97**	0.63	4.99	**0.29**

Step-Video-T2V implementation details. We train DeT on Step-Video-T2V for 500 steps using the AdamW optimizer with a learning rate of 2e-5 and a weight decay of 1e-2. The learning rate is linearly warmed up over the first 100 steps. The loss weight parameters λ_1 and λ_2 are set to 1.0 and 0.1, respectively. For the shared temporal kernel, the mid-dimension is set to 256, and the kernel size is configured to 5. During training, we integrate the shared temporal kernel into all DiT blocks. During inference, we remove it from the last 32 blocks (66%). We perform 50-step denoising using the Flow Matching (FM) scheduler with a classifier-free guidance scale of 9.0. The generated videos have a resolution of $49 \times 512 \times 768$. The entire training process takes approximately two hours on a single NVIDIA A100 80GB GPU.

Baselines implementation details. We implement MotionInversion [5] on CogVideoX-5B [10] by injecting temporal embeddings $E_t \in \mathbb{R}^{t \times c}$ and spatial embeddings $E_s \in \mathbb{R}^{h \times w \times c}$ into the 3D full attention mechanism. Specifically, we inject E_t into the query and key, while E_s is applied to the value. We train MotionInversion for 500 steps using the AdamW optimizer with a learning rate of 1e-3 and a weight decay of 1e-2, injecting embeddings into all DiT blocks. For DreamBooth, we use the LoRA variant, setting the LoRA rank to 128 and injecting LoRA weights into the query, key, value, and output linear layers. We train with a learning rate of 1e-4 and a weight decay of 1e-2. For DMT, we apply DDIM inversion to each video, which takes approximately one hour. During inference, we extract features from the 28th DiT block and compute SMM [11] via spatial pooling and first-order differencing. The optimization learning rate is set to 1e-2, and we optimize the latents for 20 steps. For other U-Net-based methods, we use their official implementations and settings.

5. Additional Analysis and Ablation Study

5.1. Analysis for drop layers

During inference, we drop the shared temporal kernel in the last 65% of DiT blocks. Experimental results show that this dropping strategy enhances text controllability. We attribute this to the fact that the DiT model relies more on the earlier layers when generating videos, as shown in Fig. 7. This

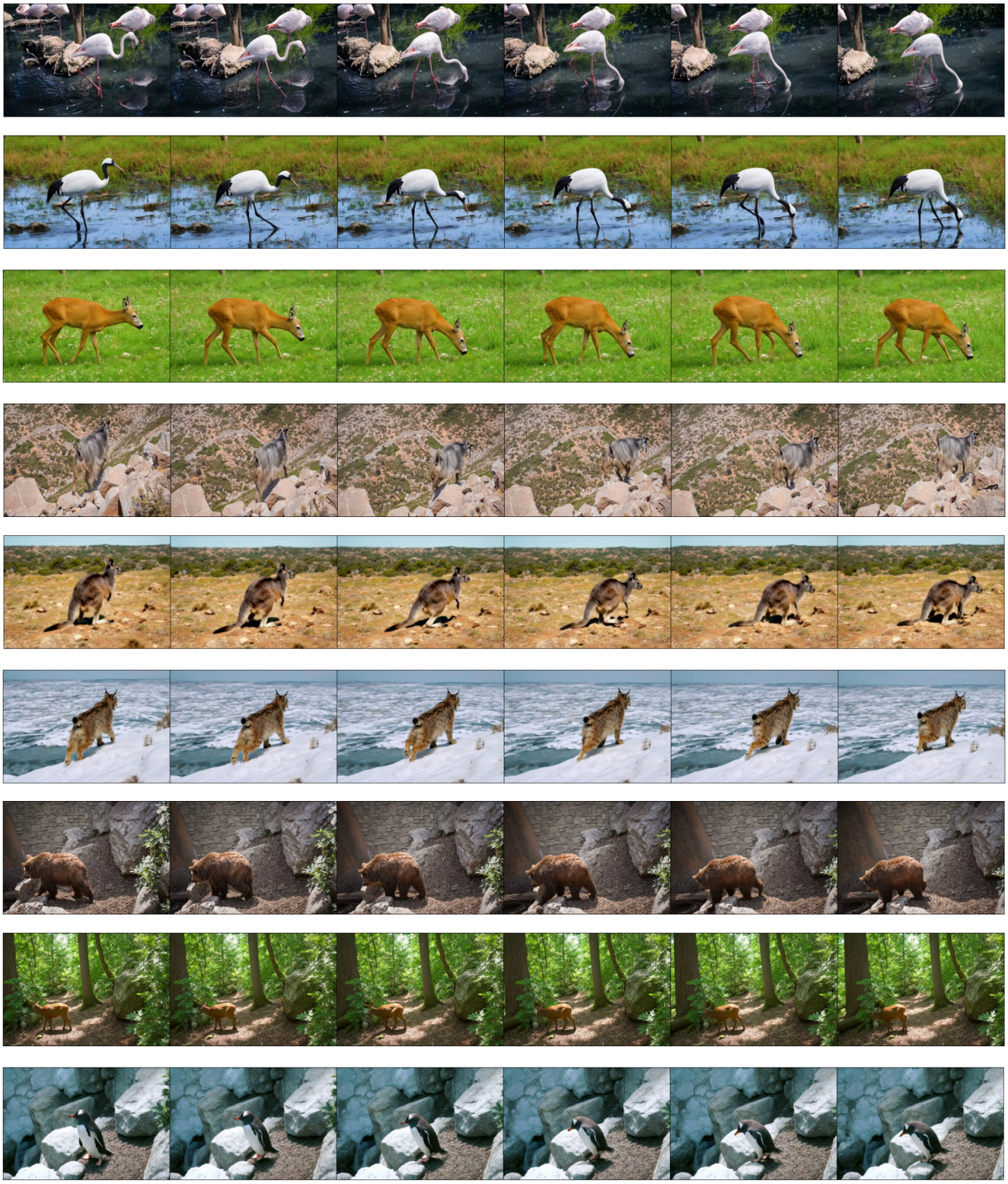

Figure 1. **More Qualitative Results.** There are three set of results. Each source video is combined with two newly generated videos on the bottom. Generated by DeT with Step-Video-T2V

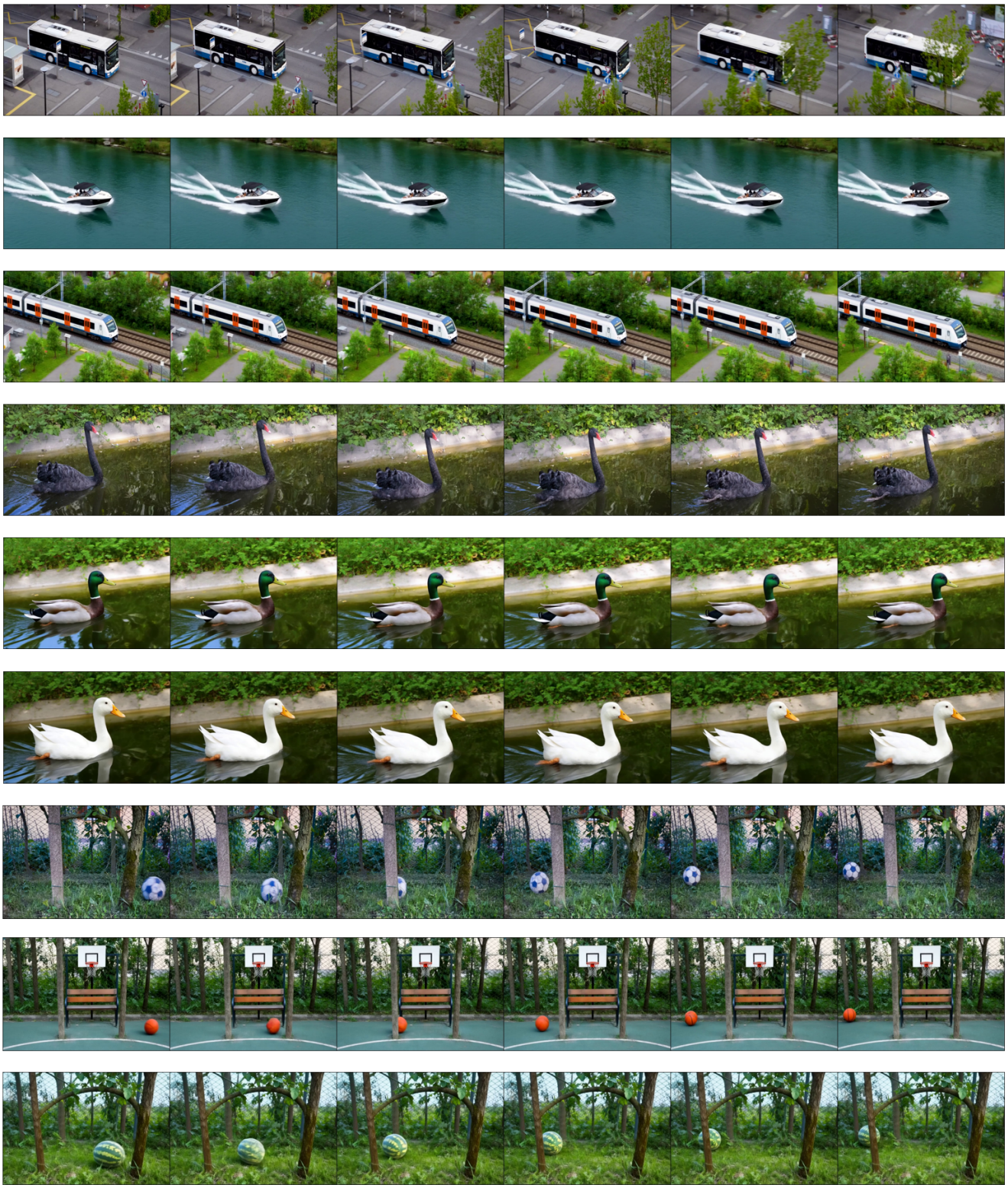

Figure 2. **More Qualitative Results.** There are three set of results. Each source video is combined with two newly generated videos on the bottom. Generated by DeT with HunyuanVideo

Figure 3. **More Qualitative Results.** There are three set of results. Each source video is combined with two newly generated videos on the bottom. Generated by DeT with Step-Video-T2V

Figure 4. **More Qualitative Comparision Results.** Our result is generated by DeT with CogVideoX-5B

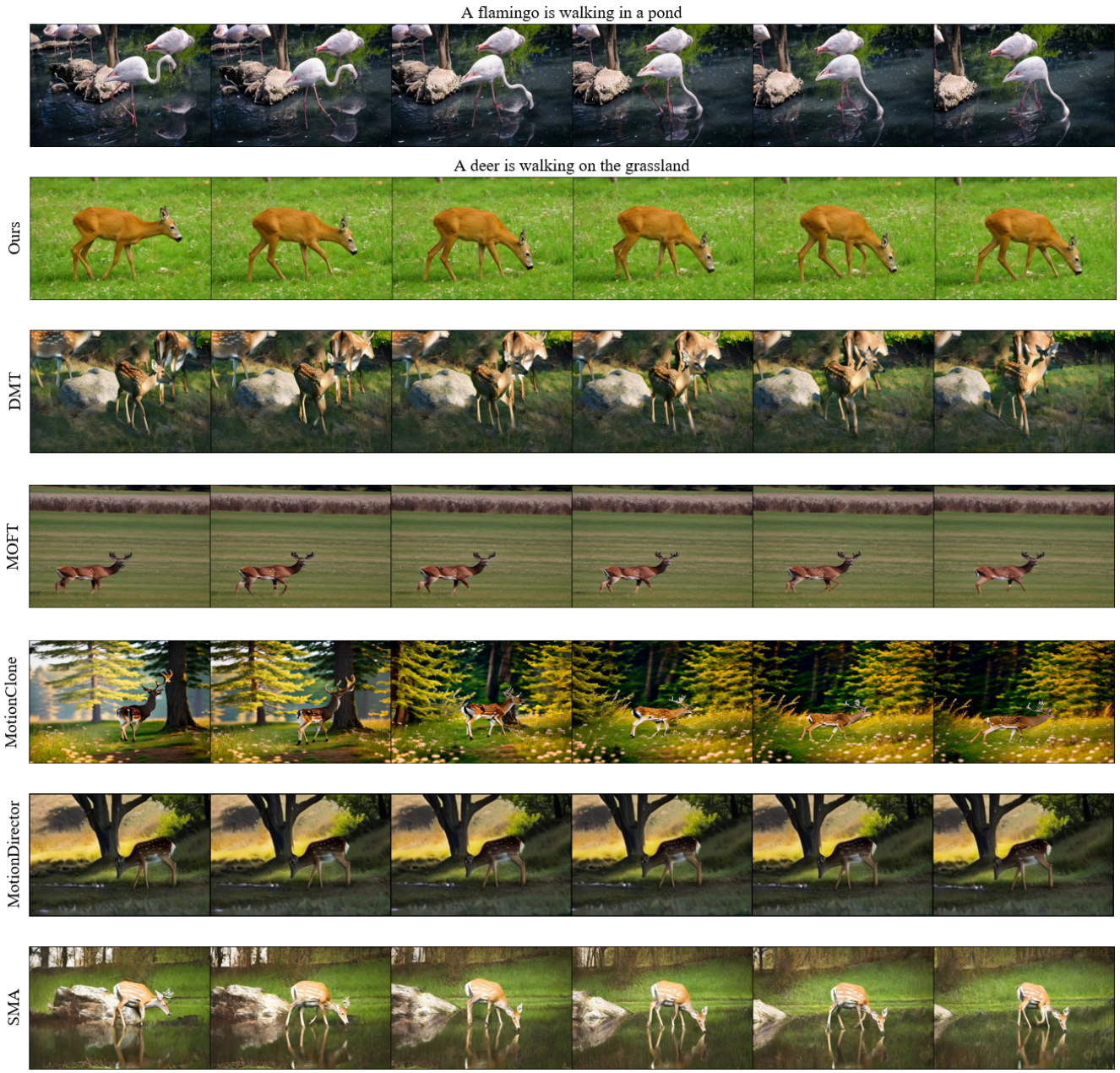

Figure 5. **More Qualitative Comparision Results.** Our result is generated by DeT with Step-Video-T2V

suggests that the shared temporal kernel inserted in the later layers may introduce redundant learnable parameters that

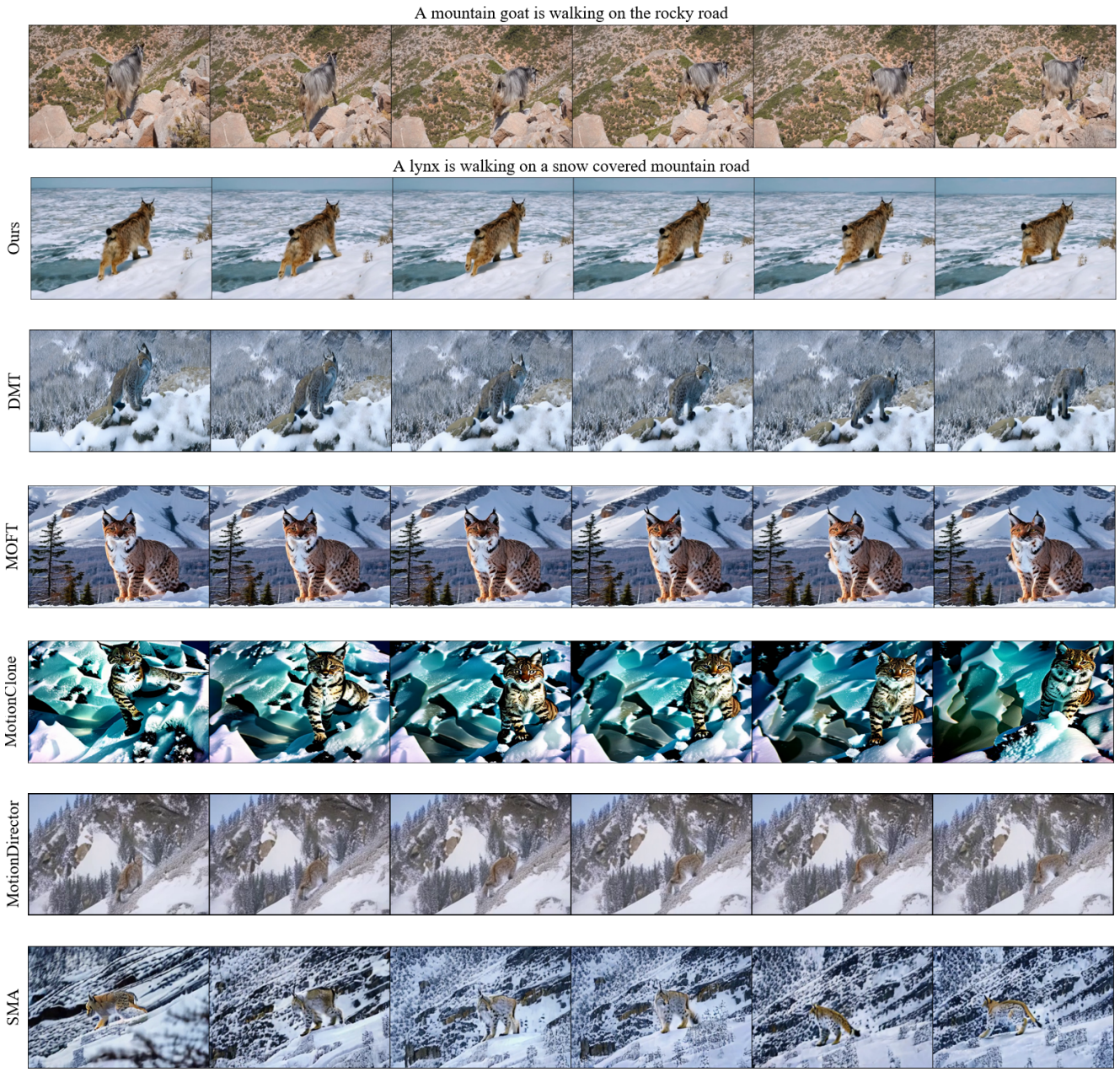

Figure 6. **More Qualitative Comparision Results.** Our result is generated by DeT with Step-Video-T2V

capture information unrelated to motion.

Furthermore, the visualization of the attention map in Fig. 8 reveals that in the earlier layers, the DiT model exhibits a mechanism similar to temporal self-attention, whereas the later layers resemble spatiotemporal attention. Spatiotemporal attention is less effective at decoupling motion from appearance, which further supports the decision to drop the temporal kernel in the later layers. Additionally, we observe that applying this layer-dropping strategy consistently improves performance across all three DiT models.

5.2. Analysis for denoising loss weight

Our ablation on the denoising loss weight λ_{DL} (Fig. 9, Tab. 4), shows that reducing λ_{DL} collapses background structure and boosts edit fidelity, but simultaneously degrades motion fidelity.

5.3. DeT with HunyuanVideo

Table 2 presents comprehensive ablation studies conducted on various hyperparameters within the DeT framework with HunyuanVideo.

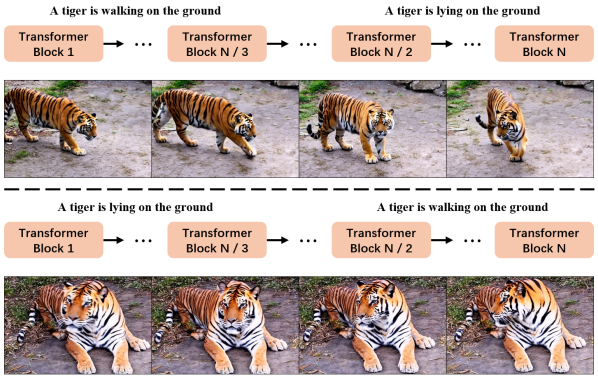

Figure 7. **Analysis of dropping layer strategy.** The generation results of DiT models rely more on features from earlier layers. The tiger's motion is determined by the text prompt in these early layers.

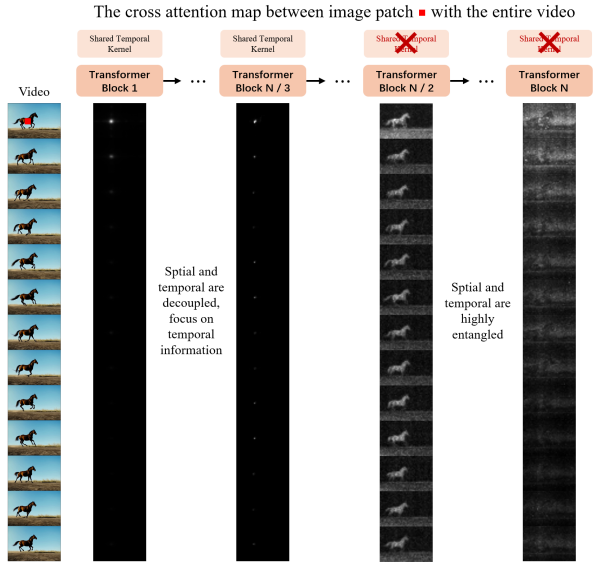

Figure 8. **Anaylsis of dropping layer strategy.** The attention map also supports our dropping layer strategy. As the later layers' attention maps show entangled spatial and temporal information, decoupling becomes ineffective. To improve decoupling, we remove the shared temporal kernel in these layers.

Drop layers. Fig. 2 (a) investigates the impact of varying the percentage of dropped layers. The best performance is achieved at 65%, as indicated by the highest scores. This result highlights the importance of a balanced dropout rate in optimizing the model's motion transfer capabilities.

Dense point tracking loss. Fig. 2b, we ablate the dense point tracking loss weight for DeT with HunyuanVideo. We find that λ_{TL} = 1e-1 achieves the best results.

Temporal kernel size. Fig. 2 (c) ablates the kernel size, with results demonstrating that a kernel size of 5 offers optimal performance. This suggests that the receptive field provided by $k = 5$ strikes the best balance between capturing local temporal dependencies and maintaining computational efficiency.

Mid dimension Fig. 2 (d) explores the mid dimension parameter, showing that a dimension of 128 is most effective.

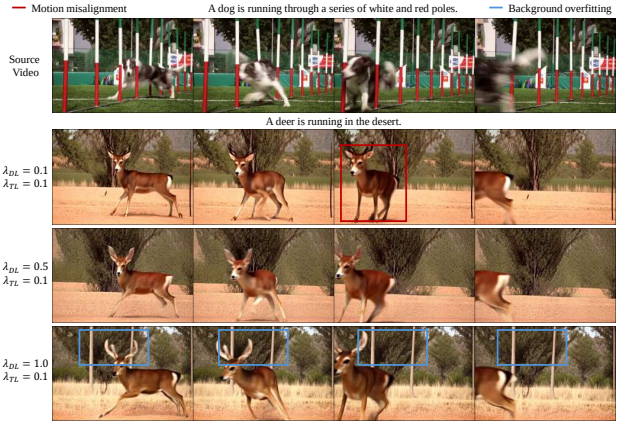

Figure 9. Ablation of the denoising loss weight λ_{DL}.

Percent	EF	TC	MF
75%	31.2	89.1	83.4
65%	**31.9**	**91.9**	**85.9**
55%	30.7	89.0	85.5

(a) Ablation study on the percentage of dropped layers.

λ_{TL}	EF	TC	MF
1e-1	**31.9**	**91.9**	**85.9**
1e-2	31.7	90.1	84.4
w/o TL	31.6	89.4	83.1

(b) Ablation study on the weight of dense point tracking loss λ_{TL}.

k	EF	TC	MF
3	31.4	90.6	85.1
5	**31.9**	**91.9**	**85.9**
7	30.7	89.6	85.7

(c) Ablation study on the kernel size k of shared temporal kernel.

m	EF	TC	MF
64	30.2	90.1	83.2
128	**31.9**	**91.9**	**85.9**
256	30.9	89.9	84.6

(d) Ablation study on the mid dim m of shared temporal kernel.

Table 2. Ablation studies on the hyperparameters in DeT with HunyuanVideo.

Percent	EF	TC	MF
75%	31.1	88.3	82.3
65%	**31.4**	**91.6**	**85.8**
55%	30.2	89.0	84.2

(a) Ablation study on the percentage of dropped layers.

λ_{TL}	EF	TC	MF
1e-1	**31.4**	**91.6**	**85.8**
1e-2	30.5	90.5	84.9
w/o TL	31.4	88.5	82.9

(b) Ablation study on the weight of dense point tracking loss λ_{TL}.

k	EF	TC	MF
3	31.1	90.1	85.3
5	**31.4**	**91.6**	**85.8**
7	30.4	88.6	85.4

(c) Ablation study on the kernel size k of shared temporal kernel.

m	EF	TC	MF
64	31.2	90.5	83.7
128	**31.4**	91.2	84.5
256	**31.4**	**91.6**	**85.8**

(d) Ablation study on the mid dim m of shared temporal kernel.

Table 3. Ablation studies on the hyperparameters in DeT with Step-Video-T2V.

5.4. DeT with Step-Video-T2V

Table 2 presents comprehensive ablation studies conducted on various hyperparameters within the DeT with Step-Video-T2V.

Table 4. Ablation on λ_{DL}.

λ_{DL}	EF	TC	MF
0.1	**32.1**	90.3	85.2
0.5	31.7	**90.4**	85.4
1.0	31.6	**90.4**	**85.6**

Table 5. User study results.

Method	EF	TC	MF
MD	16.7%	20%	5%
MI	10%	23%	40%
DeT	**73.3%**	**56%**	**55%**

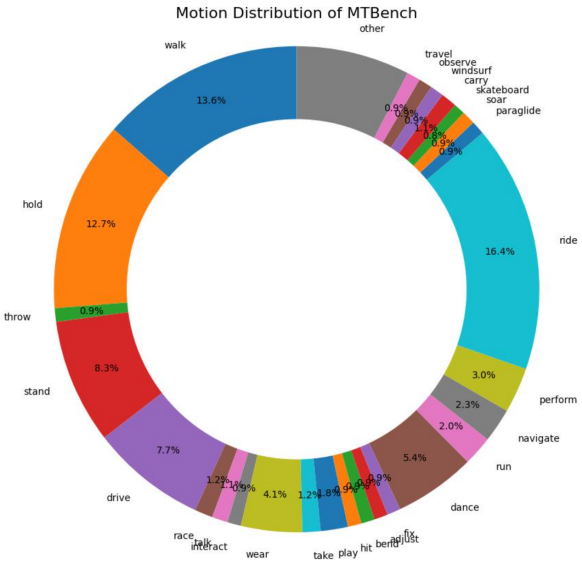

Figure 10. **Motion Distribution of MTBench.**

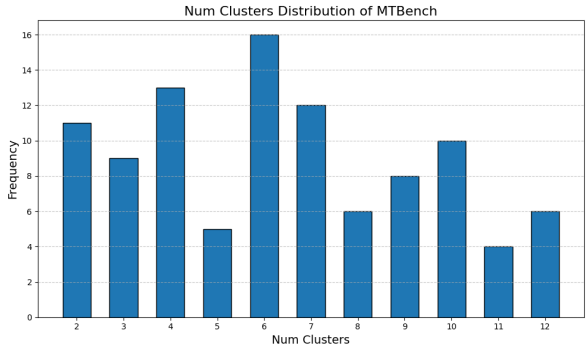

Figure 11. **Num Clusters (Difficulty) Distribution of MTBench.**

Figure 12. **Visualization of Clusters.**

Drop layers. Table 3a presents the ablation study on the percentage of dropped layers. The best performance is achieved at a 65% drop rate.

Dense point tracking loss. In Table 3b, the impact of the weight λ_{TL} for the dense point tracking loss is evaluated. The results show that setting λ_{TL} to 1e-1 yields the highest performance across all metrics.

Temporal kernel size. Table 3c investigates the effect of different kernel sizes for the shared temporal convolution. The experiment reveals that a kernel size of 5 produces the best results, indicating an optimal balance.

Mid dimension Table 3d explores the influence of the mid dimension m of the shared temporal kernel. The performance consistently improves with an increase in m, reaching optimal values when m is set to 256.

6. MTBench Details

MTBench consists of 48 diverse motion categories specifically curated for evaluating motion transfer tasks. We visualize the motion distribution and the number of clusters in MTBench in Fig. 10 and Fig. 11, respectively. Each category is annotated with its occurrence frequency. By incorporating both high-frequency and rare motions, MTBench provides a rigorous assessment of a motion transfer method's generalization and robustness across a wide range of motion complexities.

Additionally, we visualize the trajectory clusters in Fig. 12. The center of each cluster corresponds to a fun-

damental movement element of the foreground, such as a bear's limb. This suggests that aligning the cluster centers of trajectories between the generated and source videos could serve as a potential decoupling method.

7. Additional Related Works

Controllable video generaion. To better meet user demands, previous works [2, 3, 6, 7] have incorporated additional control signals into the video generation process, including control over the first frame [1], motion trajectories [12], object regions [9], and object identity [4, 8].

Specifically, trajectory-controlled methods guide subject movement through per-frame coordinates but lack fine-grained motion control. Region-controlled methods use bounding boxes to constrain subject positions in each frame, yet they still struggle to generate complex motions. Additionally, recent works [13] employ video masks to regulate motion. While masks effectively guide motion generation, they also restrict subject appearance, reducing text controllability.

In this work, we propose a tuning-based motion transfer method for DiT models, where the generated videos follow the motion of the source video while maintaining strong text controllability.

8. Limitations and Future Work

The success of DeT relies on two key assumptions.

First, the foreground and background DiT features must be separable in high-dimensional space. If this condition holds, smoothing along the temporal dimension can help the model better distinguish between foreground and background. However, if these features are not separable in high-dimensional space or remain indistinguishable within a given temporal window defined by the kernel size, then temporal smoothing alone will be ineffective. In such cases, the model may still memorize background appearance, leading to overfitting.

Second, our assumption regarding the dense point tracking loss is that the key regions of foreground motion are already present in the first frame. This enables CoTracker to track foreground motion throughout the sequence. However, if critical motion-related parts are absent in the first frame—for example, a hidden hand—then the effectiveness of the dense point tracking loss may be reduced.

In the future, we will continue to enhance the model's ability to decouple and learn motion. While the shared temporal kernel handles both tasks simultaneously, designing dedicated modules for decoupling and motion learning separately may further improve overall performance.

References

[1] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion Englis, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arxiv:2311.15127*, 2023. 10

[2] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 10

[3] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 10

[4] Qingyu Shi, Lu Qi, Jianzong Wu, Jinbin Bai, Jingbo Wang, Yunhai Tong, Xiangtai Li, and Ming-Husan Yang. Relationbooth: Towards relation-aware customized object generation. *arXiv preprint arXiv:2410.23280*, 2024. 10

[5] Luozhou Wang, Ziyang Mai, Guibao Shen, Yixun Liang, Xin Tao, Pengfei Wan, Di Zhang, Yijun Li, and Yingcong Chen. Motion inversion for video customization. *arXiv preprint arXiv:2403.20193*, 2024. 1

[6] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. In *NeurIPS*, 2023. 10

[7] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *SIGGRAPH*, 2024. 10

[8] Yujie Wei, Shiwei Zhang, Zhiwu Qing, Hangjie Yuan, Zhiheng Liu, Yu Liu, Yingya Zhang, Jingren Zhou, and Hongming Shan. Dreamvideo: Composing your dream videos with customized subject and motion. In *CVPR*, 2024. 10

[9] Jianzong Wu, Xiangtai Li, Yanhong Zeng, Jiangning Zhang, Qianyu Zhou, Yining Li, Yunhai Tong, and Kai Chen. Motionbooth: Motion-aware customized text-to-video generation. *NeurIPS*, 2024. 10

[10] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025. 1

[11] Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *CVPR*, 2024. 1

[12] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023. 10

[13] Qiang Zhou, Shaofeng Zhang, Nianzu Yang, Ye Qian, and Hao Li. Motion control for enhanced complex action video generation. *arXiv preprint arXiv:2411.08328*, 2024. 10