

# MotionMatcher: Cinematic Motion Customization of Text-to-Video Diffusion Models via Motion Feature Matching

## Supplementary Material

### A. Extended derivations

Below is the derivation of Eq. (2). We apply the generalized formula in DDIM [15] to compute the less noisy video at timestep  $t - 1$  (denoted as  $v_t$ ), using the noisy video  $z_t$  at timestep  $t$  along with the predicted noise  $\epsilon$ :

$$\begin{aligned} v_t(\epsilon, z_t) = & \underbrace{\sqrt{\bar{\alpha}_{t-1}} \left( \frac{z_t - \sqrt{1 - \bar{\alpha}_t} \epsilon}{\sqrt{\bar{\alpha}_t}} \right)}_{\text{"predicted } z_0"} \\ & + \underbrace{\sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} \cdot \epsilon}_{\text{"direction pointing to } z_t"} \\ & + \underbrace{\sigma_t \epsilon_t}_{\text{random noise}} \end{aligned} \quad (1)$$

where  $\bar{\alpha}_t := \prod_{s=1}^t \alpha_s$  are variance-scaling coefficients [4],  $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  is Gaussian noise, and  $\sigma$  is a hyperparameter controlling the stochasticity of the sampling process.

We observe that reducing randomness (*i.e.* using a lower value of  $\sigma_t$ ) improves feature extraction. Thus, following DDIM, we set  $\sigma_t = 0$ . This simplifies the equation to:

$$v_t = \sqrt{\bar{\alpha}_{t-1}} \left( \frac{z_t - \sqrt{1 - \bar{\alpha}_t} \epsilon}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \epsilon \quad (2)$$

which can be further simplified as:

$$v_t = \frac{1}{\sqrt{\bar{\alpha}_t}} z_t + \left( -\frac{\sqrt{1 - \bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} + \sqrt{1 - \bar{\alpha}_{t-1}} \right) \epsilon \quad (3)$$

Next, the DDPM objective can be reformulated to compare the previous noised videos  $z_{t-1}$ :

$$L = \mathbb{E}_{z_0, t, \epsilon} \left[ w_t \|\epsilon - \epsilon_\theta(z_t, t, c)\|^2 \right] \quad (4)$$

$$= \mathbb{E}_{z_0, t, \epsilon} \left[ w'_t \|v_t(z_t, \epsilon) - v_t(z_t, \epsilon_\theta(z_t, t, c))\|^2 \right] \quad (5)$$

where:

$$w'_t = \left( -\frac{\sqrt{1 - \bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} + \sqrt{1 - \bar{\alpha}_{t-1}} \right)^{-1} w_t \quad (6)$$

The time-dependent weight  $w_t$  is commonly set to 1. However, we employ a different weighting, where  $w'_t$  is 1 for the first 500 steps and to 0 for the last 500 steps. This weighting approach prioritizes the early stages, which are crucial for deciding video motion.

### B. Limitations

One limitation of MotionMatcher is that it requires a feature extractor to compute the objective, which introduces additional latency and results in longer training time (15 minutes) compared to pixel-level fine-tuning approaches [9, 21] (8 minutes) on an NVIDIA GeForce RTX 4090. Furthermore, since MotionMatcher relies on pre-trained T2V diffusion models, it struggles to synthesize videos that fall outside the generative prior of these models. However, we believe that this challenge can be mitigated as more advanced T2V diffusion models are developed in the future.

Like other existing approaches, another limitation of MotionMatcher lies in its reliance on DDIM-inverted noise (See Appendix F for details), which introduces a potential risk of content leakage from the reference video. As this issue is common among most existing approaches, addressing it will be an important direction for future research.

### C. Analysis of motion features

We conduct a simple retrieval experiment to verify that our motion feature extractor is capturing motion information from noisy videos. From the SVW dataset [14], we draw 139 javelin video clips with diverse motion trajectories and camera movements and randomly trim each clip to 16 frames. We obtain their motion features by adding noise to each video  $z$  and feeding them into our motion feature extractor as follows:

$$\mathcal{M}(\sqrt{\bar{\alpha}_t} z + \sqrt{1 - \bar{\alpha}_t} \epsilon), \quad (7)$$

where  $\mathcal{M}$  denotes our motion feature extractor, and the time step  $t$  is set to 500 for this experiment. After getting the motion features of all videos, we randomly select a query video and retrieve the most similar video from the dataset based on these motion features.

As shown in Fig. 1, the video with the most similar motion features shares the same motion despite having different appearances. In contrast, the video that is most similar in latent space has a nearly identical appearance but opposite motion, while the video with the most similar residual frames contain unrelated motion.

To compute the retrieval accuracy statistically, we label the videos with the top 10% smallest motion discrepancy values with the query video as positive samples and the rest 90% of the videos as negative samples. Next, we compute the average precisions (AP) for each retrieval methods to assess their retrieval accuracy. As presented in Tab. 1, our mo-

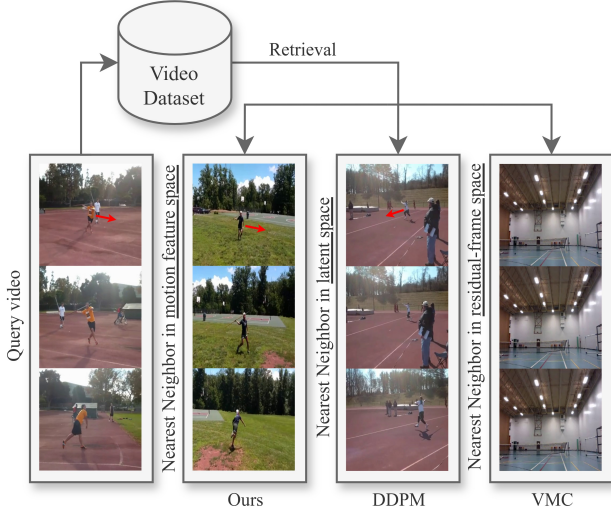


Figure 1. **Motion Retrieval.** Compared to DDPM [4] (using latent values) and VMC [9] (using frame differences), using the proposed motion features to perform motion retrieval shows preferable results. Note that the nearest neighbor in the motion feature space is retrieved by matching the motion features of the query video with those of the video dataset.

tion features yield the highest accuracy, indicating that they have the strongest correlation with actual motion. These results verify that our motion features capture rich motion information, rather than irrelevant details about visual appearance.

	Ours	DDPM	VMC	Random
AP	<b>32.78%</b>	8.20%	8.85%	10.71%

Table 1. **Retrieval accuracy.** Using our motion features to extract videos with similar motion yields the highest average precision (AP) than directly using latent videos (DDPM [4]) or their residual frames (VMC [9]).

## D. Additional qualitative results

We present additional qualitative comparisons in Fig. 4, detailed qualitative results in Fig. 2, and further samples generated using CogVideoX [19] as the base model in Fig. 3.

## E. Must motion be learned at feature level?

Analyzing video motion requires the ability to identify (1) scene composition and (2) the patterns of changes across frames (*i.e.* zooming, rotation, and displacement). Both of them are high-level concepts. The high-level nature of motion is also evident in optical flow estimation, a longstanding focus of research in video motion analysis. Early efforts

in this domain primarily relies on rule-based algorithms that use handcrafted rules to model motion [1, 2, 5, 16]. However, such methods often struggle with complex motion, such as large displacements, non-rigid movements, and motion in low-texture regions, all due to their lack of high-level understanding of videos.

With advances in machine learning, recent studies on optical flow estimation have shifted towards data-driven methods that learn motion patterns from large datasets [3, 7, 8, 13, 17]. These approaches have significantly improved motion estimation by leveraging deep neural networks to understand motion at the feature level, highlighting the importance of a high-level understanding of motion.

In the context of motion customization, given that motion is inherently a high-level concept, pixel-level objectives, such as frame-difference matching [9, 11, 21], are insufficient for capturing motion. These objectives often fail to capture complex motion, facing the same challenge as early research on optical flow estimation. In contrast, our method precisely extracts motion information with the assistance of a deep neural network. By leveraging a large pre-trained model, our method can understand at a high level and captures key information such as scene composition and patterns of changes.

## F. Implementation details

**Training** To fine-tune the diffusion model, we add LoRAs to all self-attention and feed forward layers, and set the rank to 32. Since motion is mainly determined in early stages [10, 20], we set the time-dependent weights  $w'_t$  in the objective function to 1 for the first 500 timesteps and 0 for the last 500 timesteps. The LoRA [6] are optimized for 400 steps at a learning rate of 0.0005, which takes approximately 15 minutes on an NVIDIA GeForce RTX 4090. All videos in the experiments consist of 16 frames at 8 fps and are generated at a resolution of  $384 \times 384$ .

**Feature extraction** We extract cross-attention maps and temporal-self attention maps from down\_block.2 at a  $12 \times 12$  resolution. Both  $M_{CA}$  and  $M_{TSA}$  represent the average of all extracted attention maps across heads and layers, which we omit in all equations for conciseness.

**Initial noise** Following previous work on motion customization [9, 11, 20, 21], we utilize DDIM inversion to obtain the initial noise  $z_T$  for better motion alignment. In our work, the initial noise  $z_T$  is computed as in MotionDirector’s implementation:

$$z_T = \sqrt{\beta} \epsilon_{\text{inv}} + \sqrt{1 - \beta} \epsilon \quad (8)$$

where  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  is Gaussian noise, and  $\epsilon_{\text{inv}}$  represents the inverted noise of the reference video, derived via DDIM

inversion [15]. The square root terms in the equation ensure that the variance of  $z_T$  remains consistent across all values of  $\beta$ . In quantitative experiments and human user study, we set a fix value of  $\beta = 0.3$ . In other experiments,  $\beta$  varies between the range of 0.0 to 0.3.

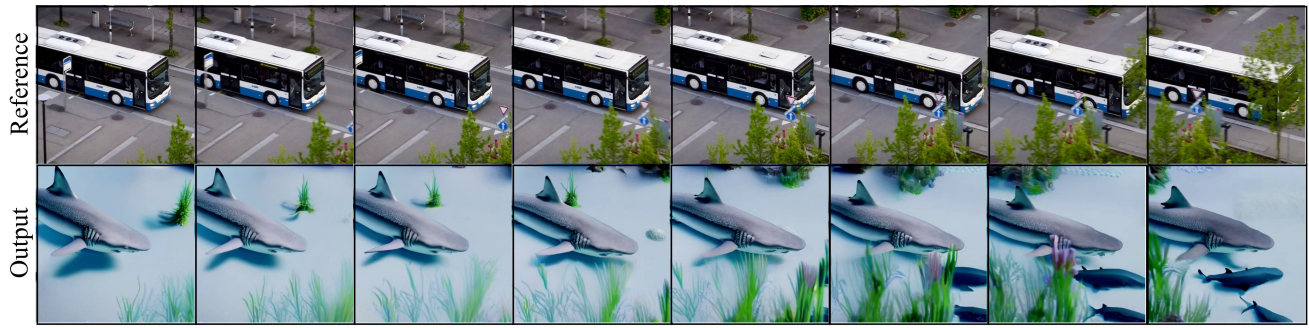
## G. Evaluation details

**Dataset** We collect a dataset of 42 video-text pairs, including 14 unique reference videos from DAVIS [12] and LOVEU-TGVE [18], many of which are also used in prior work. For each reference video, we provide exactly 3 target text prompts that describe scenes distinct from the original one and ensure that they are compatible with the motion in the reference video.

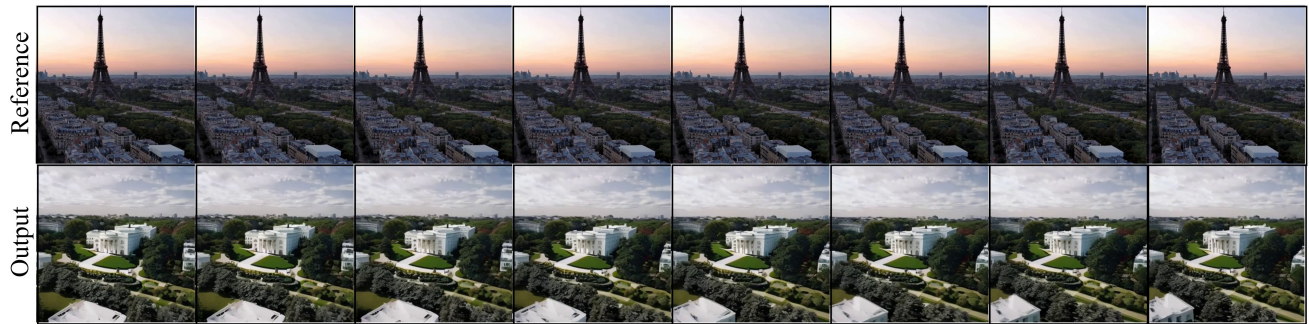
**Quantitative evaluation** To evaluate each method, we generate 5 videos per video-text pair, and calculate the average scores across all generated videos.

**Human user study** In the human user study, we employ the same set of videos generated in the quantitative experiments. Each survey consists of 32 tasks. In each task, the survey respondents are presented with a video-text pair, a video generated by our method, and a video generated by one of the four competing methods (Fig. 5). The video-text pair and videos for each task are randomly selected on the fly, resulting in a total of  $4 \times 42 \times 5 \times 5 = 4200$  different tasks. To assess motion alignment, text alignment, and video quality, the participants are asked three questions: "Which video better matches the motion of the following video?", "Which video better matches the following text?", and "Which video has better video quality (i.e., more realistic and visually appealing)?". To ensure a fair comparison, the order of the choices is randomized.

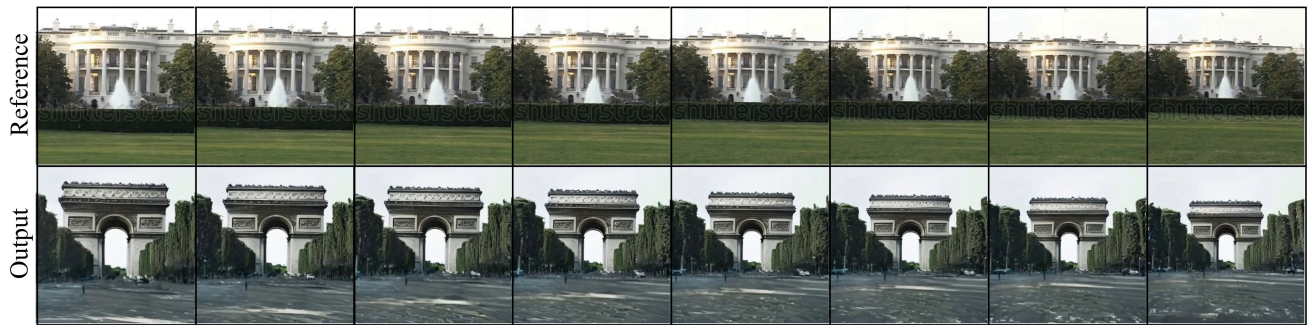




Prompt: *"A shark is swimming."*



Prompt: *"Drone flyover of the White House."*



Prompt: *"Close up shot of the Arc de Triomphe."*



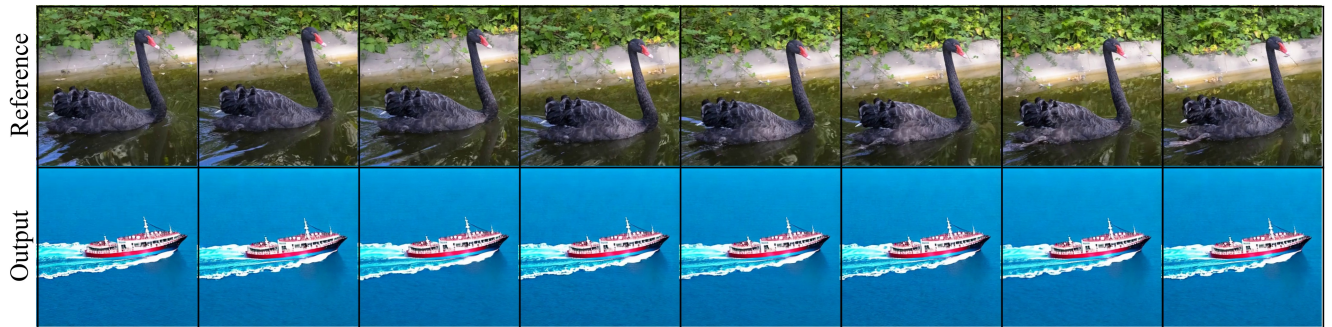
Prompt: *"An arctic fox is walking on ice."*

Figure 2. **Additional qualitative results.** The results demonstrate MotionMatcher’s capability to transfer both object movements and camera movements to new scenes.





Prompt: "An arctic fox is walking on ice."



Prompt: "A ship is sailing on the sea."



Prompt: "Basketball spin."

Figure 3. **More samples generated using CogVideoX [19] as the base model.** The results demonstrate the generality of MotionMatcher. Even with T2V diffusion models that employ full attentions, we can still extract cues for objects movement from attention weights computed between frames and cues for camera framing from attention weights computed between words and patch tokens.



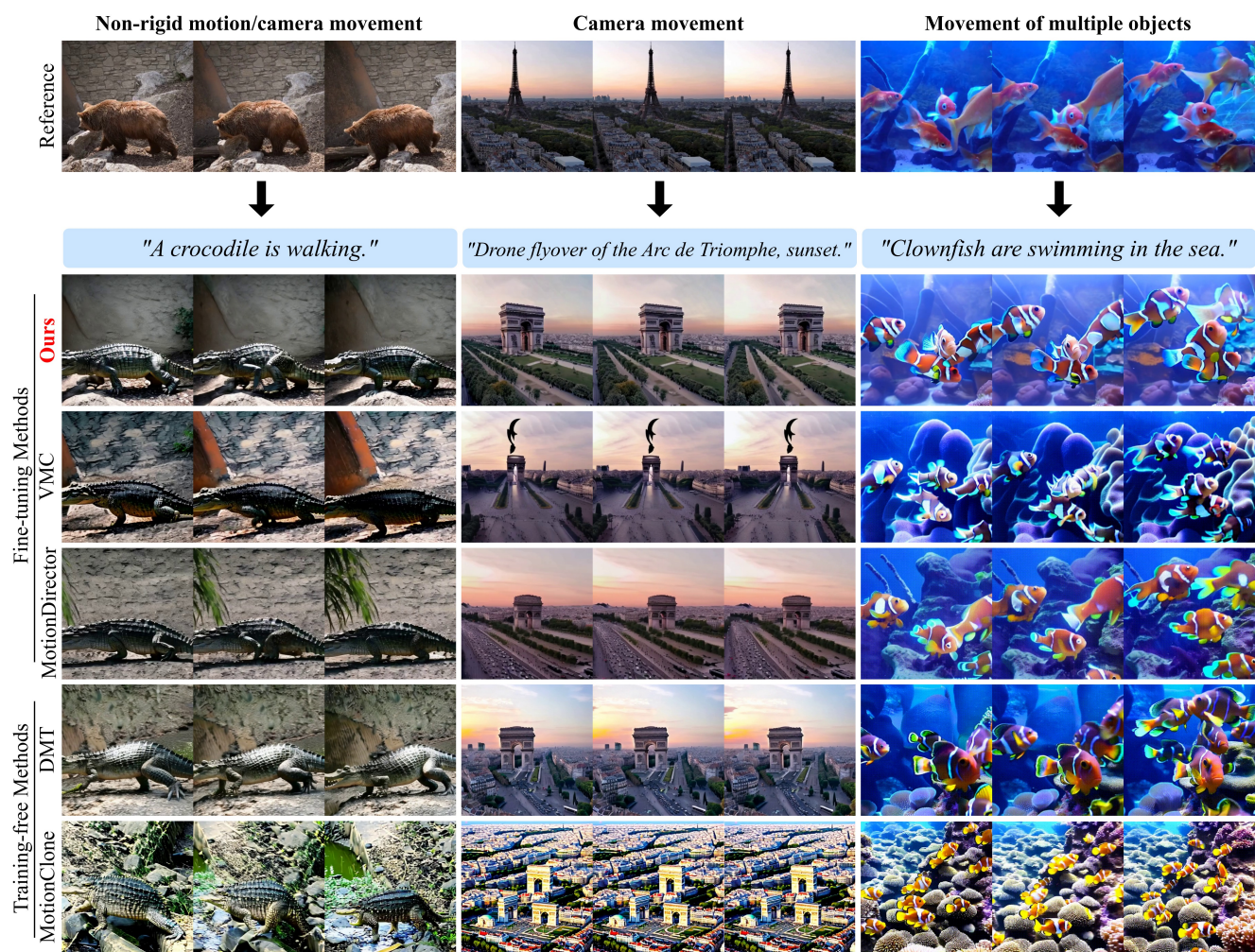
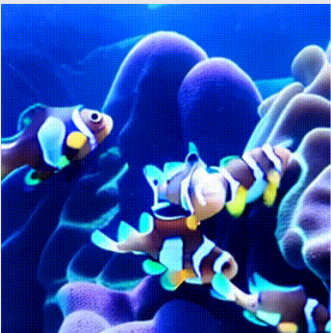
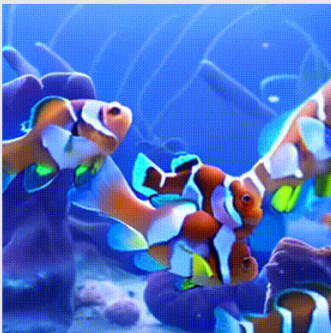


Figure 4. **Additional qualitative comparisons.** The results demonstrate MotionMatcher’s superiority over existing motion customization methods in terms of video quality, text alignment, and motion alignment.

Video 1



Video 2





☐ Video 1

☐ Video 2

☐ Video 1

☐ Video 2

☐ Video 1

☐ Video 2

Figure 5. **User interface of an evaluation task.** Each task includes three questions, each assessing a key aspect of motion customization.



## References

- [1] Michael J Black and Padmanabhan Anandan. A framework for the robust estimation of optical flow. In *1993 (4th) International Conference on Computer Vision*, pages 231–236. IEEE, 1993. 2
- [2] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International journal of computer vision*, 61:211–231, 2005. 2
- [3] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 2
- [4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2
- [5] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. 2
- [6] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2
- [7] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. In *European conference on computer vision*, pages 668–685. Springer, 2022. 2
- [8] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 2
- [9] Hyeonho Jeong, Geon Yeong Park, and Jong Chul Ye. Vmc: Video motion customization using temporal attention adaptation for text-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9212–9221, 2024. 1, 2
- [10] Pengyang Ling, Jiazi Bu, Pan Zhang, Xiaoyi Dong, Yuhang Zang, Tong Wu, Huaian Chen, Jiaqi Wang, and Yi Jin. Motionclone: Training-free motion cloning for controllable video generation. *arXiv preprint arXiv:2406.05338*, 2024. 2
- [11] Geon Yeong Park, Hyeonho Jeong, Sang Wan Lee, and Jong Chul Ye. Spectral motion alignment for video motion transfer using diffusion models. *arXiv preprint arXiv:2403.15249*, 2024. 2
- [12] 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. 3
- [13] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 2
- [14] Seyed Morteza Safdarnejad, Xiaoming Liu, Lalita Udpa, Brooks Andrus, John Wood, and Dean Craven. Sports videos in the wild (svw): A video dataset for sports analysis. In *2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, pages 1–7. IEEE, 2015. 1
- [15] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1, 3
- [16] Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106:115–137, 2014. 2
- [17] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 2
- [18] Jay Zhangjie Wu, Difei Gao, Jinbin Bai, Mike Shou, Xiyu Li, Zhen Dong, Aishani Singh, Kurt Keutzer, and Forrest Iandola. The text-guided video editing benchmark at loveu 2023. <https://sites.google.com/view/loveucvpr23/track4>, 2023. 3
- [19] 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. 2, 5
- [20] Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8466–8476, 2024. 2
- [21] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jiawei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. *arXiv preprint arXiv:2310.08465*, 2023. 1, 2