Fairy: Fast Parallelized Instruction-Guided Video-to-Video Synthesis

Bichen Wu Ching-Yao Chuang Xiaoyan Wang Yichen Jia
Kapil Krishnakumar Tong Xiao Feng Liang Licheng Yu Peter Vajda
GenAI, Meta

Project page: https://fairy-video2video.github.io

Figure 1. **Fairy for Instruction-Guided Video Editing.** Given **a video** and an **instruction** for editing, Fairy performs accurate edits while ensuring temporal coherence. Remarkably efficient, 120 frames of 512×384 video can be generated in just **14 seconds**. We refer readers to our supplementary material to check the results in video format.

Abstract

In this paper, we introduce Fairy, a minimalist yet robust adaptation of image-editing diffusion models, enhancing them for video editing applications. Our approach centers on the concept of anchor-based cross-frame attention, a mechanism that implicitly propagates diffusion features across frames, ensuring superior temporal coherence and high-fidelity synthesis. Fairy not only addresses limitations of previous models on memory and processing speed, but also improves temporal consistency through a unique data augmentation strategy. This strategy renders the model equivariant to affine transformations in both source and target images. Remarkably efficient, Fairy generates 120-frame 512×384 videos (4-second duration at 30 FPS) in just 14 seconds, outpacing prior works by at least 44×. A comprehensive user study, involving 1000 generated samples, confirms that our approach delivers superior quality, decisively outperforming established methods.

1. Introduction

The advent of generative artificial intelligence has heralded a new era of creative potential, characterized by the ability to create or modify content in an effortless manner. In particular, image editing has undergone a significant evolution, driven by text-to-image diffusion models pretrained on billion-scale datasets. This surge has catalyzed a vast array

of applications in image editing and content creation.

Building on the accomplishments of image-based models, the next natural frontier is to transition these capabilities to the temporal dimension to enable effortless and creative video editing. A direct strategy to leap from image to video is to simply process a video on a frame-by-frame basis using an image model. Nonetheless, generative image editing is inherently high-variance – there are countless ways to edit a given image based on the same text prompt. As a result, it is difficult to maintain temporal coherence if each frame is edited independently [30].

Previous and concurrent studies have proposed several ways to improve the temporal consistency, and one promising paradigm is what we call *tracking-and-propagation*: one first applies an image editing model on one or a few frames, then tracks pixels across all frames and propagates the edit to the entire video. Existing works [4, 6, 12, 14, 16, 19, 24, 33] track pixels mainly through optical flow or by reconstructing videos as some canonical layered representations. Despite some successful applications, this paradigm is not robust, since tracking is an unsolved computer vision challenge. Existing methods, including optical flow or layered video representation, often struggle with videos with large motion and complex dynamics.

In this work, we introduce *Fairy*, a versatile and efficient video-to-video synthesis framework that generates high-quality videos with remarkable speed (Figure 1). Our work re-examines the tracking-and-propagation paradigm under the context of diffusion model features. In particular, we bridge cross-frame attention with correspondence estimation, showing that it temporally tracks and propagates intermediate features inside a diffusion model. The cross-frame attention map can be interpreted as a similarity metric assessing the correspondence between tokens throughout various frames, where features from one semantic region will assign higher attention to similar semantic regions in other frames, as shown in Figure 3. Consequently, the current feature representations are refined and propagated through a weighted sum of similar regions across frames via attention, effectively minimizing feature disparity between frames, which translates to improved temporal consistency.

The analysis gives rise to our *anchor-based model*, the central component of Fairy. To ensure temporal consistency, we sample K anchor frames from which we extract diffusion features, and the extracted features define a set global features to be propagated to successive frames. When generating each new frame, we replace the self-attention layer with cross-frame attention with respect to the cached features of anchor frames. With cross-frame attention, the tokens in each frame take the features in anchor frames that exhibit analogous semantic content, thereby enhancing consistency. In addition, by sampling K anchor frames instead of computing cross-attention with respect to all frames, Fairy achieves several advantages: (1) it ensures temporal consistency by sharing the same global features, (2) it overcomes the memory issue due to extensive frame number, (3) it enhances processing speed through the caching of anchor frame features, and (4) it streamlines parallel computation, thereby facilitating remarkably rapid generation on multiple GPUs.

Despite the improvement from anchor-based cross-frame attention, the model is still sensitive to minor variations within the input frames, even with the same text prompt and initial latent noise. Such small changes could stem from natural movements within a video sequence or from affine transformations applied to the input. The gold standard solution is to train the model with pairs of original and edited videos, thereby accommodate it to recognize and adapt to such variations. However, collecting such a dataset is far from straightforward. To emulate these transformations, we employ a data augmentation strategy. Starting with an input image and its edited counterpart, we apply a sequence of affine transformations to both, generating successive frames. The assumption is that the affine transformations applied to input images should correspondingly affect the edited images. This method of *equivariant finetuning* leads to notable enhancements in temporal consistency.

To verify the effectiveness of Fairy, we conducted a large-scale evaluation consists of 1000 generated videos. Both human evaluation and quantitative metrics confirm that our model achieves significantly better quality comparing to existing works. Moreover, thanks to the simplicity of the design and the parallelizable architecture, Fairy achieves $>$44x speedup over baselines.

In short, this work makes the following contributions: (1) We adopt a series of simple yet effective adaptions that transform an image-editing model for video-to-video synthesis. (2) We evaluate our approach via extensive human study with 1000 generated videos, confirming that Fairy delivers superior quality over prior state-of-the art methods. (3) Fairy is blazing fast, achieving $>$44x speedup over previous methods when utilizing 8-GPU parallelized generation.

2. Related works

Conditional video generation: Following the success of diffusion models in text-to-image generation [7, 22, 23, 25], there has been a surge in video generation. Based on a text-to-video model, video-to-video generation can be achieved by conditioning the model on attributes extracted from a source video. For example, Gen-1 [9] conditions on the estimated depth while VideoComposer [31] integrates additional cues, such as depth, motion vectors, sketches, among others. Building such models requires training on video datasets, which are much more scarce than image datasets [26]. Training such models also imposes considerable com-

putational demands. Consequently, these constraints confine video models to reduced resolution, shorter duration, and smaller model size, leading to a decline in visual quality when contrasted with contemporary image generation models. In comparison, our model is adapted from a pretrained image-to-image model. Our finetuning only requires image data, and the training cost (30 hours on 8 A100 GPUs) is substantially smaller than video models.

Tracking and propagation: this paradigm involves initiating edits on a single image, identifying pixel correspondences across the video sequence, then propagating the edit. The key in this approach lies in tracking. Numerous efforts [12, 24, 33] have adopted optical flow, keypoint tracking, or other motion cues to tackle the tracking problem. Another stream of efforts [4, 6, 14, 16, 19] reconstruct the video using a multi-layer canonical representation, associating pixels to canonical points on the representation. However, video tracking is an unsolved computer vision challenge and often fails on complex videos. Additionally, tracking-and-propagation does not allow editing of object contours, which breaks the pixel correspondence. Instead of tracking in pixel space, our model leverages cross-frame attention to implicitly track corresponding regions and propagate features to reduce frame discrepancy. Owing to the robustness and versatility of diffusion features, as also observed in Tang et al. [27], our approach accommodates a broader spectrum of videos and offers enhanced editing flexibility.

Image model adaptation: Many works also adapt image-to-image models to video. For example, [15] modifies self-attention in diffusion models. [32] performs per-video finetuning and utilizes a inversion-denoising procedure for editing. [10, 17, 21, 29] adapt image-to-image pipelines [5, 11, 28] to edit videos, by modifying/adding cross-frame attention modules, null-text inversion, etc. Most of these methods can only generate video clips with a small number of frames, while [10] leverages a nearest-neighbor field on diffusion features to propagate key frame features to more frames. Our model improves the design of spatial temporal attention [15, 17, 21, 29] to anchor-based cross-frame attention, which enables generating long videos with arbitrarily many frames. We further improves its temporal consistency by equivariant finetuning. Our work bears resemblance to the concurrent work [10]. To edit a video, [10] first performs a latent inversion on the original video, extract a nearest-neighbor field, which is then used for feature propagation to generate the target video. Our pipeline is much simpler and more efficient. We do not require latent inversion; and the feature propagation is achieved through attention; our architectures naturally allows parallel generation. As a result, our model is 53 times faster than [10].

3. Preliminaries

Video-to-Video Diffusion Models In this work, we primarily focus on instruction-guided video editing. Given an input video with N frames $\boldsymbol{I} = \{I^1, \ldots, I^N\} \in \mathcal{I}^N$, the goal is to edit it into a new video $\boldsymbol{I'} = \{I^{1'}, \ldots, I^{N'}\} \in \mathcal{I}^N$ according to an natural language instruction $c \in \mathcal{T}$ that preserves the semantic of the original video. A straightforward baseline is to adopt an image-based editting model $f : (\mathcal{I}, \mathcal{T}) \to \mathcal{I}$ to edit the video frame by frame: $\boldsymbol{I'} = \{f(I^1, c), \ldots, f(I^N, c)\}$. In this work, we build upon this line of work and improve the consistency with a variant of cross-frame attention.

Self-attention and Cross-frame attention Self-attention has played a crucial role in the diffusion networks. In a self-attention block, features of tokens are projected into queries $\boldsymbol{Q} \in \mathbb{R}^{n \times d}$, keys $\boldsymbol{K} \in \mathbb{R}^{n \times d}$, and values $\boldsymbol{V} \in \mathbb{R}^{n \times d}$, where the output is defined as

$$\text{SelfAttention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}\right)\boldsymbol{V}.$$

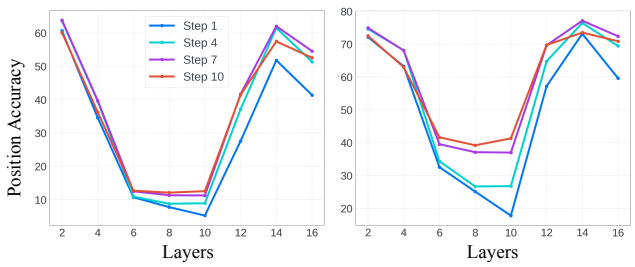
The output from the softmax is commonly referred to as the *attention score* or *attention map*. Given N frames, to extend the self-attention to cross-frame attention, one can simply concat the keys and values from all frames, e.g., $\boldsymbol{K}^* = [\boldsymbol{K}^1, \cdots, \boldsymbol{K}^N]$, and compute the self-attention as Self-Attention$(\boldsymbol{Q}, \boldsymbol{K}^*, \boldsymbol{V}^*)$. In particular, cross-frame attention provides temporal modeling capability by attending to other frames and shows encouraging results in improving temporal consistency [17, 30].

4. Implicit Tracking via Cross-frame Attention

We first bridge cross-frame attention with correspondence estimation, fostering a straightforward yet effective feature propagation mechanism for video-to-video generation.

The primary objective of self-attention is to select appropriate values $\boldsymbol{V}$ with the attention scores determined by $\boldsymbol{Q}\boldsymbol{K}^T$. In the case of cross-frame attention, given a token location p in a frame, the attention score is computed by the cosine similarity between $\boldsymbol{Q}_{p,:}$ and each token in $\boldsymbol{K}^*$, where the key values $\boldsymbol{V}^*$ are the features of tokens across both spatial and temporal dimension.

It is noteworthy that the mathematical formulation exhibits profound similarities to feature propagation mechanisms. Specifically, the attention score serves as the estimated correspondence, and the output of attention module could be interpreted as a fused representation of warped features derived from successive frames. We will empirically substantiate this hypothesis through analyses of the tracking behavior inherent in the attention score.



(a) $\delta^x = 16$ (b) $\delta^x = 32$

Figure 2. **Position Accuracy δ^x on DAVIS.** The cross-frame attention score demonstrates significant tracking proficiency, particularly evident in the initial and final stages of the UNet.

4.1. Temporal Tracking with Attention Score

In this section, we provide evidences that the attention scores in cross-frame attention implicitly serve as a correspondence estimation across frames. In particular, we adopt a conditional image-to-image diffusion model and examine the attention map between two frames of a video clip. Consider $\boldsymbol{Q}^t$ and $\boldsymbol{K}^t$ as the respective query and key representations corresponding to the frame at time t. To corroborate our conjecture regarding the role of attention scores, we designate a specific query point p at time t and endeavor to ascertain its corresponding coordinate q at a subsequent time t' through the expression:

$$q = \arg\max_{p'} \boldsymbol{A}_{p,p'}, \text{ where } \boldsymbol{A} = \text{softmax}(\frac{\boldsymbol{Q}^t \boldsymbol{K}^{t'T}}{\sqrt{d}}),$$

where $\boldsymbol{A}_{p,p'}$ denotes the element of the matrix $\boldsymbol{A}$ located at the row index p and column index p'. The correspondence is estimated by selecting the location p' with the highest attention score with respect to p. For multi-head attention, we average the attention scores from all heads. By evaluating the tracking ability of the proposed estimator, we can verify whether the attention scores are good correspondence estimator for feature propagation.

4.2. Video Tracking Experiments: TAP-Vid

In our evaluation, we utilize the DAVIS datasets from the TAP-Vid [8, 20], with 30 videos clips ranging from 34-104 frames. The frames are resize to 256×256 for evaluation. We measure the $< \delta^x$ position accuracy proposed in TAP-Vid, which calculates the fraction of points that are within δ^x pixels of their ground truth position. The dimensions of the attention map inherently impose a constraint on the precision achievable in point tracking. Since diffusion UNets adopts spatial downsampling, we configure δ^x at the values of 16 and 32 for our experiments. We set the number of diffusion step to 10 with Euler ancestral sampler [13].

Figure 2 shows the position accuracy for attention scores across different layers and diffusion step. We can see that the first and last few layers demonstrate a strong tracking results, achieving over $60\%/70\%$ accuracy for $\delta^x = 16/32$. Interestingly, the results are consistent across different diffusion step, demonstrating the strong tracking ability of cross-frame attention. The observed degradation in accuracy at the central layers of the UNet architecture can be attributed primarily to the reduction in the spatial resolution of the feature maps. For instance, within the seventh layer of the network, the feature map dimensions are constrained to 4×4. Figure 3 visualizes the attention score on a target frame given a query point. We can see that the attention map locate the corresponding position in target frame.

Figure 3. **Visualization of Attention Score.** The left image shows the query point p within the current frame, and the right image is the target frame. Cross-frame attention performs accurate temporal correspondence estimation without any finetuning.

Cross-frame Attention $\approx$ Tracking and Feature Propagation Our experimental findings disclose an unexpectedly potent tracking capability associated with the attention score. These results robustly validate our hypothesis: *even in the absence of explicit finetuning, cross-frame attention implicitly executes a formidable feature propagation mechanism*. In particular, features $\boldsymbol{V}^*$ from alternative frames are transmitted to the current frame based on the correspondence determined through the attention scores.

5. Fairy: Fast Video-to-Video Synthesis

Building on the analyses, we present Fairy, an efficacious video-to-video framework that leverages the inherent feature propagation of cross-frame attention. In particular, we propose to propagate the value features from a collection of anchor frames to a candidate frame using cross-frame attention. The performance can be further enhanced through the proposed equivariant finetuning method. We also demonstrate that Fairy is easily parallelizable, facilitating fast generation of arbitrarily long videos.

5.1. Anchor-Based Model

Inspired by prior research in tracking-and-propagation, where the edits to one or a few frames are propagated to the entire video, we sample a set of *anchor frames* and edit them with an image-based model $f : (\mathcal{I}, \mathcal{T}) \rightarrow \mathcal{I}$. Similarly, our objective is to extend the edits in the anchor frames to the successive frames, but utilizing cross-frame attention mechanisms instead of optical flow or ex-

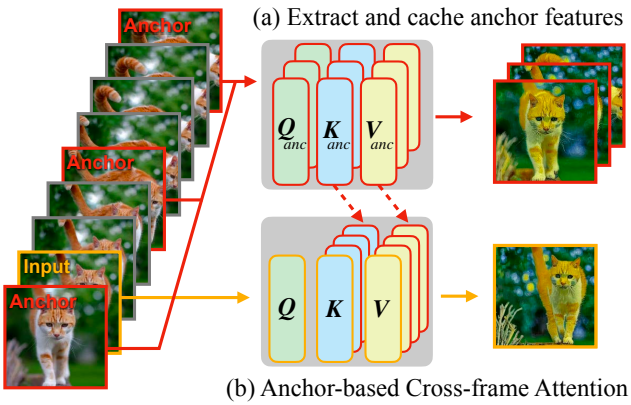


Figure 4. **Illustration of Attention Blocks** (a) Given a set of anchor frames, we extract and cache the attention feature $\boldsymbol{K}_{anc}$ and $\boldsymbol{V}_{anc}$. (b) Given an input frame, we perform cross-frame attention with respect to the cached features of anchor frames.

plicit point tracking. In particular, given a set of anchor frames $\boldsymbol{I}_{anc} = \{\tilde{I}^1, \ldots, \tilde{I}^K\} \subseteq \boldsymbol{I} = \{I^1, \ldots, I^N\}$, we treat them as a batch and feed them to the diffusion model f, where the self-attention in the model is replaced with cross-frame attention in a zero-shot manner. Throughout the diffusion process, for each anchor frame $\tilde{I}^n$, we store its key and value vectors $\boldsymbol{K}^{n,l,t}, \boldsymbol{V}^{n,l,t}$ for every cross-frame attention layer l and every diffusion step t in cache. Intuitively, $\boldsymbol{V}^{n,l,t}$ defines a set global features to be propagated to successive frames. To simplify the notation, we will drop the subscript l and t in the following sections.

Let $\boldsymbol{K}_{anc} = [\boldsymbol{K}^1, \ldots, \boldsymbol{K}^K]$ and $\boldsymbol{V}_{anc} = [\boldsymbol{V}^1, \ldots, \boldsymbol{V}^K]$ be the concatenated anchor key and value vectors. To edit any frame $I^t \in \boldsymbol{I}$, we modify the self-attention module to the cross-frame attention with respect to the key and value vectors of anchor frames as follows:

$$\text{softmax}\left(\frac{\boldsymbol{Q}[\boldsymbol{K}, \boldsymbol{K}_{anc}]^T}{\sqrt{d}}\right)[\boldsymbol{V}, \boldsymbol{V}_{anc}],$$

where $\boldsymbol{Q}, \boldsymbol{K}$ and $\boldsymbol{V}$ are the self-attention vectors of the input frame I^t. The idea is that the attention score generated by the softmax facilitates cross-frame tracking by estimating the temporal correspondence between the input frame and anchor frames. The global value vectors then be propagated to input frame by multiplying the attention score with $\boldsymbol{V}_{anc}$. By substituting the self-attention module with an anchor-based cross-frame attention mechanism, we found that the model could generate highly consistent video edits. In the default setting, we choose anchor frames uniformly across the video, and we did not notice consistent performance improvement or degradation when adopting different anchor-frame selection strategies.

Fast Generation via Parallelization Note that editing frame I_t does not require other frames as input except the cached features $\boldsymbol{K}_{anc}$ and $\boldsymbol{V}_{anc}$ from anchor frames. Therefore, we can edit arbitrary long videos by splitting them into segments and leverage multi-GPUs to parallize the generation, while the computation remains numerically identical. As a result, our method achieves significant speedup compared to previous works. Moreover, it delivers superior quality outputs without succumbing to memory-related constraints. This efficiency underscores our approach's enhanced scalability and practicality, setting a new benchmark for performance in the realm of video editing.

5.2. Equivariant Finetuning

While anchor-based attention greatly improves the quality, we still occasionally observed temporal inconsistency. In particular, we found that for generated contents that do not have semantic correspondence with the input, small changes in input frames can cause significant variances in the output frames.

To improve the consistency, we leverage the following intuition to design a data augmentation strategy. In particular, if an input frame I^t differs from I^{t-1} only in the camera position, then the output frame $\hat{I}^t$ and $\hat{I}^{t-1}$ should only be different in the camera position as well. This inspires us to come up with a data augmented strategy that can be applied to any image editing dataset to imporve the temporal consistency. Given a pair of images, the original and the edited, denoted as (I, I'), we randomly sample an affine transformation $g : \mathcal{I} \to \mathcal{I}$ and apply them to both images to obtain $(g(I), g(I'))$. We implement this using torchvision's random affine transformation [2], setting random rotations degrees to $< 5°$, random translation to $[-0.05, 0.05]$, random scaling factor to $[0.95, 1.05]$, and random shear degrees to $[-5°, 5°]$ on both axis. We also apply random resized crop, scaling the original image to 288pix, and randomly crop a square image with 256 pix. We then fine-tuned the base image-to-image model to generate the transformed $g(I')$ given the transformed $g(I)$. The proposed fine-tuning process makes the model equivariant to affine transformations, leading us to denote our approach as *equivariant finetuning*. Empirically, we observe a notable enhancements in temporal consistency after finetuning (Section 6.3).

6. Results

We implement Fairy based on an instruction-based image editing model, similar to [5], and replace the model's self-attention with cross-frame attention. We set the number of anchor frames to 3. Anchor frames are uniformly selected with equal intervals among all frames. The model can accept input with different aspect ratios, and we rescale the input resolution with the longer size to be 512, and keep the aspect ratio unchanged. We edit all frames of the input video, without temporal downsampling. We distribute the

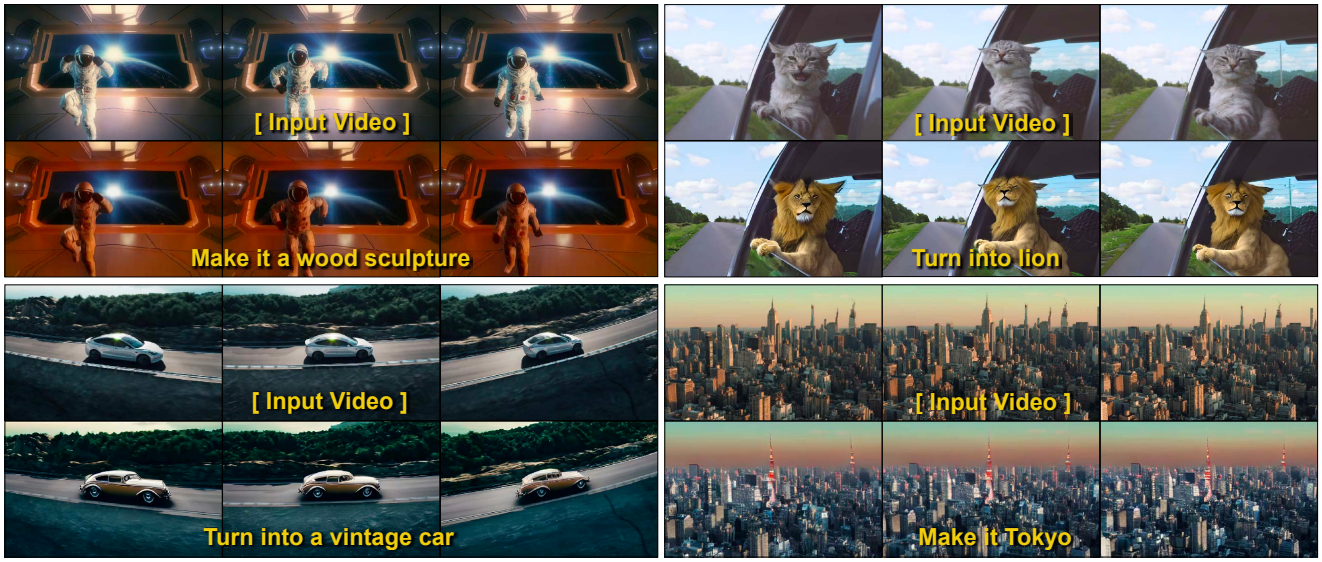


Figure 5. **Diverse Video Editing via Fairy.** Fairy enables a wide range of video edits with different types of subjects.

computation to 8 A100 GPUs. We use the Euler Ancestral sampler with 10 diffusion steps.

For equivariant finetuning, we use the same dataset that was used to train the image editing model, and apply the data augmentation discussed in Section 5.2. We load the image editing model's pretrained checkpoint, and resumed training for 50,000 steps with a batch size of 128, costing 30 hours on 8 A100 GPUs with 80GB memories.

6.1. Qualitative Evaluation

We first show qualitative results of Fairy. Since most of the PDF readers do not render videos properly, we only show a small number of frames for each video. We strongly recommend readers to checkout our supplementary materials to watch the complete videos. In Figure 5, we show that our model is capable conducting edits on different subjects. In Figure 6, we show that our model is able to conduct different types of editing, including stylization, character swap, local editing, and attribute editing, following textual instructions. In Figure 9, we show that our model can transform the source character into different target characters based on instructions. Note that our model can adapt to different input aspect ratios without need for re-training. Our input videos contain large motions, occlusions, and other complex dynamics. Despite those challenges, videos generated by our model are temporally consistent and visually appealing. We also show our model's capabilities to generate long videos in the supplementary material.

6.2. Quantitative Evaluation

Quantitatively evaluating video generative models is challenging. First, the generation task is intrinsically high-variance – there are countless ways to edit an video given the instruction. Second, previous works have adopted metrics such as CLIP scores [9, 10] to evaluate the generation quality. However, these metrics are not necessarily aligned with human perception [18]. Lastly, human evaluation is still the golden standard to judge the quality. Yet, due to the cost of human evaluation, previous works have only conducted small scale human evaluations (< 100 samples).

In this paper, we conduct a large-scale user study on an evaluation set consists of 1000 video-instruction samples. The evaluation set is divided into two parts: first, to test a model's robustness across different videos, we construct the evaluation set of 50 videos $\times$ 10 instructions. And to test a model's robustness across different instructions, we construct a dual evaluation set of 10 videos $\times$ 50 instructions. The videos are accessible from ShutterStock [3]. To our best knowledge, this is the largest evaluation in the video-to-video generation literature so far.

We conduct a A/B comparison to compare our method with three previous works, Rerender [33] (tracking and propagation), Tokenflow [2] (image model adaptation), and Gen-1 [9] (conditional video model), which are the strongest representative of the three paradigms for video-to-video generation today. Results from baselines are collected from [1]. Prompts for baselines are descriptive, e.g., "a dog running on grass, in Van Gogh style". We re-write the prompt for our method as an edit instruction, e.g., "in Van Gogh Style". Since Gen-1 is not open-sourced, the evaluation is done on a smaller evaluation set of 100 videos. In each evaluation tuple, we show the input video, the editing instruction or prompt, and the output videos generated by Fairy and a baseline. We ask human evaluators to choose the better video in terms of their single frame quality, temporal consistency, prompt faithfulness, input faithfulness, and overall quality. Each comparison is rated by 3 different

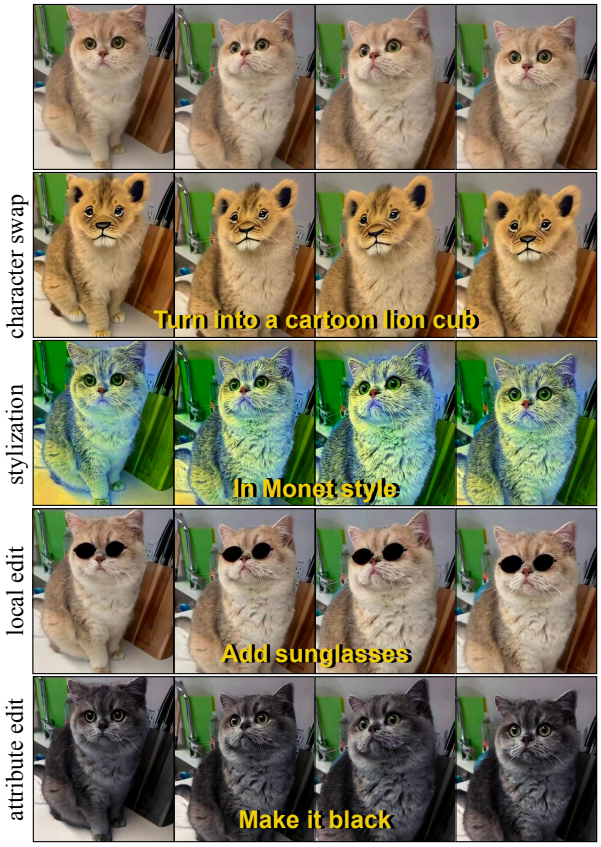
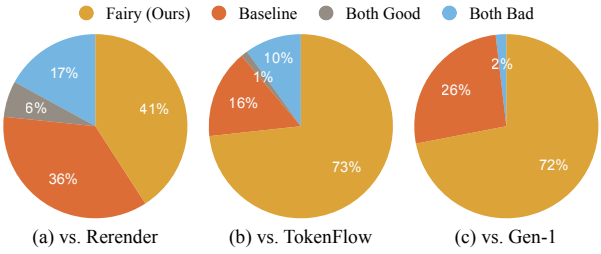


Figure 6. **Different type of editing.** Fairy is able to hand a diverse set of instructions and perform appropriate editing.



(a) vs. Rerender (b) vs. TokenFlow (c) vs. Gen-1

Figure 7. **A/B Comparison with Baselines.** Fairy significantly surpassed baseline models, demonstrating its effectivity.

	Latency (sec) ↓	Frame-Acc ↑	Tem-Con ↑
TokenFlow	744	0.537	0.973
Rerender	608	0.775	0.972
Ours	**13.8**	**0.819**	**0.974**

Table 1. We assess our method's temporal consistency and fidelity to the target text prompt using CLIP similarity metrics.

annotators and the decision is determined by the majority vote. We report the overall quality comparison in Figure 7, which demonstrates that videos produced by Fairy are more preferable, with a win rate of 41% vs 36% against Rerender, 73% vs. 16% against TokenFlow, and 72% vs 26% against Gen-1. More details in the supplementary material.

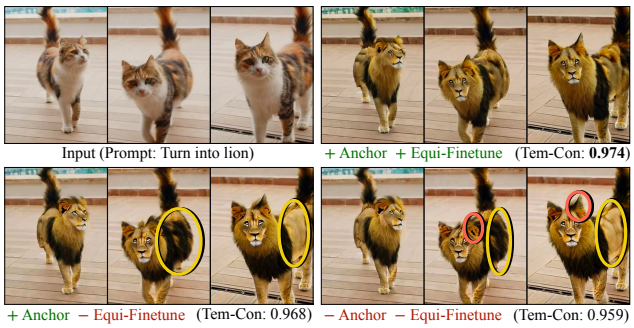


Figure 8. **Ablation Study.** Without equivariant fine-tuning and anchor-based attention, we observed inconsistencies, particularly in the changing patterns of body and earscostumes over time. This inconsistency is further exacerbated upon the removal of anchor-based attention, leading to lower temporal consistency score.

Figure 10 shows the visual comparison with the baselines. We observe that both Tokenflow [10] and Rerender [33] do not adhere closely to the provided instructions, resulting in evident inconsistencies. Outputs from Gen-1 often over-modify the entire scene and do not retain the original content effectively. In contrast, Fairy meticulously follows the instruction, delivering high-quality, temporal consistent, and authentic generations.

Lastly, we compute metrics adopted by previous works, Tem-Con and Frame-Acc [21, 33]. Tem-Con assesses temporal consistency by calculating the cosine similarity of CLIP feature across successive frame pairs, and Frame-Acc measures the percentage of frames where the edited image exhibits greater CLIP similarity to the target prompt than to the source prompt. The results in Table 1 demonstrates that Fairy achieves the best temporal consistency and frame-wise editing accuracy against the baselines.

Speed Comparison In Table 1, we also compare the latency of different models. In particular, we calculate the inference time of editing a 4-seconds, 30 FPS, 512p×384p video on a server with 8 A100 GPUs. The key-frame interval of Rerender is set to 4 instead of the default 10, since the test videos contain faster motion. This leads to improved quality for Rerender. All other parameters were default. Due to its architecture simplicity, Fairy is already significantly faster than baselines using 1 GPU. Using a single GPU, Fairy completes inference in just 78 seconds, achieving 9.5× faster than TokenFlow and 7.5× faster than Rerender. When utilizing all 8 GPUs on the node, Fairy is 53× faster than TokenFlow and 44× faster than Rerender.

6.3. Ablation Studies

We conduct an ablation study to verify the effectiveness of our model's component. We gradually remove equivariant fine-tuning and anchor-based attention, ultimately leading to the adoption of a standard frame-by-frame editing approach. The results are shown in Figure 8. The model

Figure 9. **Diverse Character Swap:** Fairy possesses the capability to interchange the individual with a diverse array of characters.

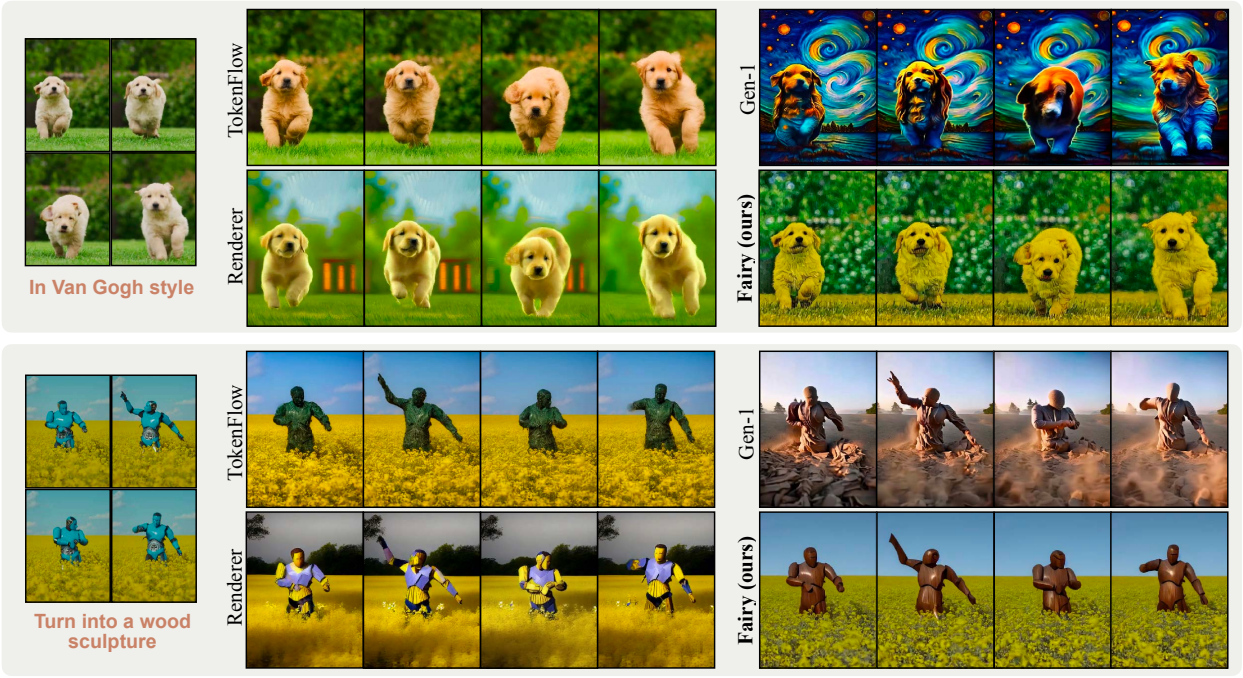


Figure 10. **Comparison with Baselines.** Fairy consistently outperform baselines in terms of consistency and instruction-faithfulness.

becomes sensitive to the camera motion without equivariant finetuning, rendering inconsistency in the details. The subsequent removal of anchor-based attention, transitioning to a frame-based model, introduces further inconsistencies in the generated video. We compute the Tem-Con metric based on 150 videos and report in Figure 8. It confirms our observation that the proposed methodology effectively improves the temporal consistency, lifting the Tem-Con from 0.959 (baseline) to 0.968 (w/ anchors) to 0.974 (w/ anchor and equivariant finetuning).

6.4. Limitations

The efficacy of Fairy is intrinsically tied to the underlying image-editing model. This means that any inherent constraints of this underlying model, e.g., face and text distortion, etc., will naturally manifest in the video editing capacities of Fairy . In our observations, a notable side effect of equivariant finetuning is the diminished ability to accurately render dynamic visual effects, such as lightning or flames. The process seems to overly focus on maintaining temporal consistency, which often results in the depiction of lightning as static or stagnate, rather than dynamic and fluid. See the supplementary material for visualization.

7. Conclusion

Fairy offers a transformative approach to video editing, building on the strengths of image-editing diffusion models. By leveraging anchor-based cross-frame attention and equivariant finetuning, Fairy guarantees temporal consistency and superior video synthesis. Moreover, it tackles the memory and processing speed constraints observed in preceding models. With the capability to produce high-resolution videos at a blazing speed, Fairy firmly establishes its superiority in terms of quality and efficiency.

References

[1] Exploring video-to-video synthesis: A comparative analysis of rerender, tokenflow, and gen-1. https://medium.com/@lwen9595/exploring-video-to-video-synthesis-a-comparative-analysis-of-rerender-tokenflow-and-gen-1-9a63f281c4e1. 6

[2] pytorch documentation. https://pytorch.org/vision/main/transforms.html. Accessed: 2023-11-02. 5, 6

[3] Stock footage video, royalty-free hd, 4k video clips, 2023. 6

[4] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *European conference on computer vision*, pages 707–723. Springer, 2022. 2, 3

[5] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 3, 5

[6] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stablevideo: Text-driven consistency-aware diffusion video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23040–23050, 2023. 2, 3

[7] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 2

[8] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adrià Recasens, Lucas Smaira, Yusuf Aytar, João Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 4

[9] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 2, 6

[10] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023. 3, 6, 7

[11] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[12] Ondřej Jamriška, Šárka Sochorová, Ondřej Texler, Michal Lukáč, Jakub Fišer, Jingwan Lu, Eli Shechtman, and Daniel Sỳkora. Stylizing video by example. *ACM Transactions on Graphics (TOG)*, 38(4):1–11, 2019. 2, 3

[13] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022. 4

[14] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM Transactions on Graphics (TOG)*, 40(6):1–12, 2021. 2, 3

[15] 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. *arXiv preprint arXiv:2303.13439*, 2023. 3

[16] Yao-Chih Lee, Ji-Ze Genevieve Jang, Yi-Ting Chen, Elizabeth Qiu, and Jia-Bin Huang. Shape-aware text-driven layered video editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14317–14326, 2023. 2, 3

[17] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv preprint arXiv:2303.04761*, 2023. 3

[18] Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. *arXiv preprint arXiv:2310.11440*, 2023. 6

[19] Hao Ouyang, Qiuyu Wang, Yuxi Xiao, Qingyan Bai, Juntao Zhang, Kecheng Zheng, Xiaowei Zhou, Qifeng Chen, and Yujun Shen. Codef: Content deformation fields for temporally consistent video processing. *arXiv preprint arXiv:2308.07926*, 2023. 2, 3

[20] 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. 4

[21] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. *arXiv:2303.09535*, 2023. 3, 7

[22] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[23] 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. 2

[24] Alexander S. Disco diffusion v5.2 - warp fusion. https://github.com/Sxela/DiscoDiffusion-Warp. 2, 3

[25] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 2

[26] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 2

[27] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence

from image diffusion. *arXiv preprint arXiv:2306.03881*, 2023. 3

[28] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 3

[29] Wen Wang, kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models. *arXiv preprint arXiv:2303.17599*, 2023. 3

[30] Wen Wang, Kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models. *arXiv preprint arXiv:2303.17599*, 2023. 2, 3

[31] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *arXiv preprint arXiv:2306.02018*, 2023. 2

[32] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023. 3

[33] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. *arXiv preprint arXiv:2306.07954*, 2023. 2, 3, 6, 7