Photorealistic Video Generation

Ayush Sharma
Boston University

Boyang Liu
Boston University

Chunhao Bi
Boston University

Abstract

Large text-to-image diffusion models have exhibited impressive proficiency in generating high-quality images. However, when applying these models to video domain, ensuring temporal consistency across video frames remains a formidable challenge. Despite of this, plenty of works proposed multiple solutions to mitigate the temporal inconsistency, and better results keep emerging. In this project, we satisfied the task of photorealistic video generation by modifying and improving a SOTA zero-shot text-guided video-to-video framework called Rerender A Video[17]. We changed the previous framework mainly in the key frame generation part, by introducing new sampling methods and changing key frame selection methods, while keeping the second part of full video translation by propagating frames between key frames. Visual results and experiments are carried out to demonstrate pros and cons of our modification. Code is available at: https://github.com/ChunhBi/Rerender_A_Video

1. Introduction

With recent advancements and breakthroughs in diffusion models [4, 10, 11], significant strides have been made in the field of text-to-image synthesis, particularly by training diffusion models on multimodal datasets. Despite these achievements, the transition to text-to-video generation poses considerable challenges, primarily due to difficulties in producing coherent and consistent video sequences. The primary challenges stem from the inherent complexities in effectively controlling text-to-image generation and the extraction of temporal information between consecutive frames.

In this study, our focus is centered on "Rerender A Video" [17]. This work distinguishes itself through three main contributions: it introduces a novel zero-shot framework that eliminates the need for extensive training; it employs hierarchical cross-frame consistency to meticulously control the connections between consecutive frames; and it implements a hybrid approach combining diffusion-based generation with patch-based propagation, effectively balancing quality with computational efficiency.

However, we identify several limitations within this framework. Firstly, the uniform sampling approach for key frames, followed by translation and interpolation for non-key frames, may not be optimal. Uniform sampling could lead to resource inefficiency in low-variation segments or inadequately capture variations in rapidly changing scenes. To address this, we propose the incorporation of a frame-to-frame change evaluation for key frame selection, encompassing latent space changes, optical flow, and orb feature variations. The efficacy of this non-uniform key frame selection approach warrants further exploration.

Secondly, the current model utilizes only the first frame as a reference anchor to maintain global style consistency. This approach might not suffice for videos with significant stylistic evolution, particularly over extended durations. We are exploring adaptive anchor frame selection methods to enhance style consistency in such scenarios.

Thirdly, the inference process in "Rerender A Video" employs DDIM [13], which, while effective, may not be as efficient as newer methodologies. We are investigating the integration of alternative approaches, such as DPM-solver [8] or LCM [9], to expedite the key frame generation process.

Lastly, our experiments have revealed that the original examples presented in "Rerender A Video" tend to simplify the complexities involved in handling abrupt motions and color transitions, leading to artifacts. We hypothesize that these issues could be alleviated through our proposed non-uniform key frame selection and improved anchor frame strategies. Additional experimentation is required to validate these hypotheses and refine the model's performance in these complex scenarios.

2. Related Works

2.1. Text-to-Image

In the text-to-image domain, significant advancements have been made to control the resulting output images. [7] in their work "Controllable Text-to-Image Generation" intro-

duced conditional generative adversarial networks (cGANs) for customizable image synthesis from textual descriptions. [19] extended this approach in "Adding Conditional Control to Text-to-Image Diffusion Models," allowing for precise image generation through added conditional controls in diffusion models. [4] proposed Denoising Diffusion Probabilistic Models (DDPMs), a generative model that incrementally denoises data, creating high-quality images. [11] in "High-Resolution Image Synthesis with Latent Diffusion Models" focus on efficient, high-resolution image generation using Latent Diffusion Models (LDMs). Furthermore, [13] in "Denoising Diffusion Implicit Models" introduce an approach for faster sampling with fewer timesteps in diffusion-based models.

2.2. Video-to-Video (Video Synthesis)

The video-to-video synthesis field has seen various innovative approaches. Some works directly generated successive key frames, ensuring the coherence during the simultaneous generation process. Initially, most works used a video as reference to generate videos. [14] in "Video-to-Video Synthesis" develop a GAN-based framework for converting semantic label maps into realistic videos. It incorporates a spatial-temporal adversarial objective, facilitating the production of high-quality videos. These methods usually need extensive text-video data for training. To deal with this problem, "ReRender a Video" [17] uses a zero-shot method to reduce fine-tuning burden, while other methods like [3] directly model the relation between frames, without knowing about the image. In "MAGViT", [18] explore the use of Masked Generative Video Transformers for video generation, applying masked image modeling principles. Additionally, [6] contribute the Kinetics Human Action Video Dataset, a substantial resource for machine learning models in video understanding.

2.3. Text-to-video

The text-to-video sector is rapidly evolving. Many works attempted to generate videos directly from prompts with few or without input videos. The advent of large language models (LLM) also catalyzed the development in the field of prompt engineering. [15] in "Tune-A-Video" present a one-shot tuning method for adapting image diffusion models to text-to-video generation. [12] in "Make-A-Video" proposed spatial-temporal modules for T2I models, improving video generation resolution, frame rate, and text fidelity, setting a new standard in text-to-video generation. Recent work "Emu Video" [2] uses factorized method to generate videos conditioned on first frame, which distinctly retains visual quality and diversity than previous work. Also, some works attempted to utilize transfer learning techniques. [5] in "Free-bloom" propose a zero-shot text-to-video generator using a Large Language Model to generate coherent prompts and a Latent Diffusion Model to achieve temporal coherence. However, T2V work are usually restricted to short videos shorter than 5 seconds. There are still difficulties in generating longer videos, as the action may go out of control without a reference action, which is provided in V2V tasks.

3. Method and Dataset

In this work, we aim to integrate text with input video as a reference for video-to-video generation. This section details the methodologies and datasets we have employed to realize this objective.

3.1. Dataset

We collected the data for video generation from some of these datasets:

- **MSR-VTT[16]**: A large-scale dataset for the open domain video captioning, which consists of 10,000 video clips from 20 categories, and each video clip is annotated with 20 English sentences. There are about 29,000 unique words in all captions. The standard splits uses 6,513 clips for training, 497 clips for validation, and 2,990 clips for testing.
- **Kinetics[6]**: A large-scale, high-quality dataset for human action recognition in videos. The dataset consists of around 500,000 video clips covering 600 human action classes with at least 600 video clips for each action class. Each video clip lasts around 10 seconds and is labeled with a single action class.
- **Web Vid[1]**: Contains 10 million video clips with captions, sourced from the web. The videos are diverse and rich in their content.
- Other web resources like Bilibili, Google, Baidu...

3.2. Method

We modified the original work mainly in key frame translation and key frame selection. The purpose is to improve the quality of images generated by diffusion model, while keeping the features from the original video.

3.2.1 Inference using DPM-Solver

Original work of Rerender is built highly based on DDIM[13], which is a classic inference method that is widely used in many diffusion-model based framework. However, as more and more improved methods emerged, DDIM gradually became less dominant. A new method called DPM-Solver[8] was proposed after DDIM, with its advantage of faster speed and higher quality. For this reason, we decided to use DPM-Solver for inference.

We substituted the first part of DDIM-based key frame sampling with vanilla DPM-Solver(as the cross-attention part from the original work is high coupled with DDIM, we

find it hard to merge it into DPM-Solver). Therefore, the key frame translation from DPM-Solver will solely image-to-image result without context consistency. The whole pipeline of the original work is showed in Figure 1.

Although frame attention is not added to DPM-Solver, we find that DPM-Solver generally generates higher quality images in single image translation than the modified DDIM in the original work, for similar calculation resources. Details are shown in Figure 2 and Figure 3 in results.

3.2.2 Non-uniform Key Frame Selection

Non-uniform key frame selection will help solve the problems caused by uniform key frame selection like waste of resources in the cases of less variation and lack of variations' representation in the cases of drastic changes. In this work, we first completed the sub-task of comparing methods for measuring the degree of changes between frames. Then we utilized the best of them to select key frames and compared the result with the baseline: uniform key frame selection.

Firstly, we compared three approaches to determining the extent of change between consecutive frames for the purpose of selecting key frames:

- **ORB Feature Points Matching:** We detected and computed the Oriented FAST and Rotated BRIEF (ORB) feature points in every images and used a brute-force matcher to calculate the relation between two images. This method is valid and efficient.
- **Optical Flow Change Calculation:** We employed the method of optical flow analysis to assess the alterations between adjacent frames. This technique enables us to quantify the motion and flow of objects within the video sequence.
- **ResNet-Based Feature Extraction:** We harnessed image feature representation models, such as ResNet, to extract frame features. Subsequently, we computed the norm differences between these features to gauge the dissimilarity between frames.

Secondly, we implemented a simple algorithm 1 to select key frames based on the extent of changes between frames in the video. With the input frames $\mathbf{F} \in \mathbb{R}^{n \times h \times w \times c}$, we calculate the difference matrix $\mathbf{dF} \in \mathbb{R}^{n-1 \times 1}$ based on the best approach to calculating differences. Then we calculate the threshold t which is used to determine whether a frame should be chosen as a key frame with $\mathbf{dF}$ and a hyper-parameter interval. Then we traverse the frames and determine a frame as a key frame if the accumulated difference in an interval of frames exceeds the threshold or the interval between key frames is sufficiently long. To balance the scenario where key frames might be selected prematurely, leading to shorter intervals in some cases, we introduce a hyper-parameter α which is slightly greater than 1 to increase the original interval. This adjustment helps to manage the number of key frames and offset the early selection effect.

Algorithm 1 Key Frame Selection

$dF \leftarrow \text{CALCULATEDIFFERENCES}(F)$
$t \leftarrow \text{CALCULATETHRESHOLD}(dF, interval)$
$sumDiff \leftarrow 0$
$lastIndex \leftarrow 0$
Select frame 0 as key frame
for $i = 1$ **to** $\text{LENGTH}(F)$ **do**
    $sumDiff \leftarrow sumDiff + dF[i]$
    if $sumDiff > threshold$ **or**
    $i - lastIndex \geq \alpha \cdot interval$ **then**
        Select frame i as key frame
        $sumDiff \leftarrow 0$
        $lastIndex \leftarrow i$
    end if
end for

In this work, we calculate the threshold t by multiply the average differences and the input interval and use a hyper parameter β to control the threshold. The parameter β is employed to determine the minimum value of the interval in regions of significant change. Since different methods will be sensitive to the extent of changes differently, we need a parameter to balance the differences. Temporally, we set β as 1 for all methods.

$$t = \beta \times interval \times \text{MEAN}(dF) \tag{1}$$

This threshold configuration might not be optimal for various methods. We will try to better define the hyper parameter, or try to combine the threshold with the algorithm itself in the future.

4. Evaluation Criteria

Since video generative models are mostly evaluated by human beings themselves, we aim to evaluate the generated videos based on various different metrics like:

4.1. Qualitative

Measures how well the generated video resembles the input text and:

- **Visual Quality**: Evaluation based on the visuals of the video generated.
- **Realism**: Evaluation based on how close the video is to reality.
- **Diversity**: Diversity will measure whether the generated samples cover the full variability of the real samples.
- **User Study**: We should also gather feedback from users to test the effectiveness of the generative video model.

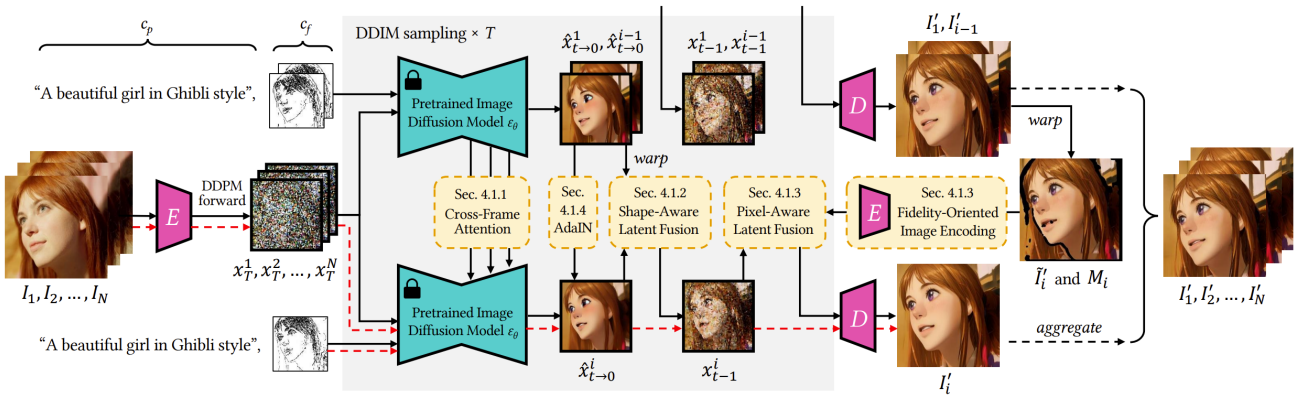


Figure 1. Rerender A Video pipeline. We substituted the DDIM part with DPM-Solver. Adapted from [17].

It also includes **Textual Faithfulness** (judging the video based on its representation of textual description).

- **Coherence**: Coherence will measure the continuity and smoothness of the generated samples. The generated video should be coherent.

4.2. Quantitative

CLIP score will measure the compatibility of frame-caption pairs. It will evaluate the similarity correlation between the input prompts and the visual contents of generated frames. Higher CLIP scores mean higher compatibility. CLIP score is found to have high correlation with human judgement. The paper, Rerender A Video[17], mentions the use of **FateZero** and **Pix2Video** to report **Fram-Acc** (CLIP-based frame-wise editing accuracy), **Tmp-Con** (CLIP-based cosine similarity between consecutive frames), **Pixel-MSE** (averaged mean squared pixel error between aligned consecutive frames). Here we use the **average CLIP score** of the whole generated video, which is calculated over each key frame. Although this is lack of evaluation for frame-to-frame coherence, this is better for evaluating the accuracy for single generated key frame.

5. Results

We tested the modifications on the MSR-VTT[16] dataset and observed the following outcomes for respective aspects.

5.1. Computing Resource Analysis

Our project requires NVIDIA graphics cards with a minimum of 8GB memory. However, in high-resolution video scenarios, 8GB GPUs may struggle with peak requirements. Generally, generating key frames for a 10-second video at 840x512 pixels resolution on an NVIDIA 4070-laptop 8GB GPU took approximately 1 hour, translating key frames selected uniformly every 10 frames for 13-15 seconds. In contrast, for the same task, using an NVIDIA 4060Ti-desktop 16GB GPU took about 30 minutes, but the key frame translation process still required approximately 12-15 seconds. This difference in efficiency is mainly due to the limitations of 8GB GPUs during intensive tasks, such as the final step of sampling. When overwhelmed, these GPUs may switch to the integrated graphics processor, significantly slowing down processing speeds.

5.2. Inference Sampling Methods

We have compared the performance between original DDIM work, with DPM-Solver version. For time analysis, when using DPM-solver, the time is slightly decreased from 13-15 seconds to 10-12 seconds for same set up, compared to original DDIM. However, as the algorithm consumes a lot of time in other parts, the total time cost isn't reduced much. We mainly use visual evaluation for qualitative evaluation, and average CLIP score for quantitative evaluation. Figure 2 shows that under same conditions (same setup and same prompt), images generated by DPM-Solver are more detailed and high-quality compared to DDIM. The edges are sharper, and stripes are more clear and organized. This can be seen in table 1. Figure 3 shows that the attention mechanisms in the original work, which are built on DDIM, is making a great difference in controlling consistency between frames. The appearance of the character is consistent during the whole video, which is sampled based on previous key frame features. However, because of lack of frame-to-frame attention, DPM-Solver generated video results are facing significant change of portraits. The faces from the characters differs greatly when motions are taken between key frames.

Average CLIP score	Ink	Painting	Sculpture
DDIM	26.62	27.68	**27.75**
DPM-solver	**29.57**	**31.17**	24.96

Table 1. Average CLIP score for two sampling methods. Bigger value means the image is more similar to its description.

Figure 2. DDIM vs DPM-solver single image comparison. Images on the left are results by DDIM, images on the right are results by DPM-solver. As shown, DPM-generated image is more detailed and high-quality compared to DDIM.

5.3. Key Frame Selection

In previous work, key frames were uniformly sampled. We noticed that the system cannot precisely grab the features of frames when the video changes drastically. In this work, we solved this problem to a certain extent, by firstly calculating the degree of changes between consecutive frames and subsequently selecting key frames based on the the changes.

We compared three approaches: ORB-based, optical-flow based and ResNet-based methods. Since we do not have groundtruth of the degree of changes, we determined them based on our observations. We chose two types of representative videos to test: one involves videos with relatively moderate variations, while the other includes videos with significant changes, such as during scene transitions. We compared our observations on videos with the key frame results, here are our conclusions.

Figure 3. DDIM vs DPM-solver video comparison. The left are results by DDIM, the right is DPM-solver. It is obvious that DPM-solver is lack of consistency between key frames, as no attention mechanism is implemented compared to DDIM.

For moderate videos, all three methods yield average results in frame selection. In complex scenarios, the ORB feature points matching method fails to effectively detect changes, resulting in average selections. Conversely, both the optical flow method and ResNet feature extraction method can identify more significant changes. More specifically, the optical flow method is very sensitive to changes, tending to generate very short intervals during great changes compared to the ResNet-based method. The ResNet-based method is relatively gentle, but it takes four times more time than the optical flow method. We tested both the optical flow method and ResNet-based method.

We use a bowing and opening speech scenario to verify the effectiveness of our algorithm. During the bowing process, the character will move up and down quickly, while during the speech process, there will only be subtle changes in the face. The original method fails to reconstruct the bowing process both stably and accurately, producing some overlapping phantoms. As shown in Figure 4, the left part is the uniformly selected key frames, the 0021 frame fails to be generated properly. In contrast, our approach yields smoother outputs. Additionally, our generated video maintains consistent quality in the smoother sections, even with fewer key frames.

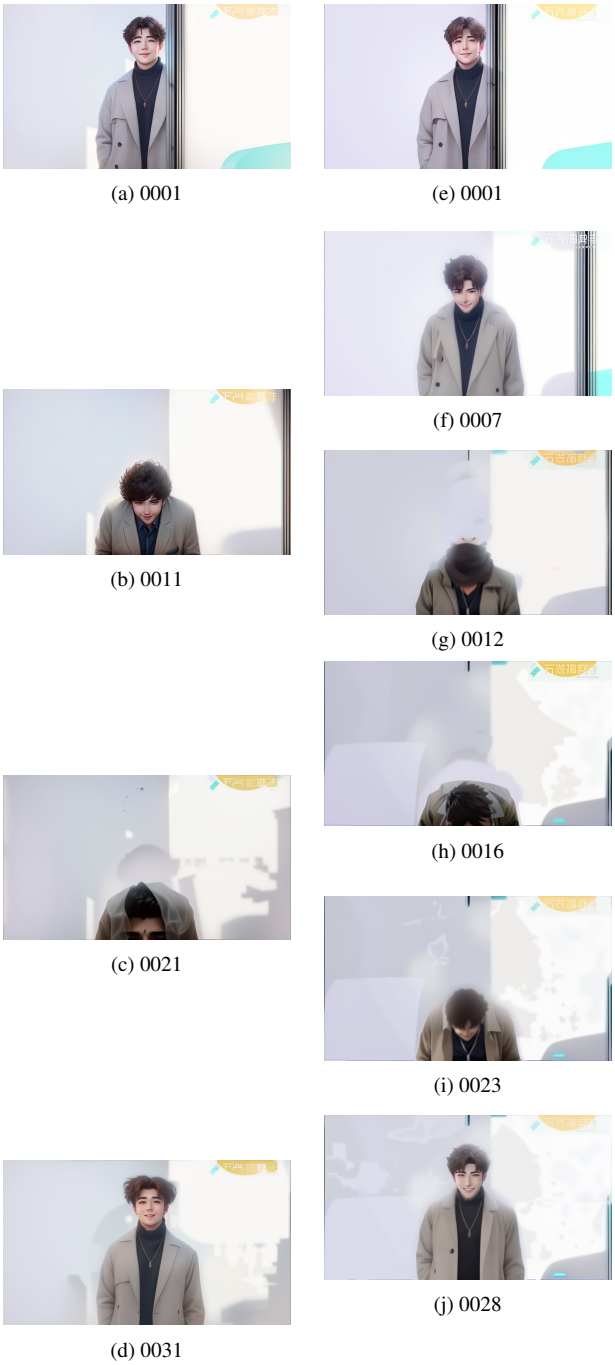
6. Conclusion

In this work, we implemented photorealistic video generation, particularly in the context of deep learning and diffusion models. We modified the state-of-the-art zero-shot text-guided video-to-video framework, introducing innovative approaches in key frame generation and selection. The utilization of DPM-Solver and implementation of non-uniform key frame selection techniques not only enhanced the quality of video generation but also addressed the challenges of temporal consistency and computational efficiency to some extent.

In our key frame selection experiments, we observed that in videos with moderate variations, all three methods provided average results. However, in more complex videos, especially with significant scene transitions, the Optical Flow and ResNet methods outperformed the ORB method in accurately capturing frame features. Our tests with different video types, such as a bowing and opening speech scenario, further confirmed the effectiveness of our non-uniform key frame selection algorithm in producing smoother video outputs and maintaining consistent quality even with fewer key frames.

The experimental findings indicate modest enhancements in the visual quality of the generated videos. However, it's crucial to recognize the existing constraints and the prospects for additional improvements. For instance, the initial approach utilized a single ControlNet; exploring the simultaneous use of multiple ControlNets could be beneficial. Integrating openpose, hed, and tile ControlNets might allow for more nuanced image control while maintaining detail, as opposed to relying solely on canny or hed for preserving image contours. Moreover, the attention mechanisms within the DPM-Solver component have not been fully explored, presenting an opportunity for development. The intricate decoupling scenarios presented in the source code require a deep understanding of complex methodologies. In future work, we aim to more extensively employ advanced techniques to further refine our approach.

Figure 4. Comparison of key frame selection methods: uniformly selected frames (left) vs. ResNet-based selection (right). The label of images denotes the specific moment in time at which the frame of the image occurs."



References

[1] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval, 2022. 2

[2] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning, 2023. 2

[3] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning, 2023. 2

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1, 2

[5] Hanzhuo Huang, Yufan Feng, Cheng Shi, Lan Xu, Jingyi Yu, and Sibei Yang. Free-bloom: Zero-shot text-to-video generator with llm director and ldm animator, 2023. 2

[6] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017. 2

[7] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H. S. Torr. Controllable text-to-image generation, 2019. 1

[8] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps, 2022. 1, 2

[9] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2023. 1

[10] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1

[11] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1, 2

[12] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data, 2022. 2

[13] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 1, 2

[14] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis, 2018. 2

[15] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation, 2023. 2

[16] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 4

[17] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation, 2023. 1, 2, 4

[18] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G. Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, and Lu Jiang. Magvit: Masked generative video transformer, 2023. 2

[19] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2