Anchoring and Rescaling Attention for Semantically Coherent Inbetweening (Supplementary Material)

Contents

Page

1. Additional Resources **2**

2. Evaluation Details **2**
2.1. Experimental Details . . . 2
2.2. Video Question Answering Evaluation Details . . . 2
2.3. User Study Details . . . 2

3. Additional Experimental Results **2**
3.1. Quantitative Results . . . 2
3.2. Qualitative Results . . . 3
3.3. Hyperparameter Experiment . . . 3

4. Additional Analysis **4**
4.1. KAB . . . 4
4.2. ReTRo . . . 4
4.3. Generative Inbetweening Challenge Analysis . . . 4

5. TGI-Bench Details **5**
5.1. Dataset Curation Details . . . 5
5.2. GPT Prompt . . . 6

6. Limitation **7**

7. Ethical Considerations **7**

1. Additional Resources

The implemented code for our method is presented in the `code` folder in the supplementary material. We also present *all* our result for the 81-frame along with the result videos of the baseline Wan [2] in the `videos` folder in the supplementary material. In addition, a sample dataset of our TGI-Bench is included in the `TGI-Bench` folder.

2. Evaluation Details

2.1. Experimental Details

All experiments were conducted on an NVIDIA RTX A6000 GPU (48GB VRAM) using mixed precision (bfloat16). We utilized the Wan2.1-FLF2V-14B-720P checkpoint [2], a 14B-parameter diffusion transformer model, UMT5-XXL text encoder, VAE decoder, and XLM-RoBERTa-Large vision encoder which can all be accessed through Wan2.1[1]. Inference was performed using the DiffSynth-Studio[2] framework, which provides efficient pipeline management and automatic VRAM optimization. In addition, videos were generated at 480×864 resolution with 15 FPS using tiled processing. For Stable Video Diffusion–based models, we used the following checkpoints in our experiments: stable-video-diffusion-img2vid-xt for GI [10], ViBiD [13], TRF [3] and stable-video-diffusion-img2vid-xt-1-1 for FCVG [18].

2.2. Video Question Answering Evaluation Details

To obtain a stable VQA-based alignment score between a generated video and its textual prompt, we evaluate each video using six vision–language models with diverse architectures and visual encoders: qwen2.5-vl-7b [12], llava-onevision-qwen2-7b-sillavaonevision, internlmxcomposer25-7b [15], tarsier-recap-7b [14], llava-video-7b [17], and gpt-4.1 [1]. For each model, we sample video frames using either an FPS-based strategy (Qwen models) or a fixed frame-count strategy (LLaVA, InternLM-XComposer, Tarsier), encode the frames through the model's vision encoder, and compute a binary VQA response to the question `Does this video show {caption}?`. Each model produces a probability score for the `Yes` response, normalized to $[0, 1]$ from the logits of the `Yes` and `No` tokens (or log-probabilities in the case of gpt-4.1). Because individual models exhibit significant variance due to differences in frame sampling, vision encoders, and temporal reasoning ability, we average the scores across all six models to obtain a more reliable and model-agnostic VQA metric.

2.3. User Study Details

Figure S1 shows the interface used in our user study, which was conducted with more than 20 participants. For each of the 12 questions (about 10% of TGI-Bench), participants were given a text prompt and 6 video clips generated by each baseline models, whose positions were randomly shuffled to ensure fairness. They then rated every clip on semantic fidelity, pace stability, and frame consistency using a five-point Likert scale. To assess inter-rater reliability, we computed ICC(2,k) [8] across the 21 participants, obtaining 0.953 for semantic fidelity, 0.967 for pace stability, and 0.964 for frame consistency.

3. Additional Experimental Results

3.1. Quantitative Results

In Tab. S1, we present quantitative results for the 25- and 33-frame sequences. All settings, except for the number of frames, are identical to those used for the 65- and 81-frame sequences in the main paper.

Table S1. **Additional Video Generation Evaluation Results.** Quantitative comparison of the baselines and our method on 25, 31 frames. We evaluate video generation quality and fidelity. The best results are in **bold**, and the second best are underlined.

Method	25-frame						33-frame					
	PSNR↑	SSIM↑	LPIPS↓	FID↓	FVD↓	VBench↑	PSNR↑	SSIM↑	LPIPS↓	FID↓	FVD↓	VBench↑
TRF [3]	16.734	0.5546	0.4612	104.393	0.2749	9.473	16.603	0.5584	0.4777	118.459	0.2893	9.147
ViBiDSampler [13]	17.029	0.5686	0.4257	93.172	0.2776	9.587	16.607	0.5574	0.4561	107.697	0.3121	9.245
GI [10]	17.418	0.5801	0.3972	91.884	0.2571	9.935	16.499	0.5587	0.4470	127.957	0.2955	9.339
FCVG [18]	18.264	0.5631	0.3859	80.276	0.2016	9.865	17.682	0.5523	0.4083	88.814	0.2508	9.781
Wan [2]	19.076	0.6180	0.3430	68.890	0.1821	**10.103**	18.174	0.5953	0.3771	74.383	0.2409	9.915
Ours	**19.557**	**0.6322**	**0.3418**	**67.888**	**0.1682**	9.991	**18.757**	**0.6127**	**0.3669**	**70.399**	**0.2086**	**9.918**

[1] https://github.com/Wan-Video/Wan2.1

[2] https://github.com/modelscope/DiffSynth-Studio

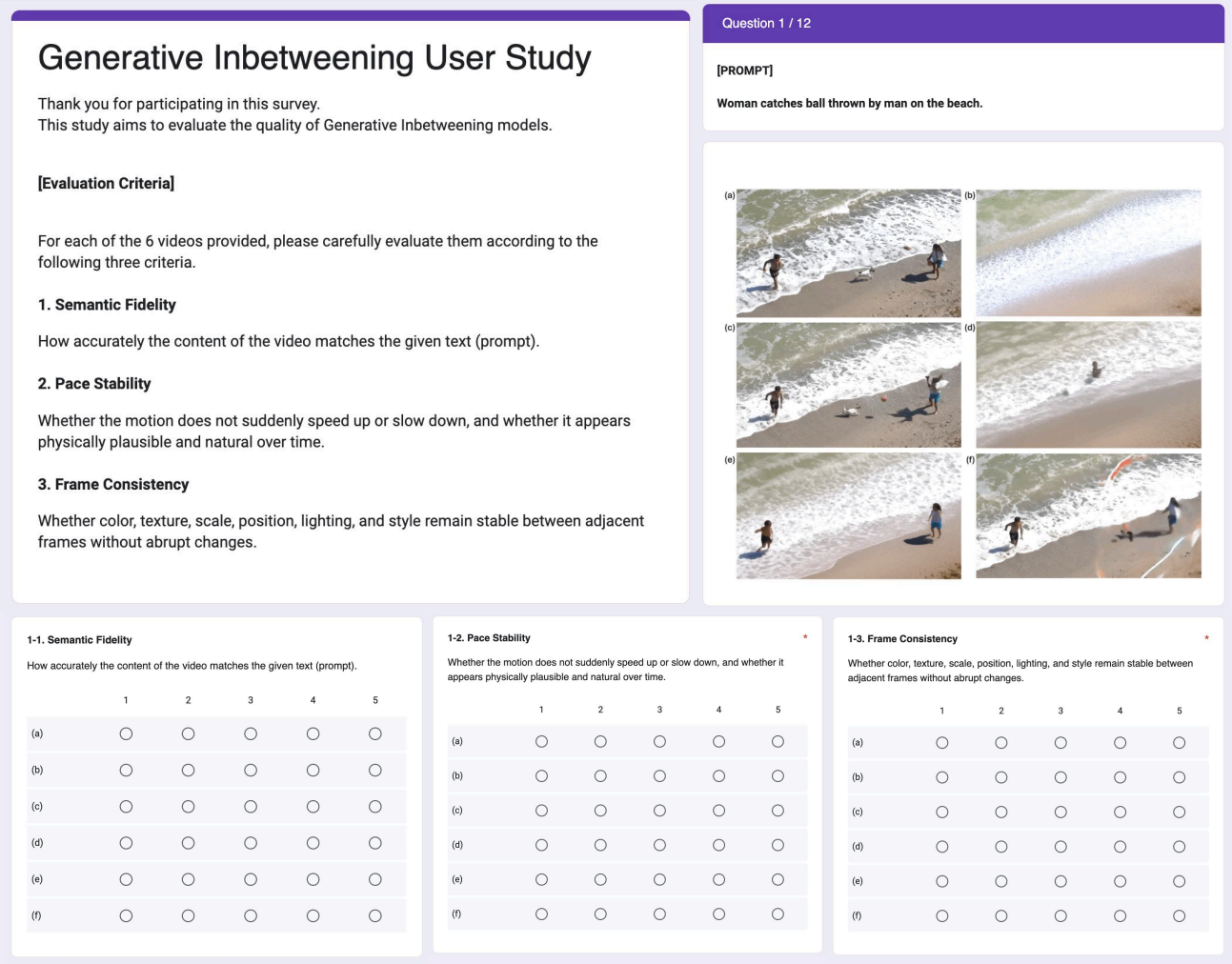

Figure S1. **User study interface used to evaluate our generative inbetweening results.** For each text prompt (top right), six candidate videos (a–f) are displayed. Participants first read the evaluation criteria (left) and then rate each video on a 5-point Likert scale for three dimensions: Semantic Fidelity, Pace Stability, and Frame Consistency.

3.2. Qualitative Results

We provide additional qualitative results for all our baseline models in Figs. S4–S12.

3.3. Hyperparameter Experiment

We conduct an ablation study on hyperparameters on the two main components of our method, Keyframe-anchored Attention Bias (KAB) and Rescaled Temporal RoPE (ReTRo). The results are summarized in Tab. S2 and Tab. S3. Overall, these ablations indicate that our chosen hyperparameters provide a good balance between fidelity and perceptual quality, and that our method is reasonably robust to moderate changes in these values.

For KAB, we experiment over the temporal range $[\beta_t^{\min}, \beta_t^{\max}]$. Narrow or overly wide ranges as well as too high or low values generally degrade performance across distortion and perceptual metrics. In contrast, our default setting ($0.3 \leq \beta_t \leq 0.7$) achieves the best overall scores, yielding clear gains in PSNR, SSIM [11], and FVD [9] while also improving perceptual quality (VBench [5]).

To analyze the effect of the ReTRo, we scale the parameters s_{mid} and s_{edge}, which control the relative emphasis on mid-sequence versus boundary frames in the temporal RoPE rescaling. Our default configuration ($s_{mid} = 0.94,\ s_{edge} = 1.06$) achieves the best performance on most metrics, including PSNR, SSIM, LPIPS [16], FID [4], and FVD, while maintaining competitive VBench scores. The alternative setting ($s_{mid} = 0.88,\ s_{edge} = 1.06$) provides the second-best overall performance and slightly higher VBench.

Table S2. **Hyperparameter Experiment Results on KAB.** The best results are in **bold**, and the second best are underlined. **Ours** denotes the hyperparameters used in our method.

Hyperparameters	PSNR↑	SSIM↑	LPIPS↓	FID↓	FVD↓	VBench↑
$0.1 \leq \beta_t \leq 0.5$	17.0651	0.5859	0.3972	84.898	0.2929	**10.237**
$0.5 \leq \beta_t \leq 0.9$	17.100	0.5856	0.3973	85.051	0.2839	10.230
$0.5 \leq \beta_t \leq 0.5$	17.107	0.5859	0.3968	83.915	0.2865	10.203
$0.1 \leq \beta_t \leq 0.9$	17.072	0.5853	0.3977	84.636	0.2887	10.208
Ours $(0.3 \leq \beta_t \leq 0.7)$	**18.169**	**0.6269**	**0.3818**	**77.587**	**0.2458**	10.022

Table S3. **Hyperparameter Ablation on ReTRo.** The best results are in **bold**, and the second best are underlined. **Ours** denotes the hyperparameters used in our method.

	Hyperparameters		Metrics					
	s_{mid}	s_{edge}	PSNR↑	SSIM↑	LPIPS↓	FID↓	FVD↓	VBench↑
	0.94	1.12	16.964	0.5819	0.4054	90.584	0.3005	10.157
	0.88	1.06	17.481	0.5941	0.3842	78.739	0.2660	**10.339**
Ours	0.94	1.06	**18.169**	**0.6269**	**0.3818**	**77.587**	**0.2458**	10.022

4. Additional Analysis

4.1. KAB

KAB is a method that uses the cross-attention of the keyframes to guide intermediate frames under three conditions: the two keyframes and the text prompt. Through rigorous experiments in the main paper and in the supplementary material, we have shown that this additional guidance is effective in maintaining both semantic fidelity and pace stability. However, when the guidance is either too weak or overly strong, it instead degrades these properties, along with the overall video generation quality.

As shown in Tab. S2, our default mid-range setting $(0.3 \leq \beta_t \leq 0.7)$ clearly outperforms all other ranges on PSNR, SSIM, LPIPS, FID, and FVD. Interestingly, ranges biased toward either lower $(0.1 \leq \beta_t \leq 0.5)$ or higher $(0.5 \leq \beta_t \leq 0.9)$ scales achieve slightly higher VBench scores, but this comes at the cost of noticeably worse distortion and distributional metrics. The very narrow range $(0.5 \leq \beta_t \leq 0.5)$ yields the second-best FID and LPIPS among the ablated settings, yet still fails to close the gap to our default configuration. Taken together, these results suggest that while concentrating guidance at specific diffusion phases can bring marginal gains in certain perceptual aspects, distributing KAB over a moderate mid-range window is crucial for obtaining consistent improvements across both fidelity and perceptual metrics. Thus, our chosen setting strikes a good balance when applied with a moderate guidance range, which we have empirically demonstrated through our ablation studies.

4.2. ReTRo

ReTRo adaptively modulates RoPE scales along the temporal axis, assigning higher scales to tokens near keyframes to sharpen locality and preserve keyframe content, while using lower scales on intermediate frames to broaden attention and promote temporal consistency. In the main paper, we showed that this method is effective in improving both frame consistency and overall video generation quality.

As shown in Tab. S3, we additionally conducted an ablation study on the ReTRo hyperparameters s_{mid} and s_{edge}. From the results, we found that the parameter setting used in the original paper, $(s_{\text{mid}} = 0.94, s_{\text{edge}} = 1.06)$, achieved the best performance among the configurations we tested. For s_{edge}, values around 1.10 or higher started to introduce noticeable visual artifacts, while for s_{mid}, smaller values tended to make the generated videos appear slightly slower in terms of motion. Consequently, we adopt $(s_{\text{mid}} = 0.94, s_{\text{edge}} = 1.06)$ as our default setting, as it offers the best trade-off between visual quality and temporal coherence in our experiments. However, since these are simple hyperparameters, they can be exposed as user-adjustable parameters, allowing users to dynamically adjust the balance between sharpness, motion speed, and temporal consistency to suit their specific applications.

4.3. Generative Inbetweening Challenge Analysis

We additionally present quantitative results for three representative frames (25, 33, and 65) from for further analysis on the generative inbetweening challenges. The examples are shown in Figs. S2. These results further confirm that the four challenge

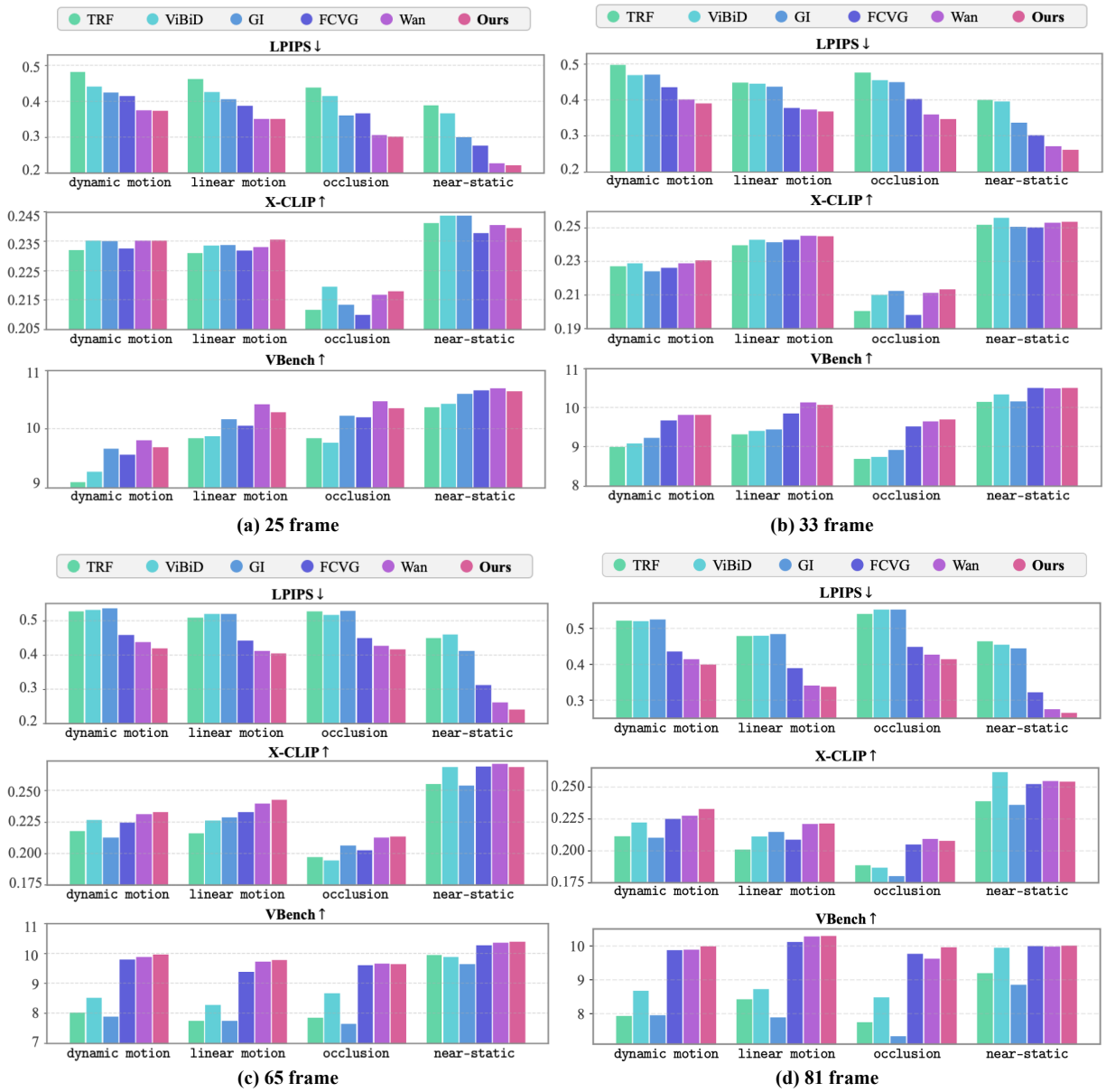

Figure S2. **Complete Results on Generative Inbetweening Challenge Analysis.** Results for all challenges at frames 25, 33, 65, and 81, including VBench, LPIPS, and X-CLIP scores for each challenge.

categories in TGI-Bench are categorized well in difficulty since most models perform reliably on the `near-static` cases, whereas performance degrades sharply on the more demanding `occlusion` and `dynamic motion` challenges. This clear differentiation demonstrates that TGI-Bench is carefully constructed to expose distinct failure modes of GI models, enabling fine-grained diagnosis of model capabilities. Consequently, TGI-Bench provides a reliable and informative metric suite for future research, particularly for identifying which generative inbetweening challenges a model handles well and where it struggles.

5. TGI-Bench Details

5.1. Dataset Curation Details

To construct the TGI-Bench dataset, we prompted GPT-4.1 [1] to generate a text description and a challenge label for each video. The text description often includes information inferred from intermediate frames that are not visible in the provided

first and last frames, thereby serving as constraints when a model attempts to generate the missing frames. Inspired by [6], the challenge label is categorized into one of four types: `dynamic motion`, `linear motion`, `occlusion`, and `near-static`, defined as follows:

- **Dynamic motion**: The primary object exhibits nonlinear or complex movement, such as rotation or abrupt directional changes.
- **Linear motion**: The primary object moves in a linear and consistent direction.
- **Occlusion**: The primary object either appears or disappears in the middle of the video due to occlusion.
- **Near-static**: The primary object remains largely stationary with minimal motion.

We sourced videos from the DAVIS [7] dataset as well as from Pexels and Pixabay[3]. Videos that were too visually complex to describe succinctly, or that lacked a clearly identifiable primary object, were excluded. For example, we removed videos where geometric patterns changing chaotically or where smoke particles moving randomly without a coherent subject. After this filtering step, we collected a total of 220 videos. For each video, we selected only the first F frames and discarded videos whose total frame count was less than F. From these, we took frames at indices $\{0, 10, 20, \ldots, \lfloor (F-1)/10 \rfloor, F-1\}$ and provided them to GPT-4.1 along with the prompt in Sec. 5.2. The resulting text descriptions and challenge labels were manually reviewed and corrected by the authors to ensure accuracy. In particular, GPT's generic label `large motion` was refined into the more specific categories of `dynamic motion` and `linear motion`. This process was repeated for $F \in \{25, 33, 65, 81\}$, yielding four distinct subsets of the dataset.

5.2. GPT Prompt

In this section, we present the detailed prompts provided to GPT-4.1. By default, we feed the model the concatenation of `SYSTEM_PROMPT` and `USER_PROMPT_BASE`. For videos where the model produced outputs that did not follow the intended format, we additionally concatenate `RETRY_PROMPT` to the input.

```
SYSTEM_PROMPT = """
You are a caption generator for a Video Frame Interpolation (VFI) evaluation set.
INPUT: two endpoint images - A (start) and B (end), optional reference images R_i sampled between A
↪  and B, and optional reference text (prompts.txt).
TASKS
1) Briefly describe A and B (visible, objective facts; ≤ 20 words each).
2) Classify the challenge as exactly one of:
   - Large motion
   - Occlusion
   - Near-static
   If ambiguous, tiebreaker: Occlusion > Large motion > Near-static.
3) Generate exactly ONE caption that best describes the plausible situation across A->B.
   - Prefer wording and nouns from the reference prompts when correct.
   - If the reference contains mistakes or conflicts with the images, FIX them in your caption.
CAPTION STYLE (strict)
- English only. <=12 words. One simple clause.
- You may include direction if clearly implied by the endpoints.
- No commas/semicolons. Avoid: and, then, while, as, because, so, therefore, hence.
- No meta words: relative, compared, background, foreground, camera, optical flow, frame,
↪  endpoint(s).
- No hedging or subjective words.
- Do NOT mention A/B or frames.
CONSISTENCY
- Match direction/size/visibility in endpoints.
- Use "emerges/appears/enters" only if absent at A and present at B.
OUTPUT JSON ONLY:
{
  "first_image_desc": "< ≤20 words >",
  "last_image_desc": "< ≤20 words >",
  "challenge": "Large motion | Occlusion | Near-static",
  "caption": "< ≤12 words >"
}
""".strip()


USER_PROMPT_BASE = """
Images follow in order: A (start), zero or more reference images R_i between A and B, then B (end).
```

[3] https://www.pexels.com/, https://www.pixabay.com/

```
Return JSON ONLY following the schema. English only.
""".strip()

RETRY_PROMPT = """
Return VALID JSON ONLY with keys:
first_image_desc, last_image_desc, challenge, caption.
Choose one: Large motion | Occlusion | Near-static.
One caption only; ≤12 words; one clause; obey all style rules.
""".strip()
```

6. Limitation

Figure S3. **Limitation.** Because our training-free plug-in is bounded by the generative capacity of Wan, it can only partially correct the severely distorted motion and geometry seen in the breakdancing example, leaving residual shaky and unnatural motion.

Although our method is a simple, training-free plug-in that can be readily applied to existing DiT-based models, it is inherently bounded by the generative capacity of the underlying baseline, Wan [2]. In challenging cases where the base model already produces severely distorted motion or object geometry over most frames, our approach has limited ability to fully recover a plausible video. For example, as shown in Fig. S3, the breakdancing subject exhibits persistent shaky and unnatural motion across time, and these artifacts are only partially mitigated by our method. We regard this as a natural limitation of training-free refinement methods and as a promising direction for future work on jointly improving both the base generator and the inbetweening modules.

7. Ethical Considerations

TGI-Bench builds on publicly available video dataset Davis [7] and open-source video websites Pexels and Pixabay that permit research use. We do not collect new data of human subjects, nor do we attempt to infer or annotate sensitive attributes (e.g., identity, race, health, or political views). Any videos containing people are used only for generic motion and scene understanding, and are treated as anonymous visual content.

We conducted a small-scale human evaluation with more than 20 participants to compare perceptual quality and consistency, under strict ethical considerations. The study followed a double-blind setup, where participants were unaware of the underlying methods being compared, and experimenters did not have access to any identifying information about individual participants. All participants volunteered to take part in the study and were informed that the evaluation was conducted solely for academic research purposes. No personal information was collected beyond basic platform metadata, and responses were analyzed only in aggregate. No offensive, violent, or explicit prompts were used in any of our experiments.

Generative inbetweening can, in principle, be misused for deceptive or non-consensual content (e.g., manipulated videos). We explicitly prohibit such uses. Our method is presented for research purposes, and any future release of code, models, or benchmarks should be accompanied by clear usage guidelines and restrictions, encouraging applications such as animation, content restoration, and creative tools while discouraging privacy-invasive or harmful deployments.

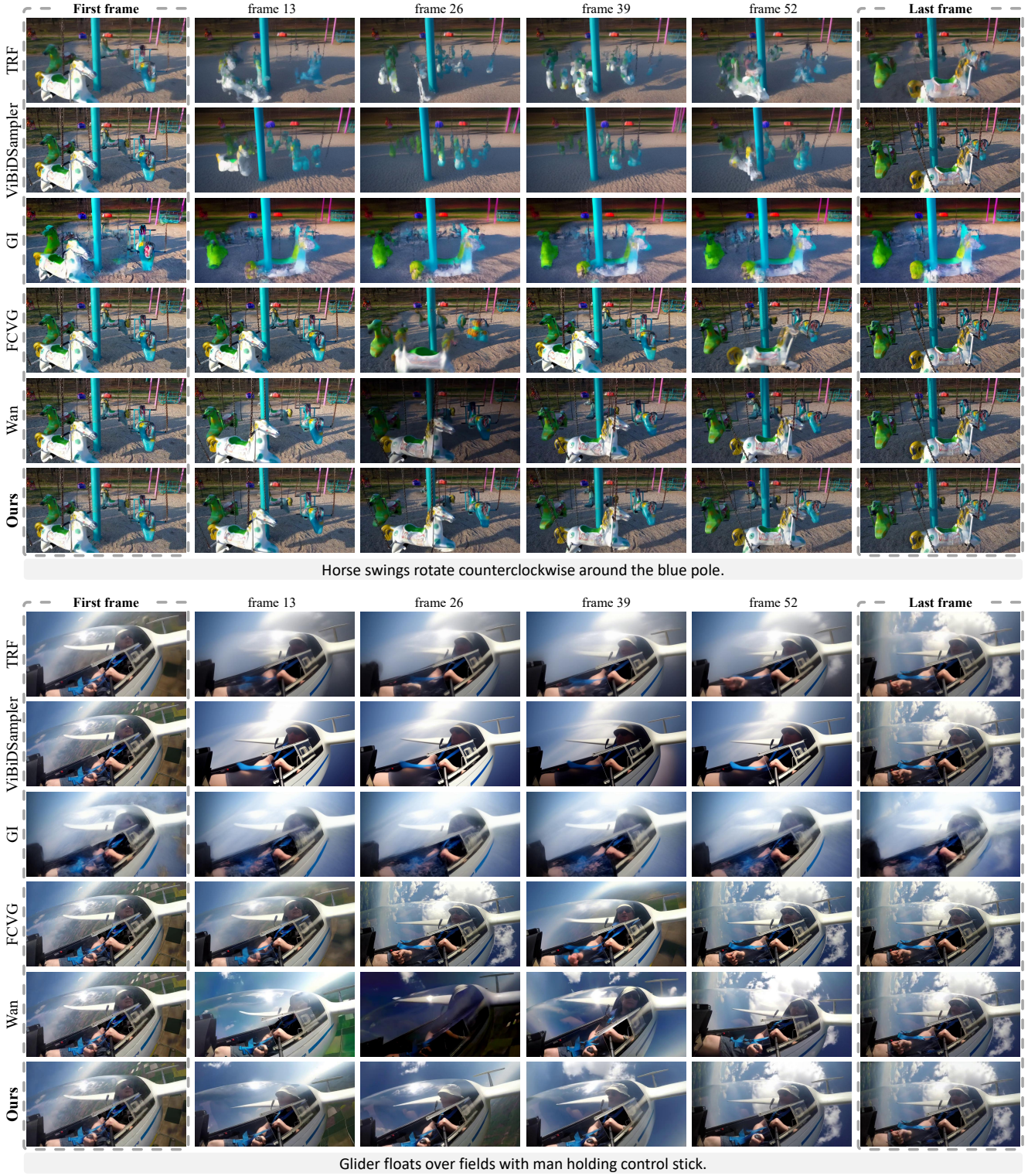

Figure S4. **Qualitative Results.** In both examples, our method generates consistent frames compared to Wan which shows artifacts or suddenly dimmed scenes. The first four models fail to maintain the object shape for the intermediate frames.

Figure S5. **Qualitative Results.** (Top) Other models, unlike ours, either show blurred objects with inconsistent frames or unnatural motion like Wan in frame 39. Our method shows high semantic fidelity as well as frame consistency through all frames. (Bottom) For the first four models, the structure of the rollercoaster collapses, failing to maintain the shape and style of the keyframes. Our model shows pace stability while maintaining the frame consistency.

Figure S6. **Qualitative Results.** Examples showing that our method performs well in highly complex scenes with multiple people and objects, preserving fine details and producing fewer blurred scenes than baseline methods.

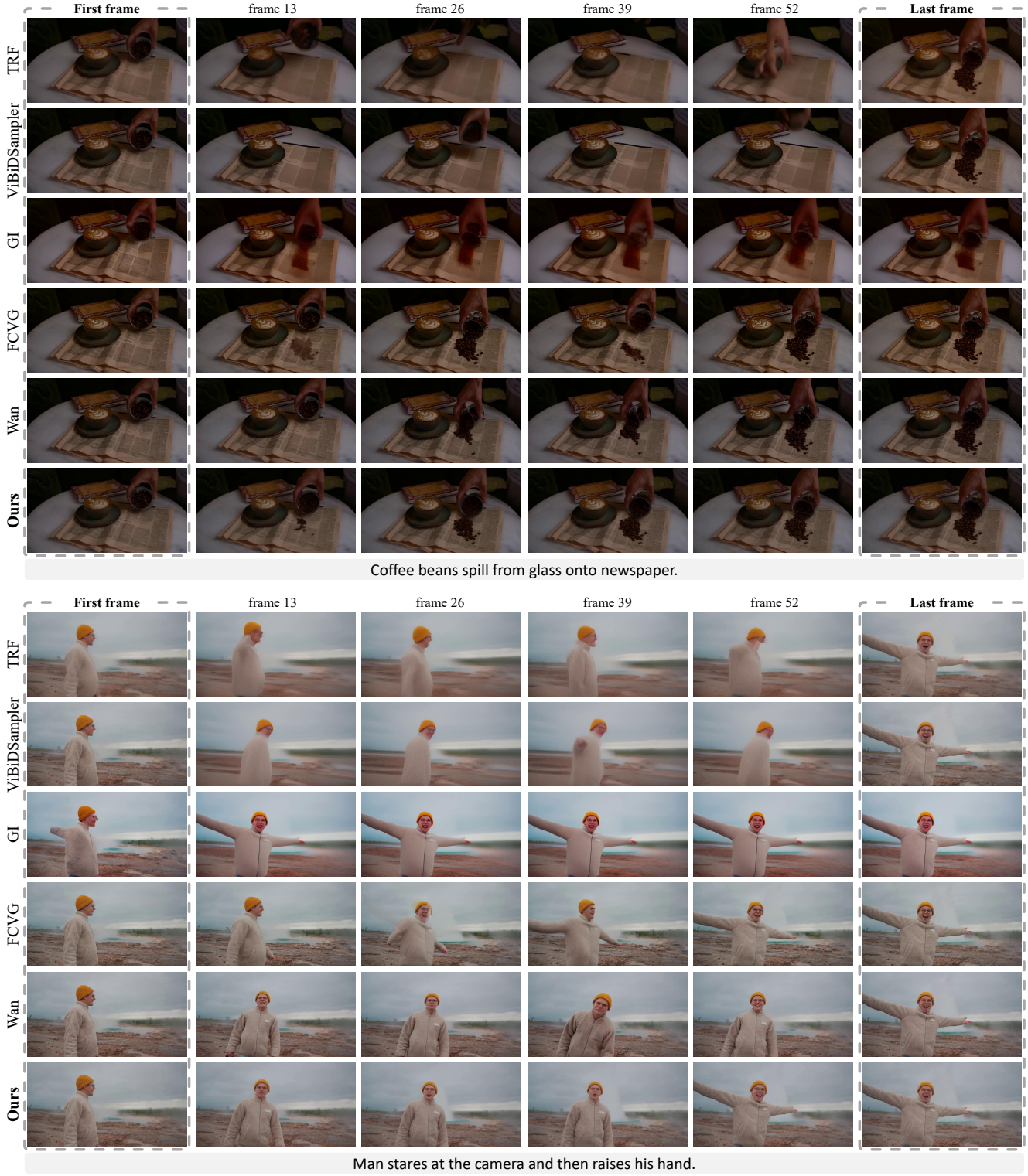

Figure S7. **Qualitative Results.** (Top) The first four baselines show unstable coffee-spilling pace and temporal inconsistency, while even compared to Wan our method generates more stably paced and temporally coherent motion. (Bottom) The first four baselines suffer from blur that distorts the human shape. Compared to Wan, our method maintains a more stable pace and generates more natural motions.

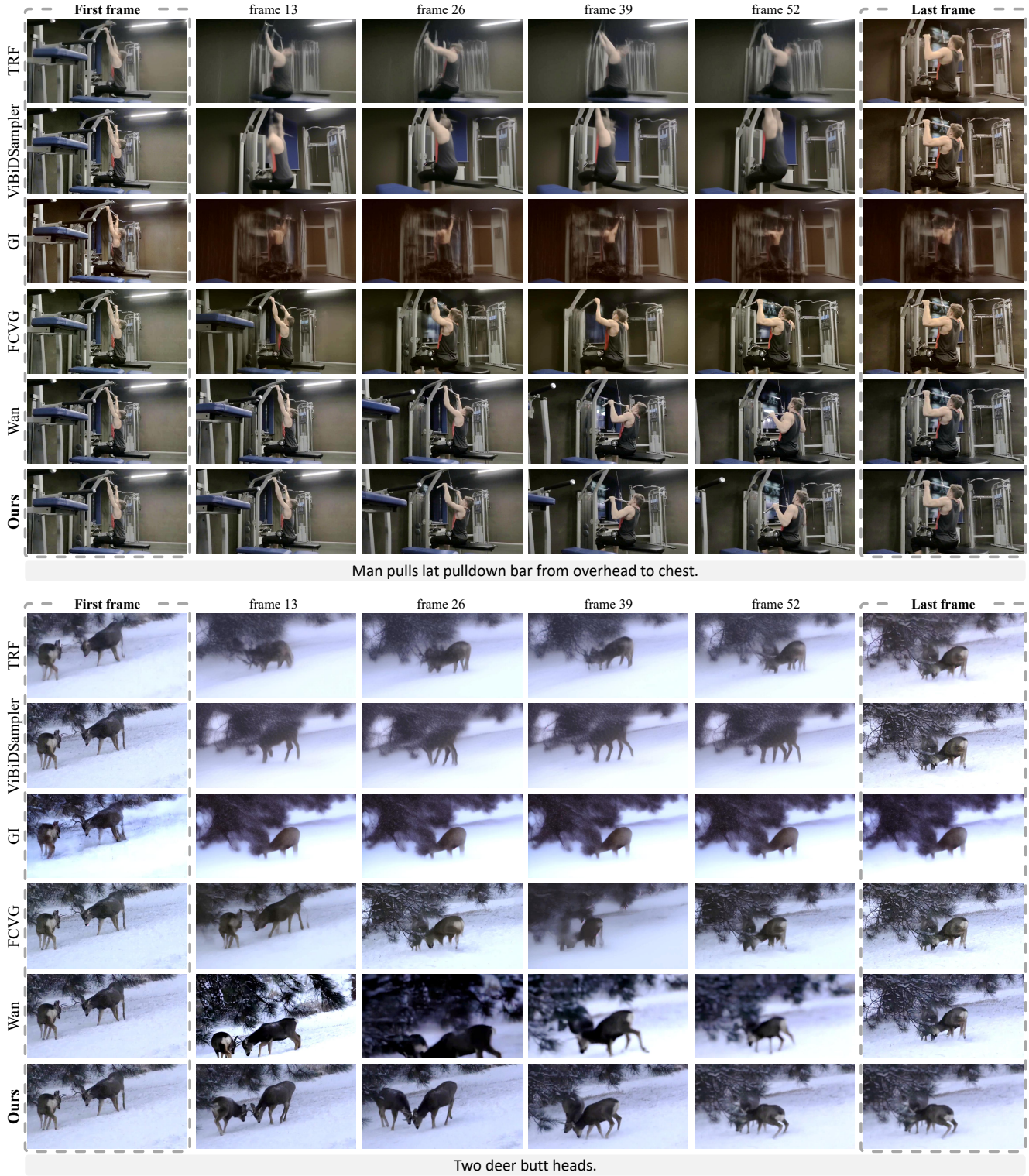

Figure S8. **Qualitative Results.** (Top) The first four baselines produce overly blurred frames where the human shape is not preserved and even compared to Wan, our method exhibits a more stable motion pace for the man performing lat pulldown. (Bottom) While other methods contain several blurred and inconsistent frames, our method generates clearer and more temporally consistent videos. For visualization purposes, we uniformly increased the brightness of both examples by 40%, while leaving all other properties unchanged.

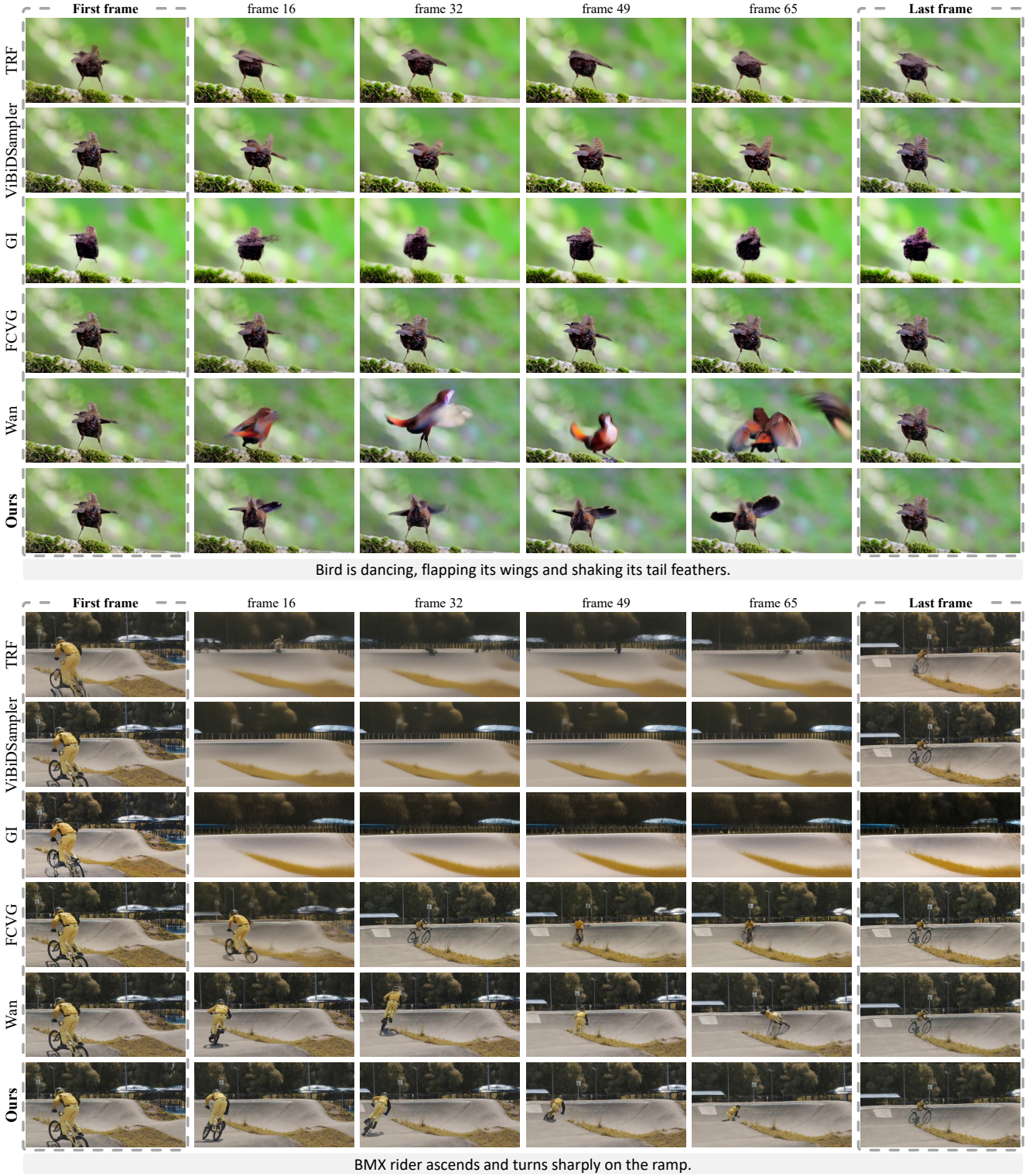

Figure S9. **Qualitative Results.** (Top) For the first four models, there is minimal wing flapping and motion, while our method and Wan show movements. However, Wan fails to maintain frame consistency and semantic fidelity. (Bottom) For Wan, the person on the bicycle goes left in the first few frames but suddenly turns from the right. On the other hand, our method shows consistent pace and consistency in movements.

Figure S10. **Qualitative Results.** The first four models, unlike Wan and ours, fail to maintain the shape of the object as well as the background style through the long frame sequences, showing the importance of correct text prompts in generative inbetweening.

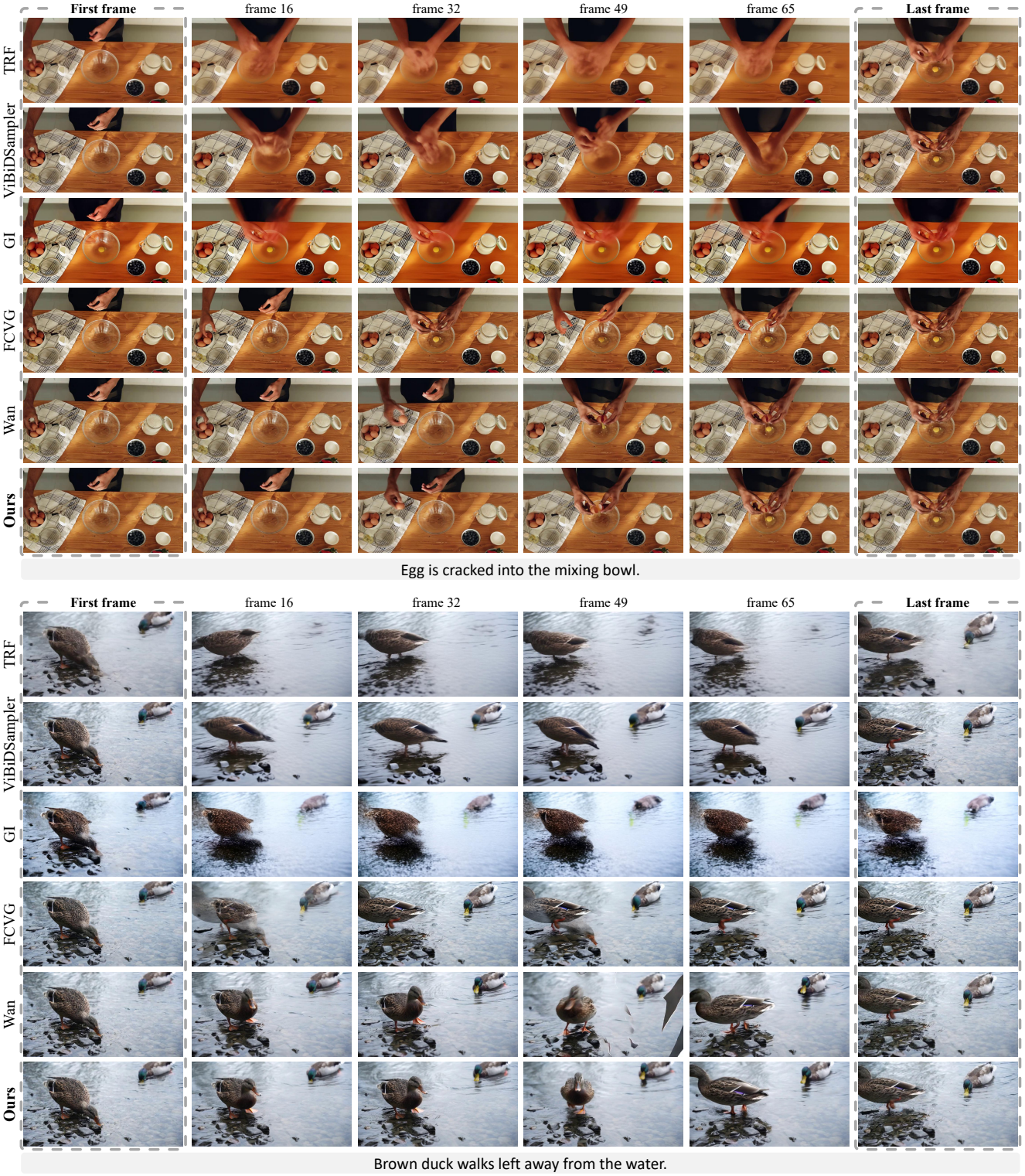

Figure S11. **Qualitative Results.** (Top) The first three models fails to maintain the shape of the hand and egg, while FCVG shows unnatural movement around frame 49 compared to the following two models, Wan and ours. (Bottom) For Wan, an artifact can be observed in frame 49, unlike our method.

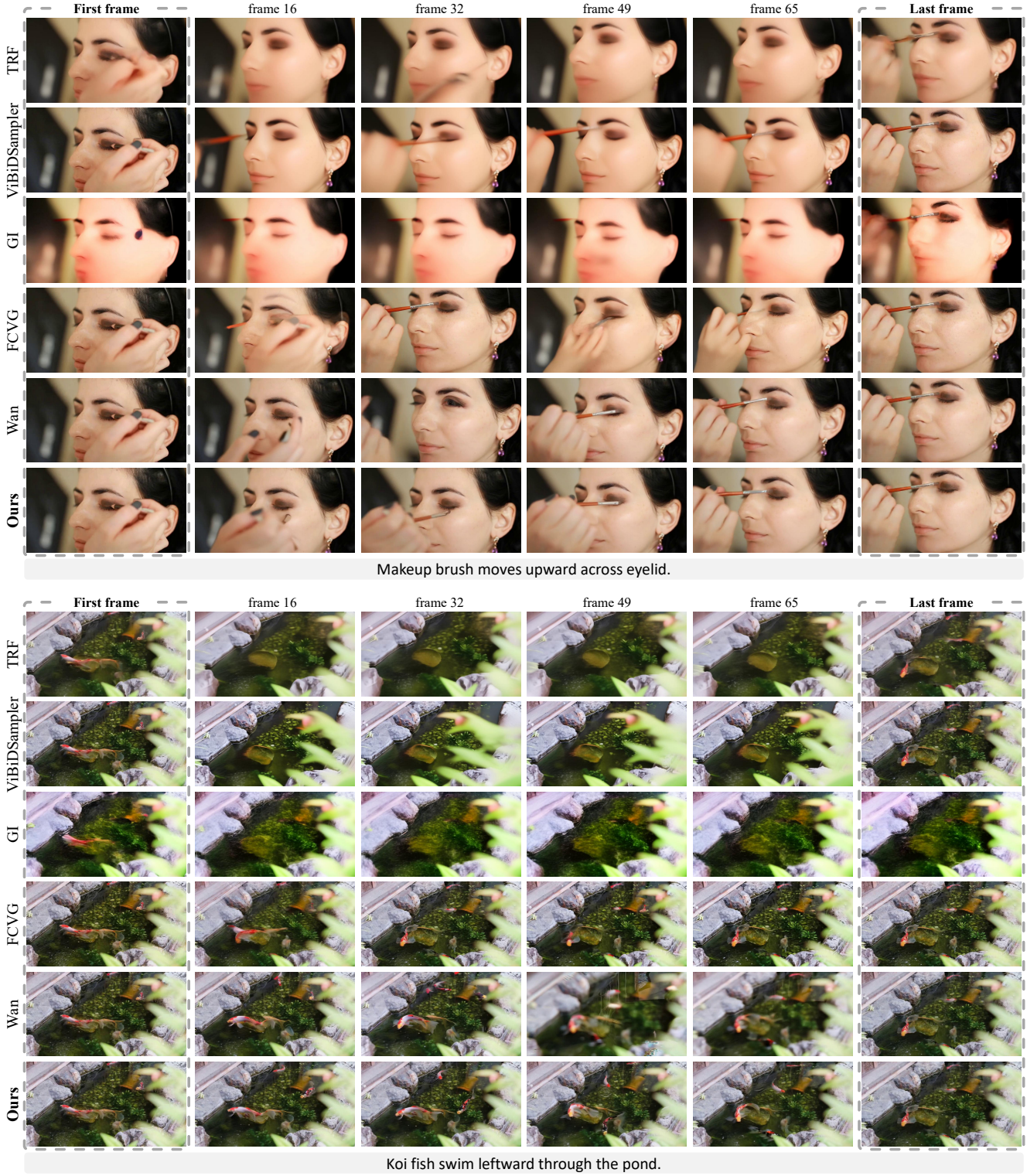

Figure S12. **Qualitative Results.** (Top) While Wan maintains the subject through the long sequence, it does not follow the prompt especially around frame 32. On the other hand, our method faithfully follows the text showing semantic fidelity. (Bottom) Around frame 49-65, Wan shows blurred scene without any context. On the other hand, this problem does not show up on our method.

References

[1] OpenAI et al. Gpt-4 technical report, 2024. 2, 5

[2] Team Wan et al. Wan: Open and advanced large-scale video generative models, 2025. 2, 7

[3] Haiwen Feng, Zheng Ding, Zhihao Xia, Simon Niklaus, Victoria Abrevaya, Michael J. Black, and Xuaner Zhang. Explorative inbetweening of time and space, 2024. 2

[4] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 3

[5] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 3

[6] Dahyeon Kye, Changhyun Roh, Sukhun Ko, Chanho Eom, and Jihyong Oh. Acevfi: A comprehensive survey of advances in video frame interpolation, 2025. 6

[7] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation, 2018. 6, 7

[8] Patrick E Shrout and Joseph L Fleiss. Intraclass correlations: uses in assessing rater reliability. *Psychological bulletin*, 86(2):420, 1979. 2

[9] 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. 3

[10] Xiaojuan Wang, Boyang Zhou, Brian Curless, Ira Kemelmacher-Shlizerman, Aleksander Holynski, and Steven M. Seitz. Generative inbetweening: Adapting image-to-video models for keyframe interpolation, 2025. 2

[11] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 3

[12] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report, 2024. 2

[13] Serin Yang, Taesung Kwon, and Jong Chul Ye. Vibidsampler: Enhancing video interpolation using bidirectional diffusion sampler, 2025. 2

[14] Liping Yuan, Jiawei Wang, Haomiao Sun, Yuchen Zhang, and Yuan Lin. Tarsier2: Advancing large vision-language models from detailed video description to comprehensive video understanding, 2025. 2

[15] Pan Zhang, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Rui Qian, Lin Chen, Qipeng Guo, Haodong Duan, Bin Wang, Linke Ouyang, Songyang Zhang, Wenwei Zhang, Yining Li, Yang Gao, Peng Sun, Xinyue Zhang, Wei Li, Jingwen Li, Wenhai Wang, Hang Yan, Conghui He, Xingcheng Zhang, Kai Chen, Jifeng Dai, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output, 2024. 2

[16] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 3

[17] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data, 2025. 2

[18] Tianyi Zhu, Dongwei Ren, Qilong Wang, Xiaohe Wu, and Wangmeng Zuo. Generative inbetweening through frame-wise conditions-driven video generation, 2024. 2