FADE: Frequency-Aware Diffusion Model Factorization for Video Editing

Yixuan Zhu[1]   Haolin Wang[1]   Shilin Ma[2]   Wenliang Zhao[1]   Yansong Tang[2]   Lei Chen[1,†]   Jie Zhou[1]

[1]Department of Automation, Tsinghua University
[2]Tsinghua Shenzhen International Graduate School, Tsinghua University

Source Prompt: *A black swan swimming in a river beside bushes*

Appearance Edit: *black swan* → *white duck*
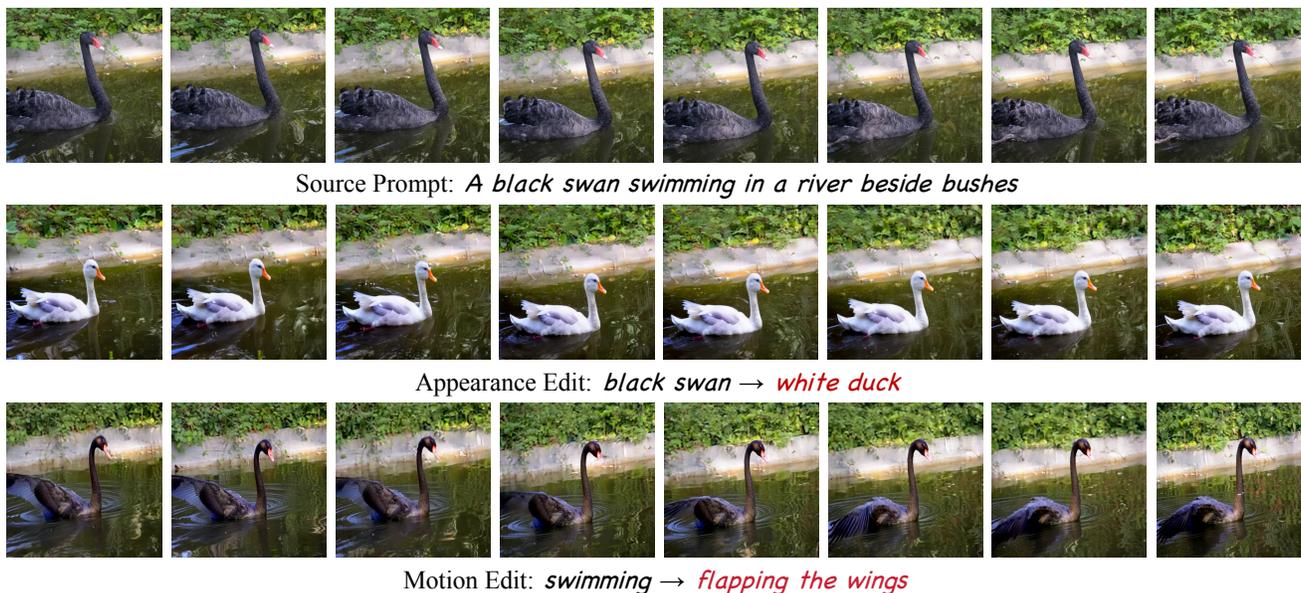
Motion Edit: *swimming* → *flapping the wings*

Figure 1. **Diverse video editing results of *FADE*.** Our training-free approach, utilizing frequency-aware factorization and modulation, achieves high-fidelity, coherent edits across a variety of video types. FADE handles both appearance and motion adjustments with impressive robustness, ensuring precise alignment with input prompts and maintaining temporal consistency. Best viewed in color.

Abstract

Recent advancements in diffusion frameworks have significantly enhanced video editing, achieving high fidelity and strong alignment with textual prompts. However, conventional approaches using image diffusion models fall short in handling video dynamics, particularly for challenging temporal edits like motion adjustments. While current video diffusion models produce high-quality results, adapting them for efficient editing remains difficult due to the heavy computational demands that prevent the direct application of previous image editing techniques. To overcome these limitations, we introduce FADE—a training-free yet highly effective video editing approach that fully leverages the inherent priors from pre-trained video diffusion models via frequency-aware factorization. Rather than simply using these models, we first analyze the atten-tion patterns within the video model to reveal how video priors are distributed across different components. Building on these insights, we propose a factorization strategy to optimize each component's specialized role. Furthermore, we devise spectrum-guided modulation to refine the sampling trajectory with frequency domain cues, preventing information leakage and supporting efficient, versatile edits while preserving the basic spatial and temporal structure. Extensive experiments on real-world videos demonstrate that our method consistently delivers high-quality, realistic and temporally coherent editing results both qualitatively and quantitatively. Code is available at https://github.com/EternalEvan/FADE.

1. Introduction

The objective of video editing is to modify specific visual content consistently across all frames while preserving the
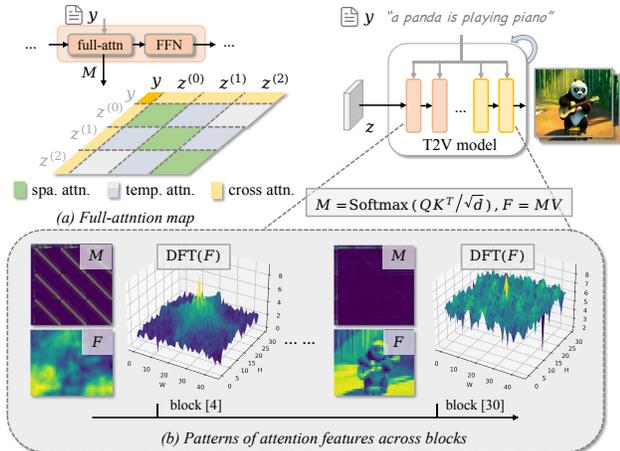
† Corresponding author

Figure 2. (a) In typical T2V models, video tokens flattened along the temporal axis yield full-attention maps split into cross, spatial, and temporal attention. (b) Video priors in T2V models are factorized based on attention feature patterns. We visualize the attention map M and result F along with its spectrum for two blocks. The early block sketches the foundational structure and movement, while the later block refines details. This insight allows us to leverage each block's unique role for more efficient editing.

integrity of unchanged regions. Recent achievements in diffusion models have greatly elevated video editing capabilities [10, 12, 15, 19, 20, 28, 32], bringing remarkable generative power to video editing tasks that require both spatial details and temporal coherence. While previous methods [1, 8, 13, 18, 30] have adapted text-to-image (T2I) diffusion models for prompt-driven image editing, their effectiveness diminishes when applied to video due to the complexities of managing spatial-temporal relationships across frames and a lack of generalized video prior knowledge. These limitations underscore the need for an efficient and robust editing strategy capable of handling the increased data and temporal coherence demands of videos. In response, we aim to design an accessible, high-quality editing solution that fully leverages the rich video priors embedded in pre-trained video diffusion models, enabling realistic, coherent, and versatile edits, including both appearance and motion adjustments, across complex video content.

The main challenge in video editing is balancing changes to the target object while preserving unedited content. Previous methods largely rely on the generative power of T2I models for controlled edits. For instance, [15, 33] achieve this through null-text inversion [18], which minimizes inversion errors via test-time optimization. However, this process requires extensive iterations to compute a series of null-text embeddings for every sampling step, making it time-consuming and limiting its real-world applicability. Another line of work is attention feature injection [5, 15, 23, 27], where attention maps are modified by

replacing or interleaving to maintain content consistency. Despite this, the absence of video priors often results in unrealistic temporal dynamics and limits motion edit capacity. Some methods, like [19], directly utilize video diffusion models by adding noise to the source video. This straightforward approach fails to fully exploit video priors, resulting in suboptimal editing quality. With the recent emergence of advanced text-to-video (T2V) models, new opportunities arise to harness their rich video knowledge. However, adapting previous T2I techniques to these models is challenging, given the extensive resources required, including memory, computation and time, to handle the increased data and complexity inherent in large-scale video networks.

To overcome these challenges, we introduce FADE, a training-free yet potent approach for real-world video editing. Our goal is to harness the rich video priors about video appearance and dynamics within pre-trained T2V models to enable realistic and diverse editing. To achieve this, we start by examining the specific properties of these models. While previous work [9] shows that diffusion models naturally emphasize different frequency components across sampling timesteps, we further observe that a similar frequency progression occurs block-wise within T2V models. We explore how video priors function in each block by analyzing their roles and patterns, revealing that early blocks sketch foundational spatial layouts and temporal dynamics while later blocks refine high-frequency details (Figure 2). Leveraging this insight, we factorize the editing problem by the distinct functions of each component, enhancing editing efficiency. Then, instead of performing optimization or attention injection, we design a spectrum-guided modulation strategy that utilizes the full-attention outputs in these sketching blocks to derive frequency-aware guidance through 3D Fourier transformation. This modulation adjusts the sampling trajectory to preserve low-frequency structure from the source video while allowing high-frequency details to support flexible and intricate edits. This cue from the frequency domain not only mitigates information leakage during guidance computation but also unleashes the potential of video diffusion priors, enabling sophisticated spatial and temporal edits with high fidelity and coherence.

We evaluate our method using extensive real-world video clips across various editing tasks. As shown in Figure 1, FADE exhibits proficiency in producing realistic and consistent edits, covering both appearance and motion modifications. Quantitative comparisons on video editing benchmark DAVIS [22] demonstrate our approach achieves superior textual alignment and temporal consistency, with 0.3561 CLIP score and 43.57 OSV. Moreover, we maintain competitive input fidelity with 21.54 Mask-PSNR and 0.3297 LPIPS, preserving unedited regions effectively. Compared to previous methods, our use of general video priors with frequency-aware factorization deliv-
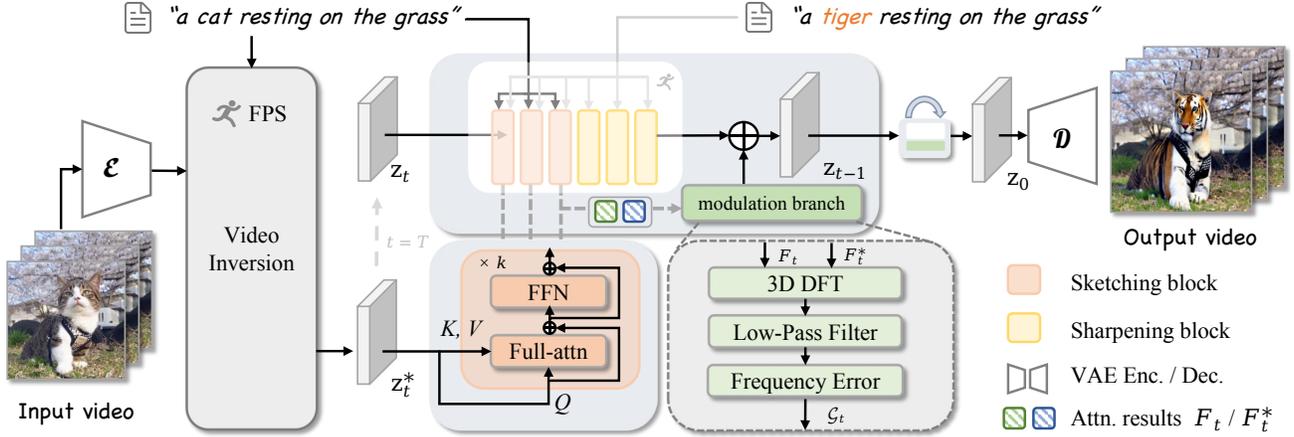
Figure 3. **The overall framework of *FADE*.** FADE is a training-free framework for video editing. Given an input video, we first perform video inversion to obtain noise z_T and trajectory $\{z_t^*\}_{t=0}^T$. Starting from this noise, we use sketching blocks in the T2V model to produce full-attention results F_t and F_t^* from z_t and z_t^*. These results are transformed to the frequency domain, where a low-pass filter isolates spatial and temporal structures. The frequency error then modulates the sampling process, enabling high-quality, consistent edits.

ers improved efficiency and a broader editing range, highlighting FADE's outstanding versatility and practicality.

2. Related Works

Video Diffusion Models. In recent years, denoising diffusion models [9, 29] have emerged as a leading class of generative models, notable for their synthesis quality and controllability. After training on large-scale datasets, text-to-image (T2I) diffusion models [6, 25] exhibit exceptional image synthesis capability. For video generation, some works [2, 11] adapt image diffusion models by inflating the denoising UNet [26] with temporal layers to handle video inputs. However, replicating the success of T2I models for video generation remains challenging due to the lack of large-scale video datasets and the increased data complexity. Most recently, Sora [4] and following studies [16, 35, 37] pioneer high-quality, high-resolution text-to-video (T2V) diffusion models. These models commonly use the diffusion transformer (DiT) architecture [21] to capture complex spatial and temporal coherence, outperforming previous UNet-based methods. However, this improved performance comes at the cost of high computational demands, mostly due to the extensive attention mechanism, which requires processing a large number of tokens. Limited research has explored the pattern within these models to reduce the computational intensity. To address this, we delve into their internal structure to analyze the roles of individual blocks, leveraging these insights to facilitate their applicability and efficiency in video editing.

Video Editing. Significant progress in image editing [3, 8, 17, 18, 20, 31] has driven advancements in video editing [7, 15, 19, 23, 34]. However, unique challenges in video editing arise from the need for consistent, coherent edits across frames, including both appearance and motion changes. Recent works [15, 23] adopt the attention feature injection techniques, manipulating various attention results to guide edits. They also employ model inflation to tune a T2I model with the input video following [34]. These approaches, however, are inefficient since they require a separate model for each sample and lack the general video priors to handle motion edits. In contrast, [19] utilizes a pre-trained video diffusion model for editing, beginning with corrupting the video by downsampling and noise. However, this corruption can result in global inconsistency with the source video and limits control over specific edits. Overall, there remains significant potential for advancing the use of video diffusion models in video editing. This application of the video diffusion model is nontrivial because (1) it requires maintaining global consistency in the generated video without relying on one-shot tuning or null-text optimization [18], and (2) applying previous attention feature injection methods is difficult with modern video models, as they involve dozens of transformer blocks with memory-intensive, full-attention computations. Our frequency-aware approach effectively addresses these challenges, setting our method apart from previous work.

3. Method

In this section, we present FADE, a training-free framework that achieves real-world video editing with frequency-aware factorization of diffusion models. Our key idea is to investigate the inherent patterns in a pre-trained text-to-video diffusion model and activate each block's specific function for various video editing tasks. We will start by reviewing the background of video diffusion models, and then describe our designs of FADE, including how to factorize video dif-

fusion models according to frequency cues and adjust the sampling trajectory via spectrum-guided modulation. The overall framework of our FADE is illustrated in Figure 3.

3.1. Preliminaries: Video Diffusion Models

Video diffusion models are a class of generative models that reconstruct the distribution of video data by modeling the reverse trajectory of a diffusion process. Given a random noise $\epsilon \sim \mathcal{N}(0, \boldsymbol{I})$, the diffusion process is defined as:

$$\boldsymbol{z}_t = \alpha_t \boldsymbol{z}_0 + \sigma_t \epsilon, \tag{1}$$

where $\boldsymbol{z}_0$ and $\boldsymbol{z}_t$ are the clean and noisy video, and $\{(\alpha_t, \sigma_t)\}_{t=1}^{T}$ denotes the noise schedule. With proper re-parameterization, the training objective of diffusion models can be expressed as:

$$\mathcal{L}_{\text{DM}} = \mathbb{E}_{\boldsymbol{z}_0, \epsilon, t} \left[\|\epsilon - \epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{y})\|_2^2 \right], \tag{2}$$

where $\boldsymbol{z}_t$ is computed as Equation (1). The denoising model ϵ_θ is learned to predict the noise given the prompt $\boldsymbol{y}$. Previous T2I models often implement ϵ_θ as UNet [26]. For video generation, recent models [4, 16, 35, 37] usually employ a series of cascaded transformer blocks to capture the spatial-temporal relationships more effectively.

To generate a video in a small number of denoising steps, diffusion models usually use deterministic DDIM sampling:

$$\boldsymbol{z}_{t-1} = \frac{\alpha_{t-1}}{\alpha_t} \boldsymbol{z}_t + (\sigma_{t-1} - \frac{\alpha_{t-1}}{\alpha_t} \sigma_t) \epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{y}). \tag{3}$$

Conversely, to convert a video to its corresponding noise, we can apply DDIM inversion, the reverse of Equation (3):

$$\boldsymbol{z}_{t+1} = \frac{\alpha_{t+1}}{\alpha_t} \boldsymbol{z}_t + (\sigma_{t+1} - \frac{\alpha_{t+1}}{\alpha_t} \sigma_t) \epsilon_\theta(\boldsymbol{z}_t, t, \boldsymbol{y}). \tag{4}$$

In practice, DDIM inversion introduces slight errors in every step, making it unreliable for fully reconstructing a real-world video or preserving fine details in the source video.

3.2. Frequency-Aware Factorization of T2V Model

Modern T2V models typically employ cascades of transformer blocks as the denoising model, with each block containing attention layers that process extensive video tokens. Given that dozens of identical blocks are involved in calculating the same form of attention, a key question arises: *Do all these blocks serve the same purpose and contribute equally to video synthesis?* To explore this, we analyze the behavior within each block of a pre-trained T2V model during video generation. Previous work [9] shows that diffusion models naturally emphasize different frequency components across sampling timesteps. Inspired by this, we adopt a frequency-aware perspective to uncover if similar trends occur at the block level in video models. Specifically, we capture and visualize each block's attention maps

$\{\boldsymbol{M}_i\}_{i=1}^{N}$ and outputs $\{\boldsymbol{F}_i\}_{i=1}^{N}$, along with their Fourier spectra, over all N blocks during sampling. As illustrated in Figure 2, a typical attention map in T2V models contains three types of attention regions: cross-attention for multimodal alignments, spatial attention for capturing intra-frame spatial structures, and temporal attention for tracking dynamics across frames. Our observation is that spatial and temporal attention are densely aligned along diagonal lines in the early blocks, creating multiple parallel diagonals in the attention maps. The main diagonal line suggests that these early blocks focus on outlining the intra-frame spatial structures, defining key shapes and boundaries of the composition. The secondary diagonal lines indicate that these blocks are simultaneously establishing temporal correspondences across frames within the same regions. Collectively, these early blocks construct the low-frequency, foundational structure of the video, setting up basic object placement and movement. We refer to them as *sketching blocks*. In contrast, the following blocks—which constitute the majority of the denoising model—display relatively sparse attention distribution, with scores more evenly spread across elements in the map. This implies that these later blocks focus on refining the high-frequency refinements, such as color, texture, and subtle vibration, providing nuanced adjustments for visual details. We call them *sharpening blocks*, as their primary role is to enhance details, enriching the visual appeal of the final output.

The variation in attention outputs of these blocks further supports our inference. The initial sketching blocks produce blurry outputs, with spectra focused on low frequencies that capture basic spatial and temporal layouts. In contrast, the sharpening blocks reveal clear foreground and background regions, involving much more high-frequency details. This distinction suggests that these two classes of blocks process features of differing levels and serve different purposes. We further verify that each block's function remains consistent throughout the sampling steps by analyzing these patterns across all steps. Thus, for video content editing—where maintaining underlying structures while adjusting finer details is essential—we can factorize each block's role, using sketching blocks for structural reconstruction and sharpening blocks for detailed generation. This division offers two key advantages: (1) it enables each block to perform optimally within its specialized function, and (2) it greatly reduces computational load by enabling frequency-aware guidance through selective use of blocks during sampling.

3.3. Spectrum-Guided Modulation

After factorizing the video model based on frequency-informed attention patterns, our next step is to activate each block's specific function and develop a robust sampling strategy that optimally balances reconstruction with generation. We begin by employing the inversion process de-

scribed in Equation (4) with a pre-trained video diffusion model ϵ_θ. This transforms the real-world video z_0^* to an approximated noise z_T^* and produces a reverse trajectory $\{z_t^*\}_{t=0}^T$. However, as discussed in prior works, DDIM inversion can introduce non-negligible errors, making it difficult to accurately reconstruct the source video from z_T^*. This issue is particularly pronounced in video inversion, where the predicted noise z_T^* often deviates far from the standard Gaussian distribution. We, therefore, focus on refining the sampling process using the observed spectrum patterns to correct the inversion error and create an ideal trajectory $\{z_t\}_{t=0}^T$ to a plausible edited video z_0. Following previous image editing approaches [18, 30], we adopt a dual branch strategy that leverages the inversion trajectory to enhance consistency with the source video. However, existing attention feature injection methods require extensive memory due to full-attention maps and limit editing flexibility through restrictive attention fusion techniques. While these methods can preserve the spatial layout, they often struggle with more complex edits involving pose, shape and motion.

To overcome these limitations, we propose a more flexible, spectrum-guided approach that factorizes visual information in the frequency domain to separate content for preservation and editing. This approach offers greater freedom for a wider range of edits. As shown in Figure 3, our first design is to harness the sketching blocks, which care about the overall spatial layout and temporal movement. Rather than swapping or interleaving the attention maps directly, we gather the attention outputs of the source video z_t^* and the target video z_t within these blocks, denoted as F_t^* and F_t, and use their difference to derive a guidance term for the sampling process. Specifically, these attention outputs are computed using the same source prompt y_{src} through the full-attention layers. This guidance term provides a more robust and adaptive way to adjust the sampling trajectory without directly altering the intermediate features within the diffusion model. Furthermore, since these features are derived from the sketching blocks, the guidance focuses on the coarse spatial and temporal content, such as basic layouts and object movements, leaving ample room for detailed edits to be made in the sharpening blocks.

However, native guidance can lead to information leakage from the source video to the target video, as the attention outputs contain the full range of visual content. To address this, it is crucial to determine which aspects of the content should be retained and which should be modified. For this purpose, we turn to the frequency domain for assistance. Real-world images and videos often exhibit redundancy across their spatial and temporal dimensions, meaning that most of their energy is concentrated in the low-frequency spectrum, while the high-frequency components contribute much less. This characteristic underpins popular compression algorithms such as JPEG for images and

MPEG for videos, and we aim to apply this principle here. Since we cannot get a clean video during the denoising process, we use attention outputs composed of tokens representing the noisy video. Unlike the discrete cosine transform (DCT) used in the spatial domain for MPEG, we employ the discrete Fourier transform on both spatial and temporal dimensions (3D DFT) to capture the spectral characteristics of intra-frame content and inter-frame motion:

$$\mathcal{F}_t = \mathrm{DFT}_{3\mathrm{D}}(F_t) \in \mathbb{R}^{h \times w \times \tau \times c}, \tag{5}$$

where h, w and τ are the spatial and temporal size of the video tokens, and c is the channel dimension. As we know, most of the energy in $\mathcal{F}_t$ is concentrated in the low-frequency range. Thus, with the observed attention patterns, we can assume that fundamental spatial structures and temporal movements are encapsulated within this range, while the high-frequency components contain Gaussian noise and specific details like texture and subtle shaking—elements we wish to suppress or remove during editing. To achieve this, we apply a low-pass filter to isolate the low-frequency content of $\mathcal{F}_t$. The same process is applied to $\mathcal{F}_t^*$ to compute the spectrum guidance $\mathcal{G}_t$:

$$\mathcal{G}_t = \|\mathrm{LP}(\mathcal{F}_t) - \mathrm{LP}(\mathcal{F}_t^*)\|_2^2, \tag{6}$$

where LP denotes the low-pass filter. This guidance measures the deviation of the target video z_t from the source video in terms of basic structure and movement. This distance is then used to modulate the trajectory during DDIM sampling by the gradient of $\mathcal{G}_t$ with respect to z_t:

$$z_{t-1} = \mathrm{DDIM}(\epsilon_\theta, z_t, t, y_{tgt}) - \lambda \mathrm{Norm}(\nabla_{z_t}\mathcal{G}_t), \tag{7}$$

where DDIM denotes the sampling process in Equation (3), λ is the guidance weight and Norm normalizes the gradient to avoid numerical instability. This modulation strategy iteratively denoises the edited video towards the ideal direction, ensuring high consistency and realistic edits.

3.4. Versatile Video Editing Framework

We utilize a pre-trained T2V diffusion model for video inversion and sampling. This model leverages a 3D VAE, comprising an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, to enable spatial and temporal compression, facilitating a reversible transformation between pixel space and latent space. In addition to text embeddings, the T2V model incorporates FPS information to better handle video data. Unlike one-shot tuned video models from T2I models [15], using a well-trained video diffusion model ensures a more stable understanding of the complex spatial and temporal relationships within videos. Moreover, this eliminates the need for model tuning, enhancing efficiency and reducing the necessity of storing individual models for specific videos. For edit types,
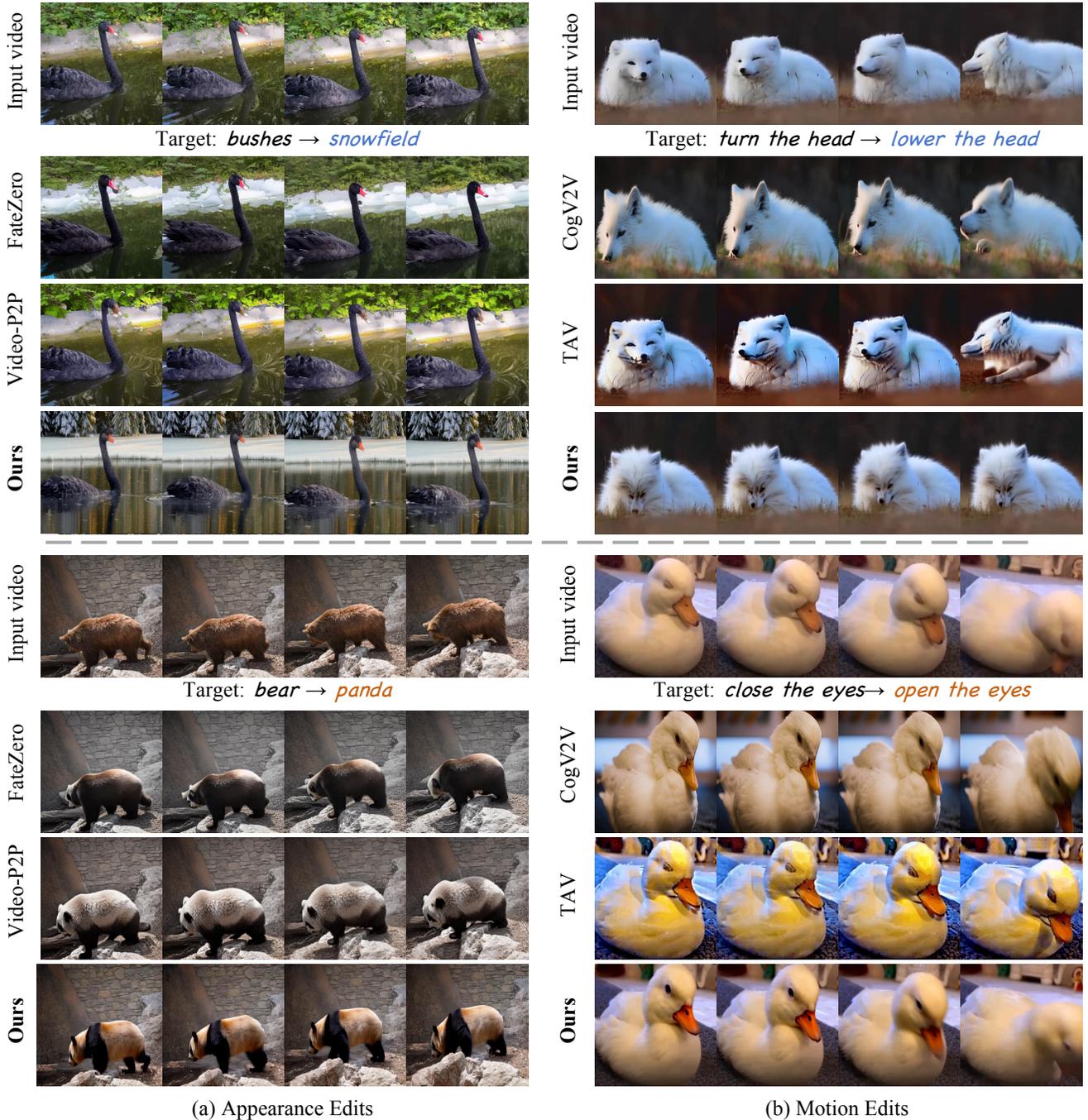
Figure 4. **Qualitative comparisons on real-world videos.** We evaluate both (a) appearance and (b) motion edits. Our framework consistently achieves accurate textual alignment across frames and high fidelity in unedited regions. Notably, compared to other methods based on video models, we produce more coherent and realistic motion edits. More results are available in the supplementary materials.

we broadly categorize prompt-driven modifications into appearance edits and motion edits. Appearance edits involve altering nouns, such as changing the main subject or background, focusing on the texture, color, and style (*e.g.*, transforming a "cat" to a "tiger" in Figure 3). Motion edits, on the other hand, involve changing verbs, which can alter spatial and temporal dynamics. By applying spectrum-guided modulation to harness inherent video priors within the T2V model, our method achieves realistic motion edits, distinguishing it from most previous approaches.

Table 1. **Quantitative comparisons on real-world video data.** We compare our FADE framework to previous methods on both appearance and motion edits, measuring editing accuracy, input fidelity, temporal consistency, and human preference. FADE achieves competitive performance across diverse editing tasks.

Method	CLIP ↑	M.PSNR ↑	LPIPS ↓	OSV ↓	PF ↑
appearance edits					
Tune-A-Video [34]	0.3522	19.86	0.4625	35.01	0.12
Video-P2P [15]	0.3589	20.57	0.3199	34.71	0.15
FateZero [23]	0.3562	20.65	**0.3057**	33.23	0.29
CogVideoX-V2V [35]	0.3754	18.96	0.4811	31.45	0.09
FADE (Ours)	**0.3762**	**20.69**	0.3085	**31.36**	**0.35**
motion edits					
Tune-A-Video [34]	0.3281	18.68	0.4637	35.85	0.10
Video-P2P [15]	0.3314	18.93	0.3885	35.44	0.15
FateZero [23]	0.3259	19.02	0.3712	34.47	0.13
CogVideoX-V2V [35]	0.3678	18.17	0.4928	35.52	0.19
FADE (Ours)	**0.3683**	**19.26**	**0.3692**	32.28	**0.43**

4. Experiments

To verify the effectiveness of our method, we conduct comprehensive experiments across various video editing tasks. We will start by describing the experimental settings and then present our main results. We will also provide detailed ablation studies and analyses of our approach to highlight the contributions of each component in our framework and to validate the impact of our design choices.

4.1. Experiment Setups

Implementation Details. To utilize video generation priors, we adopt a recent, widely used T2V model [35] for video inversion and generation. This model's denoising component employs the DiT architecture with 48 transformer blocks, each containing a full-attention layer. Based on observed attention map patterns, we designate the first 4 blocks as sketching blocks. We calculate the guidance over the interval $[0, 0.6T]$ of the DDIM sampling steps, with total steps set to $T = 50$. To generate textual prompts from the source video, we use a multimodal language model [14] as the captioner. The guidance weight varies between 10 and 15, tailored to each editing task. We perform all evaluations on a single NVIDIA LS20 GPU with 48GB VRAM.

Datasets and Metrics. We evaluate our method using the DAVIS dataset [22] as well as various in-the-wild videos, following the approach of previous studies [15, 23]. Alongside qualitative results, we provide quantitative evaluations across multiple aspects using different metrics. Editing accuracy is assessed through CLIP score [24], which reflects the degree of textual alignment. To evaluate temporal consistency, a key factor for video coherence, we compute object semantic variance (OSV), following [15], to capture semantic consistency across frames. Additionally, we assess fidelity in unedited regions by calculating Mask PSNR and LPIPS [36], which quantify the preservation of content that should remain unchanged. Furthermore, given the importance of visual quality in video editing, we conduct a user study to gauge human preference across all editing tasks.

4.2. Main Results

We compare our editing method with state-of-the-art approaches, including Video-P2P [15], FateZero [23], Tune-A-Video (TAV) [34] and CogVideoX-V2V (CogV2V) [35]. As illustrated in Figure 4, our method consistently yields superior results across all aspects when compared to previous methods. The editing results of Video-P2P and FateZero exhibit significant limitations, often failing to modify content accurately, resulting in noticeable textual mismatches. Similarly, while TAV and CogV2V leverage T2V models, they struggle with motion editing, producing subpar results in capturing dynamic changes. Leveraging the inherent priors from the T2V model and spectrum-guided modulation, our method achieves the best performance in preserving local structure, maintaining temporal consistency, and ensuring text-video alignment. We evaluate our method on 20 videos and measure 5 metrics for quantitative analysis. As shown in Table 1, our method achieves the highest CLIP and OSV scores, demonstrating its superiority in textual alignment and temporal consistency. We also obtain competitive Mask PSNR and LPIPS, indicating that our method preserves unedited regions effectively. Moreover, we achieve the highest human preference score, reflecting the favorable visual quality and appeal of our edits across tasks. Notably, our method operates without any optimization or fine-tuning, completing various edits on real-world videos in approximately 3 minutes, a significant efficiency gain compared to previous methods, which require over 15 minutes.

4.3. Analysis

Efficient Editing via Sketching Blocks. We assess the function of different blocks in the T2V model by testing various block combinations to calculate modulation, holding other settings constant. We consider three configurations: (1) using sketching blocks only (ours), (2) symmetrically combining sketching and sharpening blocks (symm. blocks), and (3) using all blocks without factorization (w/o factor.). As shown in Table 2, configuration (1) provides comparable results to configurations (2) and (3), despite requiring fewer blocks for modulation. Although configurations (2) and (3) offer slightly improved fidelity, they significantly increase test time and memory demands. Figure 5 further demonstrates that relying on only the first four transformer blocks, or sketching blocks, achieves the best editing performance. Interestingly, contrary to common assumptions, including the sharpening blocks actually diminishes editing quality. We assume this is because the sharpening blocks primarily focus on elements like color, texture, and subtle vibrations, which often do not need

| Input video | w/o factor. | w/o filter | **Ours** |
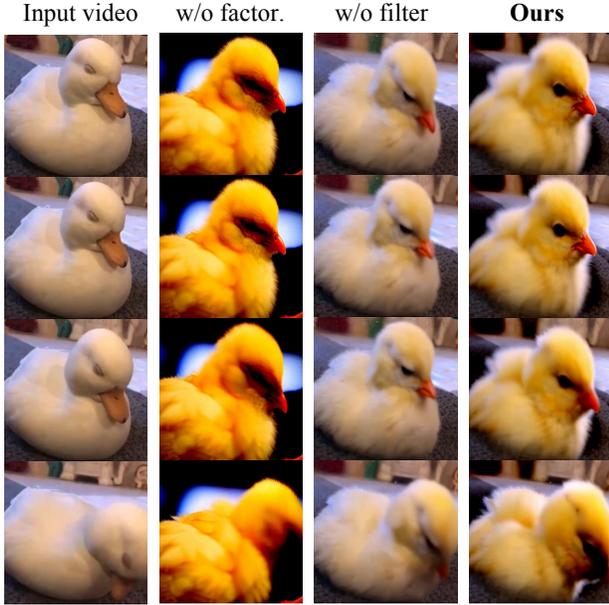
Target: *duck* → *yellow chick* + *open eyes*

Figure 5. **Qualitative comparisons of the ablations.** We find the variant frameworks fall short in terms of editing accuracy, temporal consistency and input fidelity, yielding unsatisfactory results.

to remain unchanged in most video editing tasks. Relying on these blocks for guidance may, therefore, misdirect the model, diverting its attention from building the foundational layout and movement of the input video, ultimately resulting in reduced performance. These results underscore the effectiveness of our frequency-aware factorization approach, demonstrating that our selective focus on sketching blocks provides an optimal balance between maintaining high-quality edits and achieving computational efficiency.

Impact of Spectrum-Guided Modulation. In Figure 5 and Table 2, we examine our spectrum-guided modulation by removing the low-pass filter (w/o filter) and analyzing its effect on the editing performance. When we use the entire spectrum of attention outputs, high-frequency components are retained in the guidance term, causing the target object to retain too much of the source's original characteristics, which weakens the text-video alignment. Experiments show that using about two-thirds of the frequency components in the guidance term balances reconstruction and editing quality. The quantitative results further underscore the efficacy of spectrum-guided modulation, as shown by superior performance on key metrics like textual alignment and temporal consistency. By selectively modulating the frequency components, our method achieves precise edits that align well with the given textual prompt and maintain consistency across frames, both of which are crucial for high-quality video editing. These findings highlight that spectrum-guided modulation not only supports a stable structural foundation for the edits but also enhances flexibil-

Table 2. **Ablations studies.** We perform ablations to verify the effectiveness of the components in FADE and the impact of the model factorization. We find that FADE achieves similar performance with only a subset of sketching blocks, compared to using symmetric or all blocks. Moreover, spectrum-guided modulation further enhances both preservation and precision in edits.

Method	CLIP ↑	M.PSNR ↑	LPIPS ↓	OSV ↓	Test-time
symm. blocks	0.3659	20.73	0.3367	32.61	5 min
w/o factor.	0.3691	**20.94**	**0.3328**	32.05	12 min
w/o filter	0.3612	20.89	0.3364	32.28	3 min
FADE (Ours)	**0.3728**	20.87	0.3352	**31.77**	3 min

ity, allowing detailed adjustments for various editing tasks while ensuring input fidelity across the video.

Effective Exploitation of Video Diffusion Model. Our framework capitalizes on the rich video priors embedded in the pre-trained T2V model, enabling more effective utilization of the model's inherent knowledge of spatial layouts, temporal coherence, and object motions. The video diffusion model captures nuanced understanding across frames, making it an ideal foundation for edits that require both high fidelity and temporal consistency. To validate our superior exploitation of these priors, we compare FADE with CogVideoX-V2V [35], which uses the same underlying T2V model. As shown in Figure 4 and Table 1, FADE outperforms it both qualitatively and quantitatively across various editing tasks, demonstrating our framework's ability to harness the model's rich knowledge more effectively.

Limitations. Despite FADE's effectiveness, its performance relies on the underlying video model and struggles with heavy occlusions requiring advanced temporal reasoning. We will address these in future work.

5. Conclusion

In this paper, we present FADE, a novel, training-free framework tailored for a diverse range of video editing tasks, encompassing both appearance and motion adjustments. Through frequency-aware factorization of the T2V model, FADE efficiently harnesses the model's inherent structural and temporal patterns, allowing each block to excel in its designated role. We further design the spectrum-guided modulation, which adjusts the sampling trajectory using frequency cues to preserve foundational video elements and enhance text-video alignment. Extensive experiments demonstrate our framework's competitive performance and generalization across various video editing challenges. We envision our work will inspire future research on versatile video editing and efficient video diffusion model.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, pages 18208–18218, 2022. 2

[2] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 3

[3] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, pages 18392–18402, 2023. 3

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *URL https://openai. com/research/video-generation-models-as-world-simulators*, 3, 2024. 3, 4

[5] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, pages 22560–22570, 2023. 2

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 34:8780–8794, 2021. 3

[7] 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

[8] 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. 2, 3

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 2, 3, 4

[10] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[11] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *NeurIPS*, 35:8633–8646, 2022. 3

[12] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *NeurIPS*, 35:8633–8646, 2022. 2

[13] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, pages 6007–6017, 2023. 2

[14] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, pages 12888–12900. PMLR, 2022. 7

[15] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. In *CVPR*, pages 8599–8608, 2024. 2, 3, 5, 7

[16] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 3, 4

[17] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2021. 3

[18] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *CVPR*, pages 6038–6047, 2023. 2, 3, 5, 1

[19] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv preprint arXiv:2302.01329*, 2023. 2, 3

[20] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH*, pages 1–11, 2023. 2, 3

[21] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, pages 4195–4205, 2023. 3

[22] 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. 2, 7

[23] 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. In *ICCV*, pages 15932–15942, 2023. 2, 3, 7

[24] 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 *ICML*, pages 8748–8763. PMLR, 2021. 7

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 3

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 3, 4

[27] Chaehun Shin, Heeseung Kim, Che Hyun Lee, Sang-gil Lee, and Sungroh Yoon. Edit-a-video: Single video editing with object-aware consistency. In *ACML*, pages 1215–1230. PMLR, 2024. 2

[28] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2

[29] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2020. 3

[30] Vadim Titov, Madina Khalmatova, Alexandra Ivanova, Dmitry Vetrov, and Aibek Alanov. Guide-and-rescale: Self-guidance mechanism for effective tuning-free real image editing. *arXiv preprint arXiv:2409.01322*, 2024. 2, 5

[31] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, pages 1921–1930, 2023. 3

[32] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *ICLR*, 2022. 2

[33] Wen Wang, Yan Jiang, 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

[34] 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 *ICCV*, pages 7623–7633, 2023. 3, 7

[35] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 3, 4, 7, 8, 1

[36] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 7

[37] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *URL https://github. com/hpcaitech/Open-Sora*, 2024. 3, 4

[38] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 1

FADE: Frequency-Aware Diffusion Model Factorization for Video Editing
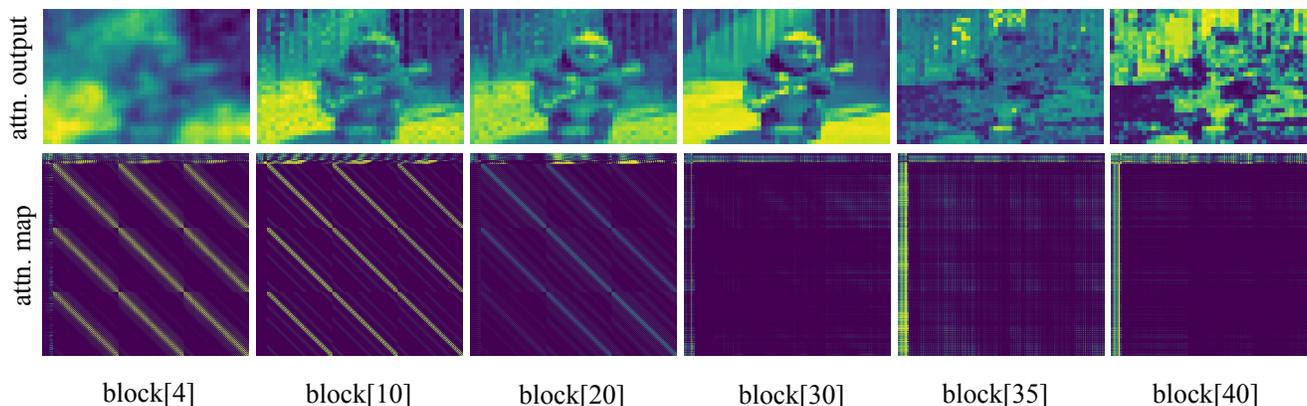
Supplementary Material

Figure A. **Visualization of attention outputs and attention maps.** We can clearly observe that the patterns of the attention features across blocks. Early blocks primarily process low-frequency structures, reflected in their blurred attention outputs and maps with multiple diagonal patterns. Conversely, later blocks focus on high-frequency details, producing attention maps with a uniformly distributed pattern.

A. Detailed Implementations

To leverage video generation priors, we employ the widely used T2V model, CogVideoX-5b [35], for video inversion and generation. In the inversion stage, the resolution of the input video is first adjusted to 720×480 to comply with the resolution constraint of CogVideoX-5b. Additionally, since the model is trained with long prompts, a chatbot is utilized to enrich the input prompt with additional details. Subsequently, DDIM inversion is performed with the total number of steps set to $T = 50$, yielding the latent trajectories. In the editing stage, the first four blocks are designated as sketching blocks. To capture spectral characteristics, we apply 3D DFT on the attention output of these blocks. Additionally, the output of the last block in DiT is utilized to compute an auxiliary guidance term. The guidance mechanism is applied during the interval $[0, 0.6T]$ of the DDIM sampling process. Furthermore, to isolate the target object from unrelated regions, we adopt the local editing trick [18] with a mask during the interval $[0, 0.8T]$. As CogVideoX-5b does not provide ideal masks, we instead utilize the method proposed in [38] to generate the necessary masks. The proposed approach is implemented on a single NVIDIA LS20 GPU with 48GB of VRAM, achieving an average processing time of approximately 4 minutes per video, making it suitable for near-real-time applications.

B. Different Functions of Attention Blocks

In this section, we delve deeper into the different functions of attention blocks in the T2V model. In Figure A, we vi-

sualize the attention output and attention map for six blocks out of a total of 42. As the block number increases, the attention output contains progressively more high-frequency components, indicating that earlier blocks establish the low-frequency, foundational structure of the video, such as object placement and movement, while later blocks focus on high-frequency refinements and details. Furthermore, the attention in earlier blocks is densely concentrated along diagonal lines, while the attention in later blocks becomes more evenly distributed. This further demonstrates that earlier blocks emphasize the key shapes and correspondences in the video, while later blocks focus on fine details. For video editing tasks, which require maintaining low-frequency features like object location and general shape while modifying high-frequency features, the best performance is achieved by using only the first four blocks, *i.e.*, the sketching blocks.

C. More Discussions

About the static background. The static background in the winter example is due to the limited prompt. As shown in Figure B, by explicitly adding '*moving forward*' to the prompt, we can generate a dynamically moving background.

Figure B. The generated video features a dynamic background, achieved by changing the prompt.

About the ablation results. Using all blocks and the entire spectrum for guidance can slightly improve the preservation (M.PSNR, SSIM) since it leverages more video features for reconstruction. However, it introduces two major issues: (1) Editing quality (CLIP) may degrade, as features in sharpening blocks often need not remain unchanged, and (2) The additional blocks increase inference time and GPU memory usage (67.6GB for all blocks), making it less feasible.

About the sensitivity of λ. Since we normalize the scale of the gradient in Eq.7, a reasonable range of λ is [10, 15], which consistently produces plausible results. As illustrated in Figure C, a larger λ improves preservation but slightly reduces textual alignment.
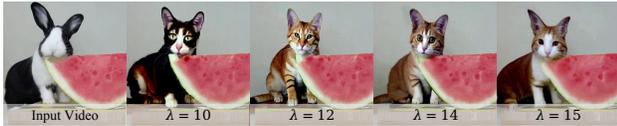

Figure C. The sensitivity of λ.

About computational cost. Our method is zero-shot and only involves sampling time (Table 2). With CPU offloading, our method uses 23.6GB GPU memory, comparable to Video-P2P (25.7GB), AnyV2V (17.1GB) and TokenFlow (11.4GB).

About the performance. We acknowledge that our method's quantitative improvement is not very significant compared with the existing methods, which heavily rely on *one-shot tuning* to enhance temporal consistency and adapt to the input sample. However, our approach is *zero-shot*. For a fair comparison, we found that fine-tuning our T2V model on the input sample significantly improves performance, as shown in Table A.

Table A. Quantitative comparisons under fine-tuning scenarios

Method	CLIP ↑	M.PSNR ↑	LPIPS ↓	OSV ↓	PF ↑
Tune-A-Video [34]	0.3522	19.86	0.4625	35.01	0.09
Video-P2P [15]	0.3589	20.57	0.3199	34.71	0.14
TokenFlow (ICLR'24)	0.3614	20.39	0.3212	34.50	0.12
FADE	0.3762	20.69	0.3085	31.36	0.27
FADE (tuned)	**0.3946**	**22.19**	**0.2937**	**30.27**	**0.38**

About video prior and block choice. In Figure D, We plotted the energy ratio of low and high-frequency components in the 3D DFT spectrum of the attention results at timestep=20 for all blocks. The frequency patterns align with the observations in Figure 2, supporting our motivation to distinguish between sketching and sharpening blocks. Moreover, this pattern remains consistent across sampling steps, as shown by the attention results in blocks [4] and [30] at timesteps $t = [10, 20, 30]$.

D. More Comparisons

In this section, we add qualitative results (Row 1&2) and comparisons with FLATTEN, TokenFlow, Rerender-
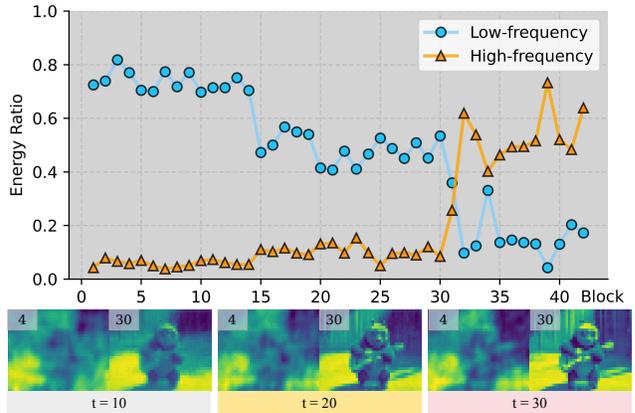

Figure D. The spectral energy distribution and visualization of the attention results.

a-Video, RAVE, and AnyV2V (Row 3), covering shape changes (sports car, duck), occlusion (desert, cat in the second figure of this PDF) and long video (taxi, >60 frames). Due to space limitations, we present one frame for comparison. As shown in Figure E, our method achieves competitive performance and textual alignment.
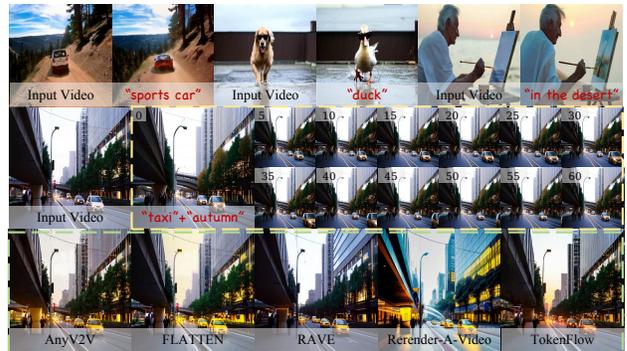

Figure E. Qualitative results and comparisons

E. More Qualitative Results

We provide additional qualitative results to further demonstrate the effectiveness and versatility of our FADE framework across a variety of editing tasks. As shown in Figure F, our method produces highly realistic and coherent outcomes, even in challenging scenarios. For instance, in the first example, transforming "child" into "panda", FADE successfully handles complex motions, such as riding a bike, which requires accurately modeling significant object movements from far to close perspectives. This highlights the framework's ability to maintain spatial consistency during large-scale transformations. Another example, the transformation from "squirrel" to "robotic mouse", emphasizes FADE's flexibility in editing both textures and shapes. The results demonstrate the framework's capacity to adaptively adjust fine details while preserving overall coherence, enabling seamless and visually plausible edits.

A child riding a bike on the road → A panda riding a bike on the road
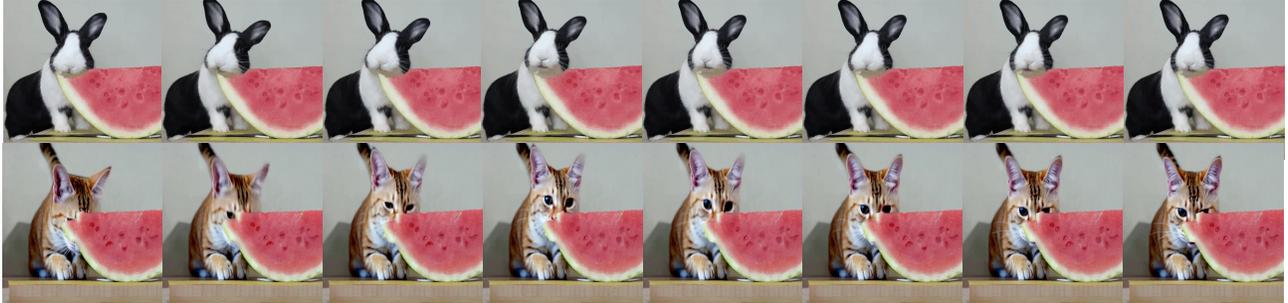
A rabbit eating a watermelon on the table → A cat eating a watermelon on the table

A squirrel eating a carrot → A robotic mouse eating a carrot

A worker balancing on a wooden plank → Spider-Man balancing on a wooden plank

Figure F. **More Qualitative Results.** FADE demonstrates impressive performance across a range of video editing tasks.

F. Failure Cases

While our framework achieves impressive results across various editing tasks, it still faces limitations in handling certain complex scenarios. As illustrated in Figure G, our method occasionally produces artifacts, such as incorrect object orientation compared to the source video. For example, the dog that should be facing left in the edited output

A man is walking a dog, which raises its head happily → A man is walking a dog, which lowers its head sadly

Figure G. **Failure Case.** Our framework may occasionally generates artifacts, such as incorrect object orientation.

incorrectly faces right. This issue arises from the absence of specific constraints to guide object placement, resulting in random variations in the generated object's orientation. Moreover, challenges also emerge when the edited objects are partially or fully occluded.