

Supplementary Material for FlexTraj

A. Experiment details

Implementation details. We start by constructing trajectory representations. For real-world videos, we first annotate points by SAM [10] for video segmentation and SpatialTracker [15] for tracking 4,900 uniformly distributed 3D points. We then project these points onto the 2D plane as videos, where the point size is dynamically adjusted to accommodate different levels of granularity: $h = \lfloor 2s \rfloor$ and $w = \lfloor 3s \rfloor$, where $s = \min(\sqrt{H/x/1.7}, 4)$ and x denotes grid size. For CG synthetic videos, we render the condition video directly in Blender, where each mesh is treated as an instance and each vertex serves as a tracking point.

After constructing trajectory representations, we next describe our annealing training schedule, which consists of four stages: a complete stage of 1,200 steps, a dense stage of 2,400 steps, a sparse stage of 14,000 steps, and finally an unaligned stage of 4,000 steps. We set p_c to 0.5, while p_s and p_t take values in the range $[0, 1]$. The learning rate is fixed at 1×10^{-4} for the aligned stages (first three) and reduced to 1×10^{-5} for the unaligned stage.

Our model is fine-tuned on the recent video diffusion model CogVideoX-5B I2V [16], which is based on the MM-DiT architecture [3]. Fine-tuning is performed with LoRA (rank 128, batch size 1) applied to the self-attention query, key, and value projections. Training requires about one week on 8 NVIDIA A800 GPUs, while inference requires approximately 5 minutes per video (49 frames) with KV-cache enabled and about 9.5 minutes without it.

Evaluation dataset. For evaluation, we use DAVIS [8] as the standard benchmark and configure it for four tasks: dense, spatially sparse, temporally sparse, and unaligned. Spatial sparsity is simulated by randomly sampling 10 points, while temporal sparsity is obtained by uniformly sampling 2–4 frames from the full sequence. The unaligned setting is generated by randomly jittering the condition videos: we first resize the video to a larger resolution and then crop it back to the target size to obtain tracking points. In addition, we curate FlexBench, which includes 10 videos for each task, half collected from online sources and half synthesized with Blender, to demonstrate applicability for both general and professional users. All videos are trimmed to 49 frames and cropped to 720×480 .

Metrics. We employ several standard metrics. For overall video quality, we report Fréchet Video Distance (FVD [11]) and Frame Consistency [2], which measures CLIP similarity [9] between consecutive frames. For motion controllability, we use Trajectory Error (TrajError) [14], defined as the average Euclidean distance between trajectories extracted from the generated video and their matched trajectories extracted from the source video. For

the unaligned setting, we adopt Trajectory Similarity (TrajSIM) [7], computed as the mean cosine similarity between the displacement directions of each extracted trajectory in generated video and its closest counterpart in source video.

B. User Study

We conduct a user study to further evaluate the perceptual quality of our method compared with representative baselines. The study covers four different tasks, including dense, spatially sparse, temporally sparse, and unaligned. For each task, we randomly select five representative examples, resulting in a total of 20 test cases.

We collected 24 valid responses in total, seven from participants with a computer graphics background and 17 from other fields. The age distribution of participants is as follows: 12 participants aged 20–30, 7 aged 40–60, 4 aged 30–40, and 1 below 20.

For each case, participants were shown the corresponding animation results produced by our method and the competing approaches. All results were presented in a randomized order to avoid bias. Participants were asked to evaluate the results according to:

- **Alignment:** how well the generated video follows the intended motion of the reference video;
- **Consistency:** the temporal stability of the video, with fewer artifacts preferred;
- **Overall quality:** the overall visual quality, considering artifacts and realism.

Participants were required to select the best result for each criterion. The preference scores were normalized to obtain the final ratios. As shown in Fig. 1, the user study demonstrates that our method consistently achieves the highest preference rates across tasks. On average, our method achieves a preference rate of 0.63 for alignment, 0.61 for consistency, and 0.63 for overall quality, validating its effectiveness from a human perceptual perspective.



Figure 1. Preference Rate for User Study. User preference comparison across four tasks (dense, spatially sparse, temporally sparse, and unaligned), evaluated by Alignment, Consistency, and Quality on 20 videos (5 per task).

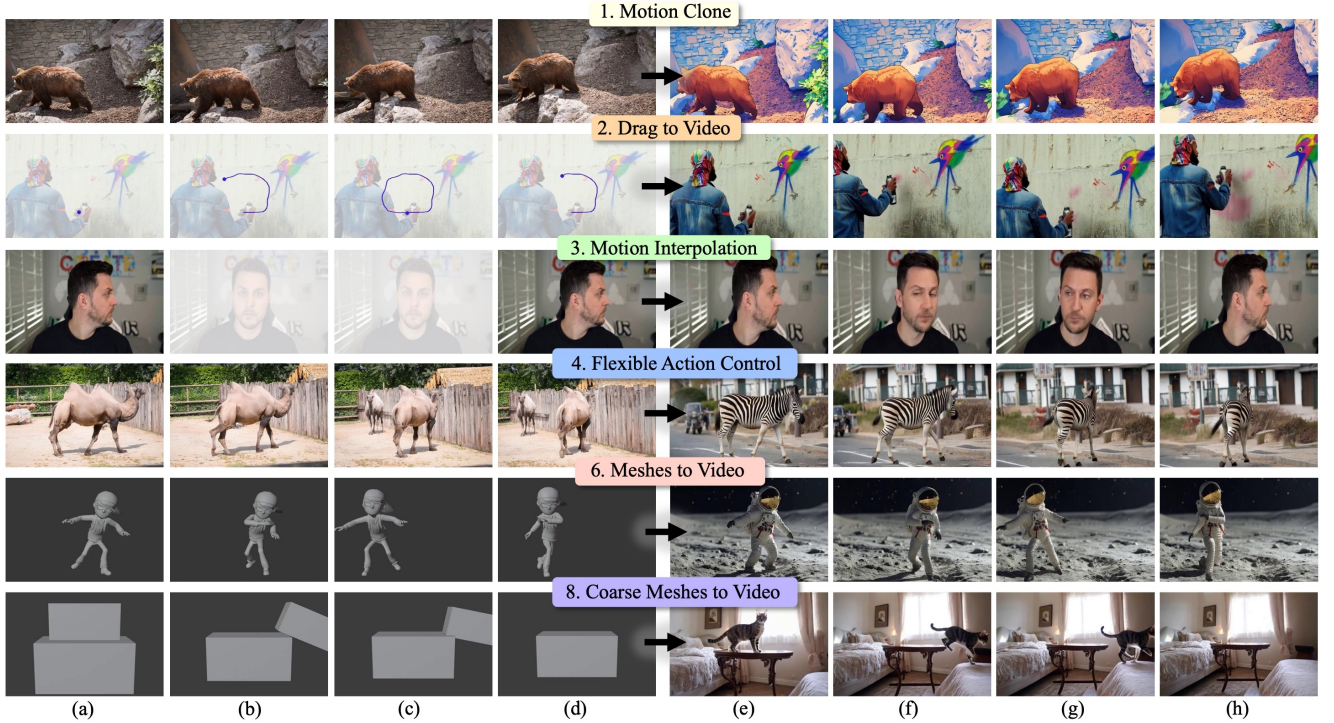


Figure 2. Generalization to WAN, which exhibits stronger capability in long-sequence. The showcased examples has 81 frames.

C. Robustness to Unaligned Cases

Our method is robust to various real-world misalignments, including camera motion, perspective shifts, tracking drift, and object emergence or disappearance, as illustrated in Fig. 3. To achieve this, our training pipeline simulates unaligned data through trajectory shifts and CG synthesis, covering a broad range of perspective changes and morphological variations.

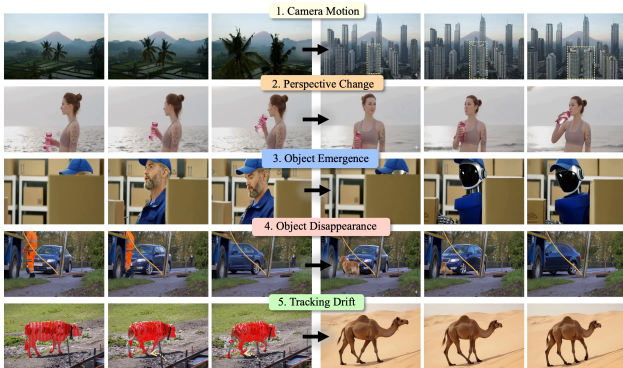


Figure 3. Examples on Unaligned Cases.

D. Analysis of ID-coded Representation

To further evaluate the effectiveness of our ID-coded representation (TrajID/SegID), we conduct an additional comparison against alternative rendering-based baselines using

the Wan model. Specifically, we consider a baseline that conditions the model directly on rendered projections (e.g., point or depth-like maps) of the tracked 3D points.

Rendering-based methods fail for two reasons. First, they rely on strict color and depth projections: when points are not associated with accurate color or depth cues, failures readily occur, such as in coarse mesh guidance (Fig. 4, row 1). Second, these approaches are ambiguous, as different points may be rendered with the same color and depth, making them indistinguishable and losing correspondence, especially in sparse settings (Fig. 4, row 2). Quantitatively, rendering-based baselines underperform our method in both control accuracy (TrajErr/TrajSim) and quality (FVD) in Tab. 1, rows 2–3, further validating our design.

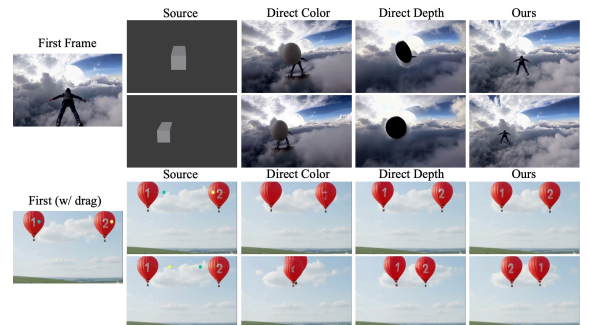


Figure 4. Comparison to Direct Rendering Baselines.

Table 1. Analysis of ID-coded Representation with Wan. Values are reported as Aligned | Unaligned. Second best in blue.

Method	FVD↓		Consistency↑		TrajErr↓ TrajSIM↑	
DR(color)	761.5	697.8	0.982	0.975	0.031	0.905
DR(depth)	738.5	671.3	0.981	0.976	0.042	0.896
Ours(wan)	708.2	670.0	0.980	0.974	0.030	0.905

Table 2. Quantitative result of FlexTraj on CogvideoX/Wan. We present CogvideoX results before Wan for each dataset.

Task	Dataset	FVD↓	Consistency↑	TrajErr/TrajSIM
Dense	DAVIS	532.4	0.979	0.017
		578.2	0.976	0.026
	FlexBench	1397.8	0.982	0.014
		1237.5	0.982	0.021
Spatially Sparse	DAVIS	710.4	0.980	0.025
		795.2	0.980	0.032
	FlexBench	851.6	0.991	0.017
		1143.2	0.991	0.024
Temporally Sparse	DAVIS	837.0	0.983	0.031
		751.2	0.983	0.032
	FlexBench	1144.8	0.994	0.017
		1100.8	0.993	0.016
Unaligned	DAVIS	622.3	0.976	0.908
		670.0	0.974	0.905
	FlexBench	2654.2	0.993	0.757
		2872.9	0.993	0.726

E. Generalization to WAN

Our method generalizes to other video generators such as Wan2.2. As shown in Table 2, our method achieves comparable performance on Wan2.2 [12] and CogVideoX [16], both outperforming or remaining competitive with all baselines. Wan [12] further demonstrates stronger potential for long-sequence generation, highlighting the scalability of our framework. We provide 81-frame examples in Fig. 2.

F. More Comparisons

We provide additional qualitative results in Fig. 6, which further demonstrate our superiority.

G. More Results

We provide additional results in Fig. 7, covering all applications introduced in teaser, to more clearly demonstrate the effectiveness of our method.

H. Limitation

Our method faces two main limitations. First, it relies on tracking quality: when tracking fails, regions missed by tracking default to free generation, as in Fig. 5 where the woman’s glove is misaligned. Second, it inherits constraints from the underlying video generator, including difficulty with large rotations and limited long-term memory. For instance, after a 360° camera orbit, scene quality degrades and

the original scene cannot be faithfully recovered. Future work includes exploring the integration of explicit memory mechanisms to enhance long-term scene consistency.

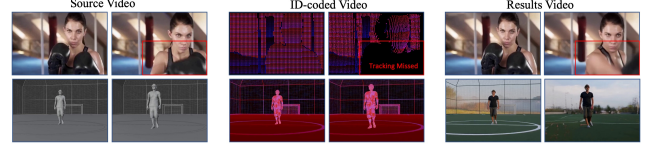
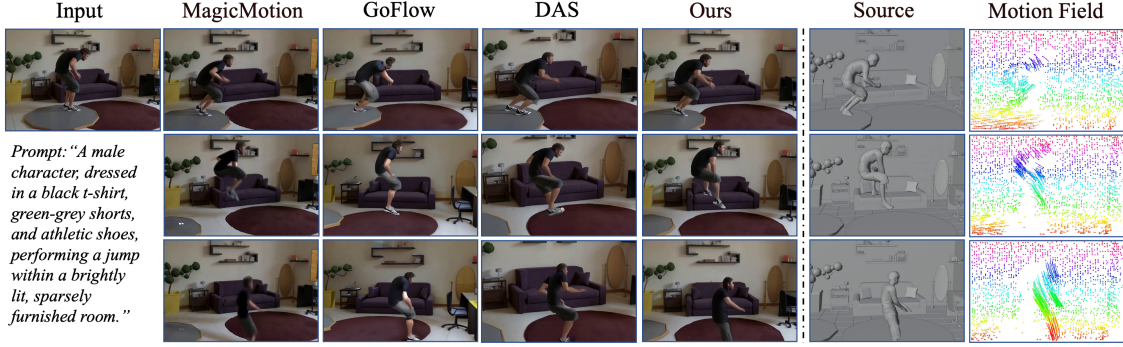


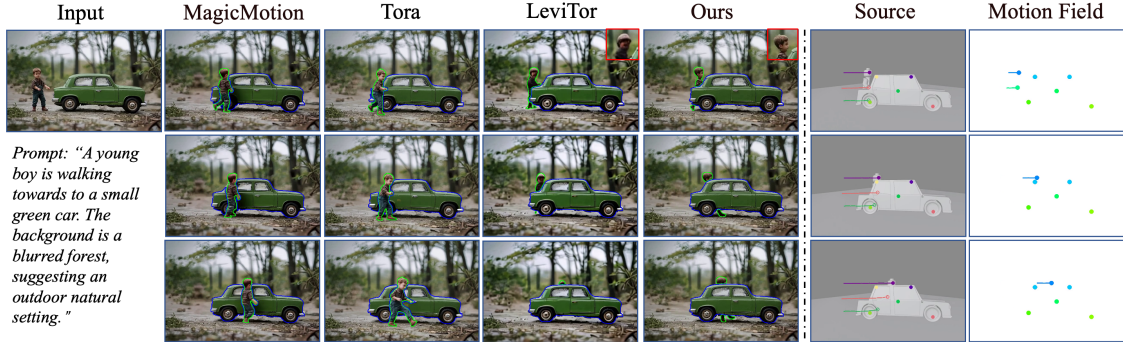
Figure 5. Limitations. Motion alignment is limited by tracking quality (top row: glove), and generation is constrained by the base video model (bottom row: fails on a 360° camera orbit.)

References

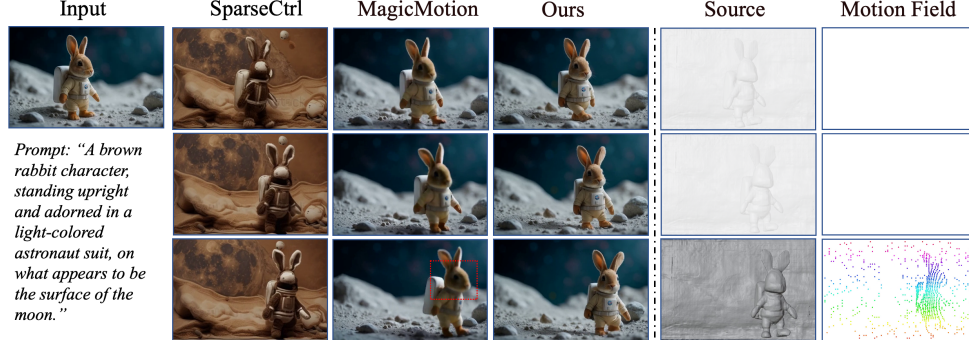
- [1] Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13–23, 2025. 4
- [2] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 1
- [3] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 1
- [4] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, et al. Diffusion as shader: 3d-aware video diffusion for versatile video generation control. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–12, 2025. 4
- [5] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *European Conference on Computer Vision*, pages 330–348. Springer, 2024. 4
- [6] Quanhao Li, Zhen Xing, Rui Wang, Hui Zhang, Qi Dai, and Zuxuan Wu. Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. *arXiv preprint arXiv:2503.16421*, 2025. 4
- [7] Alexander Pondaven, Aliaksandr Siarohin, Sergey Tulyakov, Philip Torr, and Fabio Pizzati. Video motion transfer with diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22911–22921, 2025. 1
- [8] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 1
- [9] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,



(a) Dense control. MagicMotion [6] and Go-with-the-Flow [1] struggle with fine-grained details; DAS [4] fails to handle newly emerging points, whereas our method closely follows the source motion.



(b) Spatially sparse control. The subject outlined in green is occluded by the subject outlined in blue. MagicMotion [6] and ToRA [17] fail under occlusion, LeviTor [13] introduces artifacts, while ours accurately captures occlusion.



(c) Temporally sparse control. SparseCtrl [5] yields unsatisfactory results, while MagicMotion [6] shows weak alignment and blurriness. Ours aligns with the anchor-frame motion and generates coherent in-between frames.



(d) Unaligned control. DAS [4] introduces artifacts (red artifacts around the subject) from strict alignment, while Go-with-the-Flow [1] produces implausible results (e.g. distorted tails). Our method flexibly follows input motion.

Figure 6. Qualitative comparison across four evaluate tasks: dense, spatially sparse, temporally sparse, and unaligned.

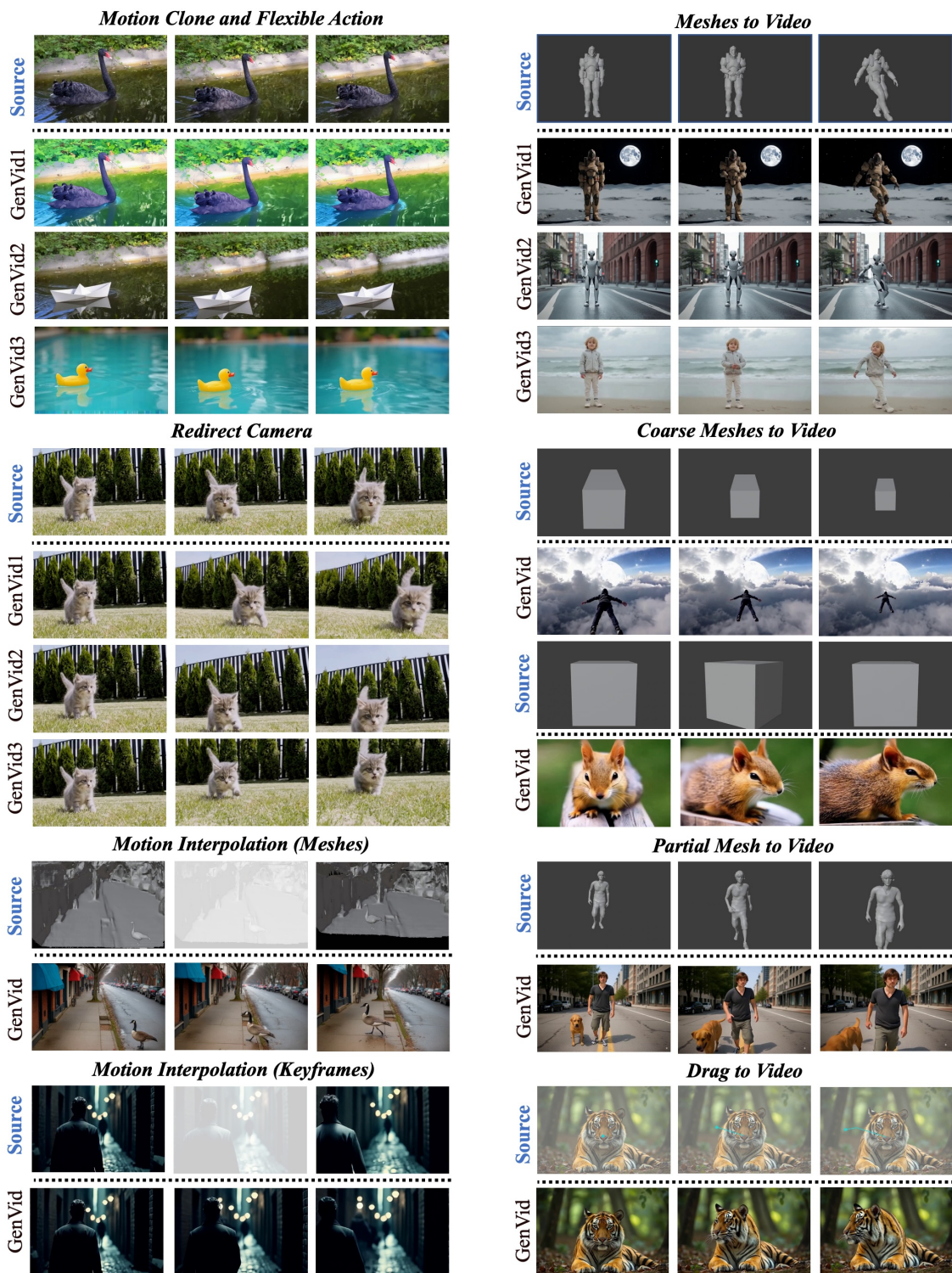


Figure 7. More results. We provide additional results on all the applications here.

Amanda Asbell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages

8748–8763. PMLR, 2021. 1

[10] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman

- Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. [1](#)
- [11] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. [1](#)
- [12] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. [3](#)
- [13] Hanlin Wang, Hao Ouyang, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Qifeng Chen, Yujun Shen, and Limin Wang. Levitor: 3d trajectory oriented image-to-video synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12490–12500, 2025. [4](#)
- [14] Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. Draganything: Motion control for anything using entity representation. In *European Conference on Computer Vision*, pages 331–348. Springer, 2024. [1](#)
- [15] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024. [1](#)
- [16] 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. *arXiv preprint arXiv:2408.06072*, 2024. [1](#), [3](#)
- [17] Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2063–2073, 2025. [4](#)