

IM-Zero: Instance-level Motion Controllable Video Generation in a Zero-shot Manner

Supplementary Material

1. Implementation Details

1.1. Instance Motion Control Implementation

Motion Generation Stage. For keyframe candidates generation, we employ InstanceDiffusion [21] as the grounded text-to-image model. For coarse motion video generation, we utilize AnimateDiff [2] as the text-to-video model. In this process, we incorporate Stable Diffusion v1.5 [19] for spatial layers and animatediff-motion-adapter v1-5-3 [2] for motion layers. Additionally, we integrate SparseCtrl [3] as the ControlNet, which leverages the canny image of keyframe candidates to produce videos. The rationale behind using SparseCtrl [3] lies in the possibility of the keyframe candidates’ quantity being less than the number of frames in the video. Furthermore, in the modification of AnimateDiff [2] through cross-frame attention, we specifically substitute self-attention layers within the `mid_block.attentions.0.transformer.blocks.0.attn1` block of UNet with cross-frame attention layers.

Video Refinement Stage. In this stage, we utilize AnimateDiff [2] as the text-to-video model for motion injection, where we also incorporate Stable Diffusion v1.5 [19] for spatial layers and animatediff-motion-adapter v1-5-3 [2] for motion layers. Besides, we utilize Stable Diffusion v1.5 [2] as the text-to-image model for detail injection.

Injection Timestep Strategy. Moreover, we discovered that implementing motion injection and detail injection only at specific timesteps is enough, which remarkably reduces the inference time. We have found that motion injection can be confined to the initial half of the denoising process, while detail injection can be focused on the latter half. Specifically, we introduce a threshold $t_{\text{threshold}}$ to delineate the denoising phase. For $t \sim [t_{\text{threshold}}, \dots, T]$, we sample several timesteps for motion injection, and for $t \sim [1, \dots, t_{\text{threshold}}]$, we sample several timesteps for detail injection. This can be explained by the properties of the denoising process of diffusion models. According to [25], diffusion models tend to reconstruct low-frequency components at the initial timesteps and reconstruct high-frequency components later. For T2V models, low-frequency components often correspond to the spatio-temporal correlation within videos, closely linked to motion [23]. Hence, we conduct motion injection in the early stages. On the other hand, as high-frequency components correspond to the finer details within frames, we perform detail injection in the later stages.

1.2. Versatile Capacity Implementation

In addition to overall instance motion control, IM-Zero offers versatile capacity advantages over comparative methods. First, IM-Zero can control the subparts of instances. Users can first determine the spatial location of the instance and then separately specify the spatial location and corresponding movement trajectory of the subparts. This allows the grounded text-to-image model to generate the instance and its subparts at the specified locations. IM-Zero then generates a coarse motion video and refines it through the Video Refinement Stage.

Second, IM-Zero allows users to specify instance shapes more precisely using masks. We use InstanceDiffusion [21] as the grounded text-to-image model. InstanceDiffusion processes the mask input similarly to bounding boxes by sampling it into a sequence of points and encoding it into instance tokens, ensuring that the generated instance’s shape conforms to the mask input.

Third, IM-Zero allows users to perform motion transfer to customize more complex motion patterns using reference videos. As is introduced in Section 3.5, we first extract control signals (e.g., depth maps) from the source video. The control signals and the target prompt are then input into a text-to-video model with ControlNet to generate a coarse video, which aligns with both the source motion pattern and the target prompt. Finally, the coarse motion video is refined through the Video Refinement Stage. Additionally, we can also add ControlNet to the Video Refinement Stage for better motion transfer effects, and in this case, we draw control signals from the coarse motion video.

Moreover, IM-Zero allows high-quality text-to-video generation with text inputs only. Similarly, we employ only a text-to-video model to generate a coarse video according to the text input. Then the coarse video is refined via the Video Refinement Stage to enhance video quality.

2. Experimental Settings

2.1. Dataset Curation

For quantitative evaluation, real-shot videos are needed as ground truth, from which instance spatial locations and movement trajectories are extracted to simulate user inputs. Therefore we utilize the training set and validation set of the DAVIS-17 dataset [16] and the test set of the GOT10k dataset [5]. We performed a series of preprocessing steps on the videos. First, we uniformly sampled 16 frames from each video. Then, we cropped a

frame_height×frame_height region from the center of each frame and resized it to 512×512. Finally, we filtered out videos that no longer contained instances.

After sampling and cropping the videos, we use BLIP-2 [11] to generate text prompts and GroundingDINO [13] to extract bounding boxes from the first and last frames, generating the inputs required by IM-Zero and the comparative methods. Then, we eliminate videos where GroundingDINO [13] failed to detect bounding boxes. Finally, we randomly select 50 videos and 70 videos from the rest videos of DAVIS-17 [16] and GOT10k [5]. The extracted inputs are then used to generate videos with a resolution set at 512×512, a frame number of 16.

2.2. Metric Selection and Implementation

For the evaluation metrics, we employ mean Intersection over Union (mIoU) and Centroid Distance (CD) to assess the match between the generated videos and the spatial control conditions specified by the user inputs. We use Fréchet Video Distance (FVD) [20] and Kernel Inception Distance (KID) [1] to evaluate video quality, and CLIP similarities (CLIPSim) [18] to assess frame-to-frame consistency.

Specifically, mIoU and CD calculations utilize the OWL-ViT-large open-vocabulary instance detector [15], following the methodologies of Peekaboo [7] and TrailBlazer [14]. For videos where instances are undetectable, mIoU is set to 0, and CD selects the farthest point within the video frame for computation as a penalty. The calculation of FVD is based on FID [4] following [20], and the implementation of FID follows StyleGAN [8]. The calculation of KID follows the implementation by TorchMetrics. And the implementation of CLIPSim employs the pre-trained clip-vit-large-patch14 version of the CLIP model [18].

For the parameters, we set $\lambda_1 = 0.8$ in Equation 2 and $\lambda_2 = 0.8$ in Equation 5. We employ 40 denoising timesteps and set $M = 4$ in Equation 2 and $D = 4$ in Equation 5. For the injection timestep strategy, we set $t_{\text{threshold}} = 20$. We perform motion injection 5 times at timesteps [22, 26, 30, 34, 38] and detail injection 10 times at timesteps [6, 7, 8, 9, 11, 13, 15, 17, 19].

3. Ablation Studies

3.1. Ablation on Motion Generation Stage

In the Motion Generation Stage, we apply a series of methods to enhance consistency in generating coarse motion videos. To validate the effectiveness, we conduct an ablation study on the DAVIS-17 dataset. We remove cross-frame attention from Keyframe Candidates Generation, eliminate cross-frame attention from Coarse Motion Video Generation, replace the designed initial noise with random noise, and omit the IP-Adapter, respectively. The parameters for each experiment are set entirely consistent.

The results, as depicted in Table A1, indicate that these methods effectively improve the quality of the final generated videos and also enhance the alignment with user inputs.

3.2. Ablation on Injection Method

We also conduct an ablation study on Motion Injection and Detail Injection. As shown in Figure A1, the coarse motion video exhibits issues such as low consistency and poor image quality (e.g., watermark). Solely utilizing motion injection can enhance consistency but lacks detail. Detail injection alone increases detail but introduces consistency problems like distortion. Our complete method enhances both consistency and detail, demonstrating the effectiveness.

3.3. Ablation on Injection Timestep Strategy

In our method, we employ an Injection Timestep Strategy as follows. We use a total of 40 denoising timesteps, performing motion injection at 5 timesteps and detail injection at 10 timesteps. To verify the effectiveness, we conduct ablation studies on this Injection Timestep Strategy using the DAVIS-17 dataset. Specifically, we conduct three additional sets of experiments. First, we try higher frequency injection which performs **injection for all timesteps**. We perform motion injection at every denoising timestep t for $t \sim [20, 21, \dots, 40]$, and perform detail injection at every denoising timestep t for $t \sim [6, 7, \dots, 20]$. Second, we tried **lower frequency injection**, performing motion injection 3 times at timesteps [21, 28, 35] and performing detail injection 5 times at timesteps [6, 9, 12, 15, 18]. Finally, we tried **only 1 timestep injection**, performing motion injection at the last timestep [40] and performing detail injection at the first possible timestep [6] (as $t - 1 - D$ should be at least 1, where we set $D = 4$). The result is as shown in Table A2.

For the first set of experiments, we performed injection at all timesteps. We observed an improvement in frame consistency on CLIPSim, which verifies the effectiveness of motion injection in enhancing motion smoothness and consistency. However, excessive motion injection led to blurry frames with a lack of detail, which could not be compensated by detail injection. Consequently, the video quality decreased in terms of FVD and KID, and this may also lead to incorrect detection of instances, resulting in a decrease in accuracy as measured by mIoU and CD. Additionally, excessive injection increased inference time. Therefore, it is necessary to control the frequency of injection.

For the second set of experiments, we slightly reduced the injection frequency, performing motion injection 3 times and detail injection 5 times. This resulted in outcomes very close to our method. The second set of experiments outperformed our method in mIoU and matched our method in CD, but was inferior in terms of video quality as measured by FVD and KID. For the third set of experiments, we performed only one motion injection and one

Table A1. Ablation on different components in Motion Generation Stage.

Method	FVD (\downarrow)	KID (\downarrow)	CLIPSim (\uparrow)	mIoU (\uparrow)	CD (\downarrow)
w/o 1st CFA	4320.07	26.84	98.84	0.26	0.29
w/o 2nd CFA	4528.09	28.08	98.78	0.21	0.31
w/o init noise	4786.17	26.69	98.82	0.26	0.31
w/o IP-Adapter	4549.94	27.75	98.87	0.26	0.28
Ours	4070.01	25.75	98.91	0.27	0.26

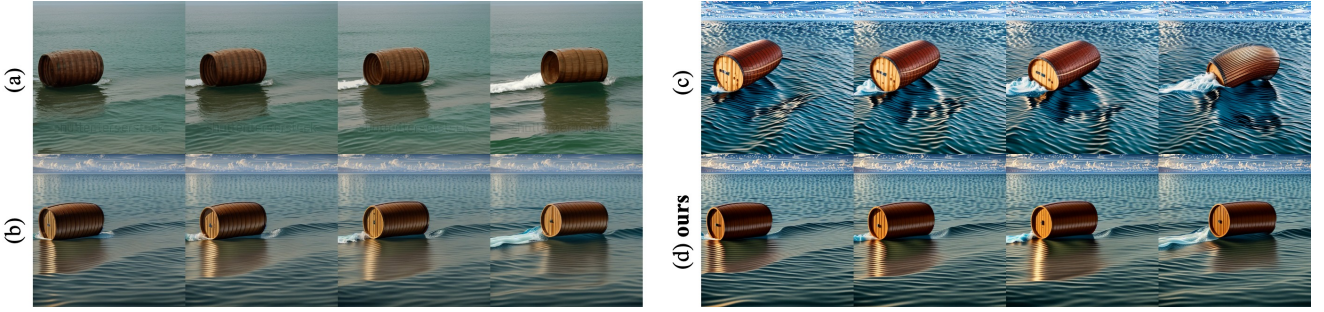


Figure A1. Ablation on different components in Zero-shot Video Refinement. (a) Coarse motion video exhibits low consistency and poor image quality like watermark. (b) Refinement w/o detail injection exhibits low detail richness. (c) Refinement w/o motion injection exhibits consistency problems like distortion. (d) Our complete refinement enhances both consistency and detail.

detail injection, leading to a decline in almost all metrics. Considering the actual visual observations by humans, we ultimately chose the current strategy of 5 motion injections and 10 detail injections.

4. Supplementary Evaluations and Discussion

We provide more comparison results and discussions in this section. As we present qualitative results in the video version, [please open the pdf with Adobe Acrobat Reader](#).

4.1. Instance Control Compared With T2V-Zero, Peekaboo, TrailBlazer and MotionBooth.

We present the video version qualitative results of Figure 4 in Figure A2. The results further demonstrate that our method is better in alignment with the input, motion effects, and video quality compared with other methods.

4.2. Instance Control Compared With FreeTraj

We further compare our method with FreeTraj [17]. For each instance, FreeTraj [17] requires inputting boxes with a fixed size across frames to make noise flow; otherwise, it fails to generate the corresponding video. This additional constraint introduces two issues. Firstly, it restricts the practical applicability of FreeTraj [17]. In contrast, neither T2V-Zero [9], Peekaboo [7], TrailBlazer [14], MotionBooth [22], nor our method is subject to this limitation. Secondly, this constraint precludes a direct comparison between FreeTraj and other methods under the experimental

settings described in Table 1, as the bounding boxes extracted by GroundingDINO [13] generally exhibit varying sizes across frames. Therefore, to ensure a fair comparison, we conduct an extra evaluation of FreeTraj [17] and our method under new experiment settings that satisfy the aforementioned constraint.

Experiment Settings. We generated 50 sets of input trajectories with unchanged box sizes using GPT. For each set of input trajectories, the corresponding text prompts were also generated by GPT. In terms of metrics, due to the absence of ground truth videos, we evaluate CLIPSim for temporal consistency across frames, and mIoU and CD for alignment with the input.

Quantitative results. The quantitative results are shown in Table A3. Our method significantly outperforms FreeTraj in all metrics, demonstrating superior temporal consistency across frames and better alignment with the trajectory input.

Qualitative results. The qualitative results are shown in Figure A3. The input can be found in the supplementary video. The results demonstrate that our method surpasses FreeTraj [17] in terms of alignment with the input, motion effects, and video quality.

4.3. Results on Motion Transfer

To validate the effectiveness of our method in motion transfer task, we compare with MOFT [10], MotionClone [12], and MotionDirector [26].

Experiment Settings. For the evaluation dataset, we use

Table A2. Ablation on Injection Timestep Strategy.

Method	FVD (\downarrow)	KID (\downarrow)	CLIPSim (\uparrow)	mIoU (\uparrow)	CD (\downarrow)
injection for all timesteps	4213.42	30.84	98.99	0.20	0.33
lower frequency injection	4214.44	26.23	98.73	0.28	0.26
only 1 timestep injection	4079.40	26.84	98.13	0.27	0.27
Ours	4070.01	25.75	98.91	0.27	0.26

(a) T2V-Zero

(b) Peekabo

(c) TrailBlazer

(d) MotionBooth

(e) **ours**

Figure A2. Instance control qualitative results compared with T2V-Zero [9], Peekaboo [7], TrailBlazer [14] and MotionBooth [22].

Table A3. Quantitative results compared with FreeTraj [17].

Method	CD (\downarrow)	mIoU (\uparrow)	CLIPSim (\uparrow)
FreeTraj	0.21	0.21	97.21
Ours	0.13	0.39	99.64

Table A4. Motion transfer results compared with other methods.

Method	Motion Fidelity (\uparrow)	Imaging Quality (\uparrow)
MotionClone [12]	74.5	0.72
MOFT [10]	51.2	0.71
MotionDirector [26]	75.5	0.68
Ours	63.7	0.73

Table A5. T2V results compared with UNet baseline.

Method	Imaging Quality (\uparrow)	CLIPSim (\uparrow)
AnimateDiff [2]	0.67	99.05
Ours	0.74	99.73

the open-sourced data from MotionClone [12]. For metrics, we use Motion Fidelity [24] and Imaging Quality [6] following MOFT [10]. The implementation of Motion Fidelity follows that of [24], and the implementation of Imaging Quality follows that of [6].

Quantitative results. The quantitative results are shown in Table A4. The results demonstrate that our method is competitive in Motion Fidelity [24] and is the best in Imaging Quality [6].

Qualitative results. The qualitative results are shown in Figure A4. The results also demonstrate that our method exhibits competitive motion transfer performance compared with other methods.

4.4. Results on Text-to-video Generation

As we use UNet-based models, we also compare T2V generation with UNet baseline AnimateDiff [2].

Experiment Settings. We randomly generate 50 prompts randomly generated by GPT and use these prompts as the only inputs to generate videos via our method and AnimateDiff. For metrics, as there are no ground truth videos, we use Imaging Quality [6] and CLIPSim to evaluate the video quality and consistency across frames.

Quantitative results. The quantitative results are shown in Table A5. The results demonstrate that our method performs better on both Imaging Quality and CLIPSim, indicating better video quality and consistency across frames.

Qualitative results. The qualitative results are shown in Figure A5. The results also demonstrate our method per-

(a) FreeTraj

(b) ours

Figure A3. Qualitative results compared with FreeTraj [17].

(a) source

(b) MotionClone

(c) MOFT

(d) MDirector

(e) ours

Figure A4. Motion transfer qualitative results compared with MOFT [10], MotionClone [12], and MotionDirector [26].

forms better in T2V generation compared with the baseline.

4.5. Autoregressive Consistency Improvement Results

The decomposition design of our pipeline allows autoregressive usage of the second stage to autoregressively enhance temporal consistency. For instance, we first generate the skateboard demo in the supplementary video and then input the generated video to the second stage to get a further refined video, as is shown in the last demo in Figure A6. As shown in Figure A6, the refined video has improved shoe consistency. To further validate the effectiveness of our method, we also provide comparison results with other zero-shot methods.

4.6. Results on Multi-instance Control

Our method also has multi-instance control capacity, as is shown in Figure A7. In comparative methods, only TrailBlazer [14] supports multi-instance control, we also present the video generated by TrailBlazer [14] as a comparison. The results demonstrate that our method has better multi-instance control capacity compared with TrailBlazer [14].

5. Additional Examples of Generated Videos

More videos are available in the MP4 file in the supplementary multimedia.

References

- [1] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 2
- [2] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 1, 4
- [3] 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, 2025. 1
- [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. 2
- [5] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1562–1577, 2019. 1, 2
- [6] Ziqi Huang, Yanan 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. 4
- [7] Yash Jain, Anshul Nasery, Vibhav Vineet, and Harkirat Behl. Peekaboo: Interactive video generation via masked-diffusion. In *Proceedings of the IEEE/CVF Conference*

(a) AnimateDiff

(b) **ours**

Figure A5. T2V results.

(a) Peekaboo

(b) TrailBlazer

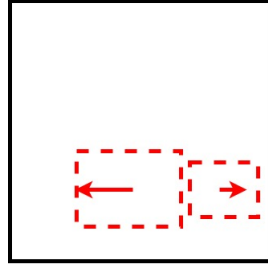
(c) MotionBooth

(d) FreeTraj

(e) **ours**

Figure A6. Comparison of skateboard demo.

- on *Computer Vision and Pattern Recognition*, pages 8079–8088, 2024. 2, 3, 4
- [8] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 2
- [9] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023. 3, 4
- [10] Karlo Koledić, Igor Cvišić, Ivan Marković, and Ivan Petrović. Moft: Monocular odometry based on deep depth and careful feature selection and tracking. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6175–6181. IEEE, 2023. 3, 4, 5
- [11] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2
- [12] 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. 3, 4, 5
- [13] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 2, 3
- [14] Wan-Duo Kurt Ma, John P Lewis, and W Bastiaan Kleijn. Trailblazer: Trajectory control for diffusion-based video generation. *arXiv preprint arXiv:2401.00896*, 2023. 2, 3, 4, 5
- [15] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, pages 728–755. Springer, 2022. 2
- [16] 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, 2
- [17] Haonan Qiu, Zhaoxi Chen, Zhouxia Wang, Yingqing He, Menghan Xia, and Ziwei Liu. Freetraj: Tuning-free trajectory control in video diffusion models. *arXiv preprint arXiv:2406.16863*, 2024. 3, 4, 5
- [18] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, 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. 2
- [19] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1
- [20] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. To-



(a) Input

(b) TrailBlazer

(c) ours

Figure A7. Multi-object results.

wards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 2

- [21] Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6232–6242, 2024. 1
- [22] 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. *arXiv preprint arXiv:2406.17758*, 2024. 3, 4
- [23] Tianxing Wu, Chenyang Si, Yuming Jiang, Ziqi Huang, and Ziwei Liu. Freeinit: Bridging initialization gap in video diffusion models. *arXiv preprint arXiv:2312.07537*, 2023. 1
- [24] 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. 4
- [25] Mingyang Yi, Aoxue Li, Yi Xin, and Zhenguo Li. Towards understanding the working mechanism of text-to-image diffusion model. *arXiv preprint arXiv:2405.15330*, 2024. 1
- [26] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jia-Wei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. In *European Conference on Computer Vision*, pages 273–290. Springer, 2024. 3, 4, 5