DFVEdit: Conditional Delta Flow Vector for Zero-shot Video Editing

Anonymous Author(s)
Affiliation
Address
email

Abstract

The advent of Video Diffusion Transformers (Video DiTs) marks a milestone in video generation. However, directly applying existing video editing methods to Video DiTs often incurs substantial computational overhead, due to resource-intensive attention modification or finetuning. To alleviate this problem, we present DFVEdit, an efficient zero-shot video editing method tailored for Video DiTs. DFVEdit eliminates the need for both attention modification and fine-tuning by directly operating on clean latents via flow transformation. To be more specific, we observe that editing and sampling can be unified under the continuous flow perspective. Building upon this foundation, we propose the Conditional Delta Flow Vector (CDFV) – a theoretically unbiased estimation of DFV – and integrate Implicit Cross Attention (ICA) guidance as well as Embedding Reinforcement (ER) to further enhance editing quality. DFVEdit excels in practical efficiency, offering at least 20x inference speed-up and 85% memory reduction on Video DiTs compared to attention-engineering-based editing methods. Extensive quantitative and qualitative experiments demonstrate that DFVEdit can be seamlessly applied to popular Video DiTs (*e.g.*, CogVideoX and Wan2.1), attaining state-of-the-art performance on structural fidelity, spatial-temporal consistency, and editing quality.

1 Introduction

In the wave of digitization, video creation has become a dominant form of entertainment. In response, research on controllable video generation holds considerable practical importance. While Video Diffusion Transformer (DiT) models [1–4] have revolutionized video synthesis quality, and DiT-based image editing methods [5–10] have achieved remarkable success, video editing remains challenging in preserving spatiotemporal fidelity. Critically, existing video editing methods do not fully exploit the capabilities of Video DiTs, limiting the potential for high-quality controllable video generation.

Existing video editing techniques mainly follow two paradigms: training-based methods [11–14] and zero-shot methods [15–20]. Given that the former requires resource-intensive finetuning, our work focuses on training-free video editing. For training-free video editing, a high-quality pre-trained base model is cru-

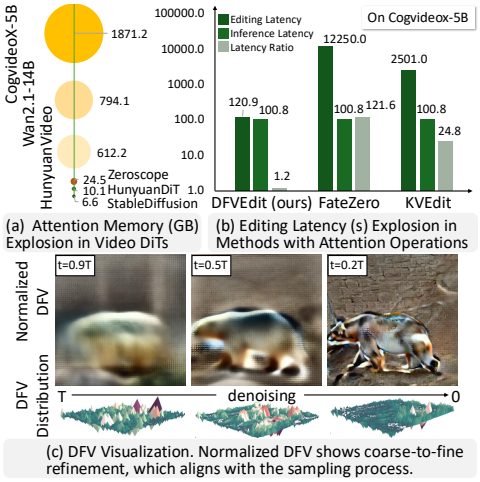


Figure 1: Key insight and motivation.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

cial. Early video editing methods primarily utilized image diffusion models [21, 22], which suffered from temporal inconsistencies due to the lack of capable video diffusion models. These early methods [23, 15, 17, 24] not only had to ensure structural integrity and editing accuracy but also required significant effort to enhance temporal coherence. In contrast, methods [16, 25] based on video diffusion models naturally excel in temporal consistency, leading us to leverage the latest Video DiTs [4, 1, 2] for video editing. Regardless of the type of base models, achieving high fidelity and temporal consistency hinges on attention engineering in most existing methods, including various attention caching and modification techniques. The key to effective attention engineering is that attentions (including keys, queries, and values) contain the spatial-temporal information of the source video, allowing for smooth editing of target regions while preserving the original content's integrity. However, attention mechanisms now consume hundreds of gigabytes of memory (Fig. 1(a)) in Video DiTs [1, 4, 2], a significant increase from previous usage in Unet-based diffusion models [26, 21, 22] and image DiT models [27, 28] at the gigabyte scale. This suggests that traditional attention engineering techniques are incompatible with Video DiTs, creating an urgent need for methods that preserve editing quality while improving computational efficiency.

Motivated by this inefficiency, we shift the focus from attention to input latents and introduce a continuous flow transformation framework for direct video latent refinement. We observe that the standard sampling process in video diffusion models—whether based on Score Matching [29] or Flow Matching [30]—can be unified under a continuous flow perspective. Based on this insight, we demonstrate that editing from the source to the target video naturally forms a time-dependent flow vector field (Fig. 1(c)), which we term the Delta Flow Vector (DFV).

Building upon this foundation, we introduce the Conditional Delta Flow Vector (CDFV) to estimate the flow from source to target latent, incorporating Implicit Cross Attention Guidance (ICA) and Embedding Reinforcement (ER) to further improve editing accuracy. The CDFV in Video DiTs inherently enforces spatial-temporal dependencies while its divergence directly determines update weights. This physically grounded formulation provides two fundamental advantages over approximation-based latent-refinement approaches like DDS [31] and SDS [32]: (1) *theoretical unification* by modeling both sampling and editing from the continuous flow perspective and (2) *computational efficiency* through divergence-determined and hyperparameter-free weights that eliminate heuristic scheduling and overcome low convergence issues inherent to shallow approximations. Moreover, for the seamless application to video editing, we enhanced spatiotemporal coherence by intrinsically avoiding randomness bias while incorporating ICA guidance and ER mechanisms (Fig. 5). Experiments show DFVEdit achieves at least 20× speed-up and 85% memory reduction over attention-engineering-based methods on Video DiTs (*e.g.*, CogVideoX, Wan2.1), while maintaining SOTA performance in fidelity, temporal consistency, and editing quality. Consequently, our approach offers an efficient and versatile solution for zero-shot video editing on Video DiTs.

2 Related work

Video Diffusion Transformer. Video Diffusion Transformers have evolved from early 3D-UNet-based designs [33, 26, 34, 35] to modern 3D-Transformer-based designs [3]. Advanced models such as Open-Sora [36, 37], CogVideoX [1], HunyuanVideo [2] and Wan [4] have all or part of the following key innovations: replacement of 3D-UNets with scalable 3D-Transformer blocks; integration of cross-attention and self-attention into a unified 3D-full-attention [1, 2]; and adoption of 3D-VAE [1] for spatiotemporal latent compression. Some Video DiTs [27, 4] are combined with Flow Matching [30] while others [1] adopt SDE [29] samplers like DPM-solver [38].

Image editing on Diffusion Transformer. With the rise of Diffusion Transformer [3], DiT-based image editing methods [28, 27] have emerged. However, directly applying image editing methods to videos often fails to address temporal consistency and motion fidelity. Additionally, adapting them to Video DiTs introduces extra challenges. Firstly, generalization limitations occur when applying methods [8, 6, 9, 10, 39, 40] that rely on rectified flow [41] or distilled few-step models [42] to Video DiTs that are not combined with rectified flow or distillation techniques. Secondly, efficiency limitations are present for image editing methods [43] that require finetuning. Furthermore, even generalized and efficient methods like DiT4Edit [5] and KVEdit [7], which use attention or key-value caching and modification, still face prohibitive computational costs due to the more massive attention overhead in Video DiTs compared to image DiTs.

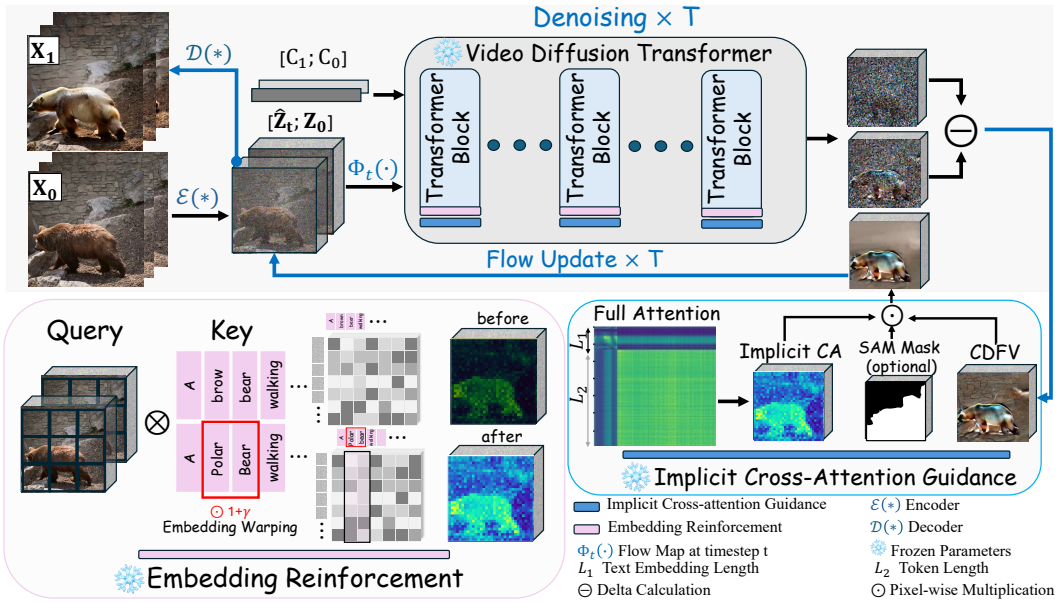


Figure 2: **DFVEdit overview.** Follow these steps for DFVEdit: (1) Encode $\mathbf{X}_0$ into the latent space $\mathbf{Z}_0$, and initialize the target latent variable as $\hat{\mathbf{Z}}_T = \mathbf{Z}_0$. (2) Transform $[\hat{\mathbf{Z}}_T; \mathbf{Z}_0]$ via the flow map $\Phi_T(\cdot)$. (3) Feed the result with prompt embeddings $[C_1, C_0]$ into the Video Diffusion Transformer, compute the delta difference to obtain the CDFV at timestep T, then refine it using ER and ICA. (4) Update $\hat{\mathbf{Z}}_T \rightarrow \hat{\mathbf{Z}}_{T-1}$ using the enhanced CDFV, and iterate (1)-(4) until reaching $\hat{\mathbf{Z}}_0$. (5) Decode $\hat{\mathbf{Z}}_0$ to generate the target video $\mathbf{X}_1$.

Video editing. Video editing via diffusion models is dominated by two paradigms: training-based and training-free methods. Training-based approaches [44–49, 12, 14] enhance pre-trained image diffusion models [21] with spatiotemporal modules, optimizing for complex edits but at high computational costs, limiting real-time applications. Conversely, training-free methods emphasize computational efficiency and real-time capability. Training-free video editing commonly involves two stages: latent space initialization and editing condition injection. Latent space initialization typically follows three paradigms: (1) forward diffusion with some steps for preserving low-frequency features [50, 51], (2) DDIM [22] inversion for enabling deterministic reconstruction [15, 17], or (3) direct source latent usage [31, 32]. For editing condition injection, most existing zero-shot methods heavily rely on *attention engineering* to maintain spatial-temporal fidelity. For instance, FateZero [15] enhances temporal consistency by caching attention maps from DDIM [22] inversion and integrating them into the denoising process; TokenFlow [17] improves spatiotemporal coherence by leveraging cached attention outputs from DDIM inversion for inter-frame correspondences and incorporating extended attention blocks during denoising; VideoDirector [20] achieves fine-grained editing via SAM [52] masks by fusing self-attention with reconstruction attention and mask guidance; and VideoGrain [19] realizes complex semantic structure modifications through SAM masks while operating on complex attention map modifications. These attention-engineered methods face scalability challenges in Transformer blocks [53], particularly for Video DiTs [2, 4] where attention memory demands grow dramatically (Fig. 1). Moreover, approaches [54–56, 24] free of attention engineering suffer from structural degradation: FRAG [54] mitigates blurring and flickering through frequency processing but compromises fidelity due to basic DDIM inversion [57] for source content retention; DMT [24] employs SSM [24] loss for motion transfer yet underperforms in detail preservation; and first-frame propagation methods (*e.g.*, StableV2V [55], AnyV2V [56]) introduce accumulating artifacts without full-frame coordination. In conclusion, designing efficient and high-quality editing methods tailored for Video DiTs remains a critical challenge.

3 Method

Fig. 2 provides an overview of DFVEdit. Given a source video $\mathbf{X_0} \in \mathbb{R}^{F\times 3\times H\times W}$ comprising F RGB frames at resolution $H \times W$, together with source and target text prompts (P_0, P_1), our method supports both global stylization and local modifications (shape and attribute editing). The edited

video $\mathbf{X_1}$ preserves spatiotemporal integrity in unedited regions while ensuring motion fidelity and precise alignment with P_1. Our approach leverages two key insights: manipulating latent space is more computationally efficient than manipulating attention (Fig. 1), and editing can be modeled as the continuous flow transformation between the source and target videos (Sec 3.1). We introduce the Conditional Delta Flow Vector (CDFV) (Sec 3.2) for this transformation. To enhance video editing performance, we utilize Implicit Cross-Attention Guidance and Embedding Enforcement (Sec 3.3) to improve spatiotemporal fidelity.

3.1 Unified continuous flow perspective on sampling and editing

Diffusion models include inverse and forward processes. The inverse process is typically parameterized as a Markov chain with learned Gaussian transitions, mapping noisy inputs to clean outputs. Conversely, the forward process gradually adds Gaussian noise to the clean input according to a variance schedule. As mentioned in [58, 59, 29], given a data input x, both inverse and forward processes can be regarded as overdamped Langevin Dynamics [60] (named Stochastic Differential Equation (SDE) in Score Matching [29]):

$$dx_t = f(x_t, t)dt + g(x_t, t)dW \tag{1}$$

where $f(x_t, t)$ is the drift coefficient corresponds to deterministic direction and $g(x_t, t)$ is the diffusion coefficient corresponds to disturbing intensity and dW is a Wiener process and the probability density $P(x_t, t)$ can be described by introducing the Fokker-Planck equation [61] combined with the Ito's lemma [62] and the concept of probability flow:

$$\frac{\partial P(x_t, t)}{\partial t} = -\nabla \left[\left(f(x_t, t) - \frac{g^2(x_t, t)}{2} \nabla log P(x_t, t) \right) P(x_t, t) \right] \tag{2}$$

Eq. 2 generalizes traditional sampling methods like DDPM [63] and DDIM [22]. ***This formulation reveals that methods based on SDE [29] obey the continuity equation principle of Flow Matching [30] and can be unified under a continuous flow perspective.*** The continuous flow is characterized by a vector field $v_t(x_t) = f(x_t, t) - \frac{g^2(x_t,t)}{2} \nabla log P(x_t, t)$, enabling state transitions from x_t to $x_{t+\Delta t}$ either through flow map Φ_t in Eq. 3 or through its Euler discretized approximation in Eq. 4:

$$\begin{cases} \dfrac{d}{dt}\Phi_t(x) = v_t(\Phi_t(x)) \\ \Phi_0(x) = x \end{cases} \tag{3}$$

$$x_{t+\Delta t} = x_t + \Delta t * v_t(\Phi_t(x)) \tag{4}$$

As discussed in Section 2, zero-shot video editing includes two stages: latent space initialization and editing condition injection. The first stage involves a standard sampling process. In the second stage, we derive an isomorphism with sampling process by formulating video editing as:

$$X_{t-1}^{\text{edit}} = g_{\theta_{2,t}}\Big(X_t^{\text{edit}},\ \underbrace{\epsilon_{\theta_1}(X_t^{\text{edit}}, t)}_{\text{Canonical Denoiser}} + \lambda \underbrace{C(X_t^{\text{edit}}, t, *)}_{\text{Control Term}} \Big) \tag{5}$$

where $\{X_t^{\text{edit}}\}_{t=0}^{T}$ defines the state trajectory of the edited video in the sampling process; $g_{\theta_{2,t}}$ is differentiable transition function parameterized by learnable θ_2; ϵ_{θ_1} is pretrained diffusion model with frozen θ_1; $C(x, t, *)$ is the control term with intensity $\lambda \geq 0$ and optional extra input $*$. Under the Euler discretization scheme with step size $\Delta t \to 0$ and $\theta_2 = \mathcal{I}$, the discrete process in Eq. 5 converges to the controlled SDE:

$$dX_t^{\text{edit}} = \underbrace{\left[-\frac{\beta(t)}{2} X_t^{\text{edit}} + \frac{\beta(t)}{2} \nabla \log p_t(X_t^{\text{edit}}) + \lambda \frac{\beta(t)}{2} \sigma(t) C(X_t^{\text{edit}}, t, *) \right]}_{f_{\theta_1}(X_t^{\text{edit}}, t)} dt + \underbrace{\sqrt{\beta(t)}}_{g(t)} dW \tag{6}$$

where $\nabla \log p_t(X_t^{\text{edit}})$ is the score function, and $\sigma(t) = \sqrt{(1 - \alpha(t))/\alpha(t)}$ is the signal-to-noise ratio coefficient with $\alpha(t) = e^{-\int_0^t \beta(s)ds}$. ***The structural isomorphism between Eq. 6 and the stochastic differential equation in Eq. 1 indicates that video editing processes can be represented within a continuous flow sampling framework***, as shown in Eq. 3 (see Appendix for more details).

3.2 Conditional Delta Flow Vector

Building upon the isomorphic correspondence between editing and sampling, we introduce the Conditional Delta Flow Vector (CDFV) to establish a direct continuous flow bridge from the source video to the target video.

Delta Flow Vector. Given the initial distribution $p(Z_T) = \mathcal{N}(Z_T; 0, I)$ for the reverse process and a clean video latent Z, Eq. 3 implies the existence of a time-dependent flow map Φ that:

$$Z = Z_T - \sum_{t=0}^{T} \Delta t v_t(\Phi_t(Z)) \tag{7}$$

Assuming the source and target latents $(\hat{Z}_0, Z_0)$ and their corresponding prompts (P_1, P_0) are given, we replace Z in Eq. 7 with Z_0 and $\hat{Z}_0$ respectively and define the Delta Flow Vector (DFV) as $\Delta v_t(\hat{Z}_0, Z_0) = v_t(\Phi_t(\hat{Z}_0)) - v_t(\Phi_t(Z_0))$, and the target latent $\hat{Z}_0$ can be expressed in terms of the source latent Z_0 as:

$$\hat{Z}_0 = Z_0 - \sum_{t=0}^{T} \Delta t \, \Delta v_t(\hat{Z}_0, Z_0). \tag{8}$$

Eq. 8 establishes a continuous flow directly from the source latent Z_0 ***to the target latent*** $\hat{Z}_0$, with the vector field defined as $v_t = \Delta v_t(\hat{Z}_0, Z_0)$. While prior works [64, 65, 31] heuristically observed that latent differences indicate editing regions, we rigorously prove this as a special case of DFV when the transformation state and vector field satisfy the continuity equation (Eq. 3).

Conditional Delta Flow Vector. The direct computation of $\Delta v_t(\hat{Z}_0, Z_0)$ is intractable since $\hat{Z}_0$ is the editing target. To resolve this problem, ***we leverage the terminal condition of diffusion processes to derive an unbiased estimation of DFV.*** From Eq. 2 we obtain $v_t(x_t) = f(x_t, t) - \frac{g^2(x_t,t)}{2} \nabla log P(x_t, t)$. As t approaches T, and given that $P(x_t, t)$ is the probability density of x_t, if we set winner process of Z_0 and $\hat{Z}_0$ is equal, then $g(Z_0, t) = g(\hat{Z}_0, t)$. Consequently, as $t \to T$, both $P(Z_0, t)$ and $P(\hat{Z}_0, t)$ follow a normal distribution $\mathcal{N}(Z_T; 0, I)$ with zero mean and unit variance. Moreover, $\hat{Z}_t$ is equivalent to Z_t as $t \to T$, and we have:

$$\Delta v_t(\hat{Z}_0, Z_0) \underset{t \to T}{=} f_{\theta_1, c_1}(Z_t, t) - f_{\theta_1, c_0}(Z_t, t) \tag{9}$$

The latent $\hat{Z}_{T-\Delta t}$ can be updated using Eq. 10, which corresponds to applying the continuous flow map from $\hat{Z}_0$ as defined in Eq. 11:

$$\hat{Z}_{T-\Delta t} = Z_{T-\Delta t} - \Delta t \left[f_{\theta, c_1}(Z_T, t) - f_{\theta, c_0}(Z_T, t) \right], \tag{10}$$

$$\hat{Z}_{T-\Delta t} = \Phi_{T-\Delta t}(\hat{Z}_0). \tag{11}$$

We sequentially obtain all $v_t(\Phi(\hat{Z}_0))$ and define the Conditional Delta Flow Vector (CDFV) in Eq. 12.

$$\begin{cases} \Delta v_t(Z_0, c_0, c_1) = v_{t,c_1}(\hat{Z}_t) - v_{t,c_0}(\Phi_t(Z_0)) \\ \hat{Z}_T = \Phi_T(Z_0) \end{cases} \tag{12}$$

Theoretically, the CDFV provides an unbiased estimate of DFV. By using the CDFV as a control term, defined in Eq. 13, and integrating it into Eq. 6, we maintain a computational complexity similar to that of the basic sampling process. See the Appendix for more details.

$$C(\hat{Z}_t, t, *) = \frac{\nabla \log P(\hat{Z}_t, t) - \nabla \log P(\Phi_t(Z_0), t)}{\sigma(t)} \tag{13}$$

3.3 Spatiotemporal enhancement for CDFV

Implicit Cross-Attention Guidance. Although CDFV extracted from Video DiTs theoretically captures semantic differences between P_0 and P_1 with temporal coherence (Sec 3.2), empirical studies reveal persistent background leakage (Fig. 2). We attribute this phenomenon to the score function $\nabla_X \log p_t(X; \theta)$, which is learned by the model and may not perfectly align with theoretical

expectations. This discrepancy can introduce local distributional drift in unedited regions, and such shifts have the potential to cause noticeable alterations in the background of edited videos (see Fig. 5 for examples). Segmentation masks play a crucial role in effective structure guidance, and cross-attention, as highlighted in [16, 15, 66], exhibit significant potential for shape editing tasks. This is attributed to their time-aware adaptability and target-following characteristics, which enhance the capability to maintain structural integrity and motion consistency over time. Although most recent Video DiTs have moved from discrete cross-attention to Full Attention [1] for more accurate spatial-temporal learning, we introduce Implicit Cross-Attention derived from Full Attention. ICA still retains the essence of traditional cross-attention and guides shape editing effectively. Given text embeddings $\mathbf{E} \in \mathbb{R}^{N\times d}$ and latent video tokens $\mathbf{B} \in \mathbb{R}^{M\times d}$, Full Attention mechanism first concatenates them to form a larger matrix $\mathbf{C} = [\mathbf{E}; \mathbf{B}] \in \mathbb{R}^{(N+M)\times d}$, each row of $\mathbf{C}$ can be considered as both Query (Q), Key (K), and Value (V). The full attention map is computed as follows:

$$\mathcal{A} = \text{Softmax}\left(\frac{\mathbf{C}\mathbf{C}^\top}{\sqrt{d}}\right) = \begin{bmatrix} \mathcal{A}_{EE} & \mathcal{A}_{EB} \\ \mathcal{A}_{BE} & \mathcal{A}_{BB} \end{bmatrix} \in \mathbb{R}^{(N+M)\times(N+M)} \tag{14}$$

We identify that the off-diagonal block $\mathcal{A}_{EB}$ or $\mathcal{A}_{BE}$ inherently encodes cross-modal interactions. Our *Implicit Cross-Attention* extracts this block of different timesteps and binarizes it into M_t. We mask $\Delta v_t(Z_0, c_0, c_1)$ with M_t to restrain the changes in the unedited region as Eq. 15. M_t can also be optionally combined with the popular SAM [52] masks using Boolean operations.

$$\Delta v_{t,M_t}(Z_0, c_0, c_1) = M_t \odot \left[v_{t,c_1}(\hat{Z}_t) - v_{t,c_0}(\Phi_t(Z_0))\right] \tag{15}$$

Target Embedding Reinforcement. We observe that in 3D Full-Attention, the effect of text embeddings diminishes as frame length increases. This phenomenon is particularly evident in global editing tasks such as stylization. We attribute this issue to the competition between fixed-length text tokens $\mathbf{E} \in \mathbb{R}^{N\times d}$ and an increasing number of spatiotemporal tokens $\mathbf{Z} \in \mathbb{R}^{F\times H\times W\times d}$. As the video duration grows, vectors associated with stylization embeddings become increasingly sparse across frames. This sparsity may further reduce the guidance fidelity of the text embeddings. To address these challenges, we propose Embedding Reinforcement (ER) for prompt alignment:

$$\tilde{\mathbf{E}}^{(k)} = \mathbf{E} + \gamma^{(k)} \odot \mathbf{E} \tag{16}$$

where k is used to locate the target embedding for editing, and its value is amplified by $\gamma + 1$. Specifically, we set $\gamma = 0.2$ for shape editing and $\gamma = 5$ for stylization. By reinforcing the embeddings, the cross-attention map is reweighted to focus on regions more relevant to the editing target, enhancing editing precision.

4 Experimental results

Experimental setup. We adopt the pretrained CovideoX-5B [1] as the base model and also extend our method to Wan2.1-14B [4] to demonstrate the robustness and flexibility of DEVEdit. All experiments are conducted on one A100-80G GPU. We evaluate our methods on public DAVIS2017 [68] videos and Internet open-source videos from Pexels [69]. In comparison experiments, we test 40-frame videos with a resolution of 512 × 512. Our focus is on training-free appearance editing, including local editing (shape and attribute editing), and global editing (stylization).

Baselines. For baselines, we compare against image diffusion-based training-free editing methods, including FateZero [15], TokenFlow [17], VideoDirector [20], and VideoGain [19], which rely on attention engineering; ControlVideo [18], FLATTEN [67], and DMT [24], which are free of attention engineering; FreeMask [16], which is based on a U-net-based video diffusion model with attention engineering; and SDEdit [50] (directly applied to CogVideoX-5B [1] base model for video editing).

4.1 Evaluation

Qualitative evaluation. Fig. 3 provides qualitative comparison results, showcasing our method's superiority in structure fidelity, motion integrity, and temporal consistency over other prominent baselines. For **single object editing** (first column), FateZero [15], TokenFlow [17], and VideoDirector [20] exhibit noticeable flickering, while ControlVideo [18], FLATTEN [67], and DMT [24]

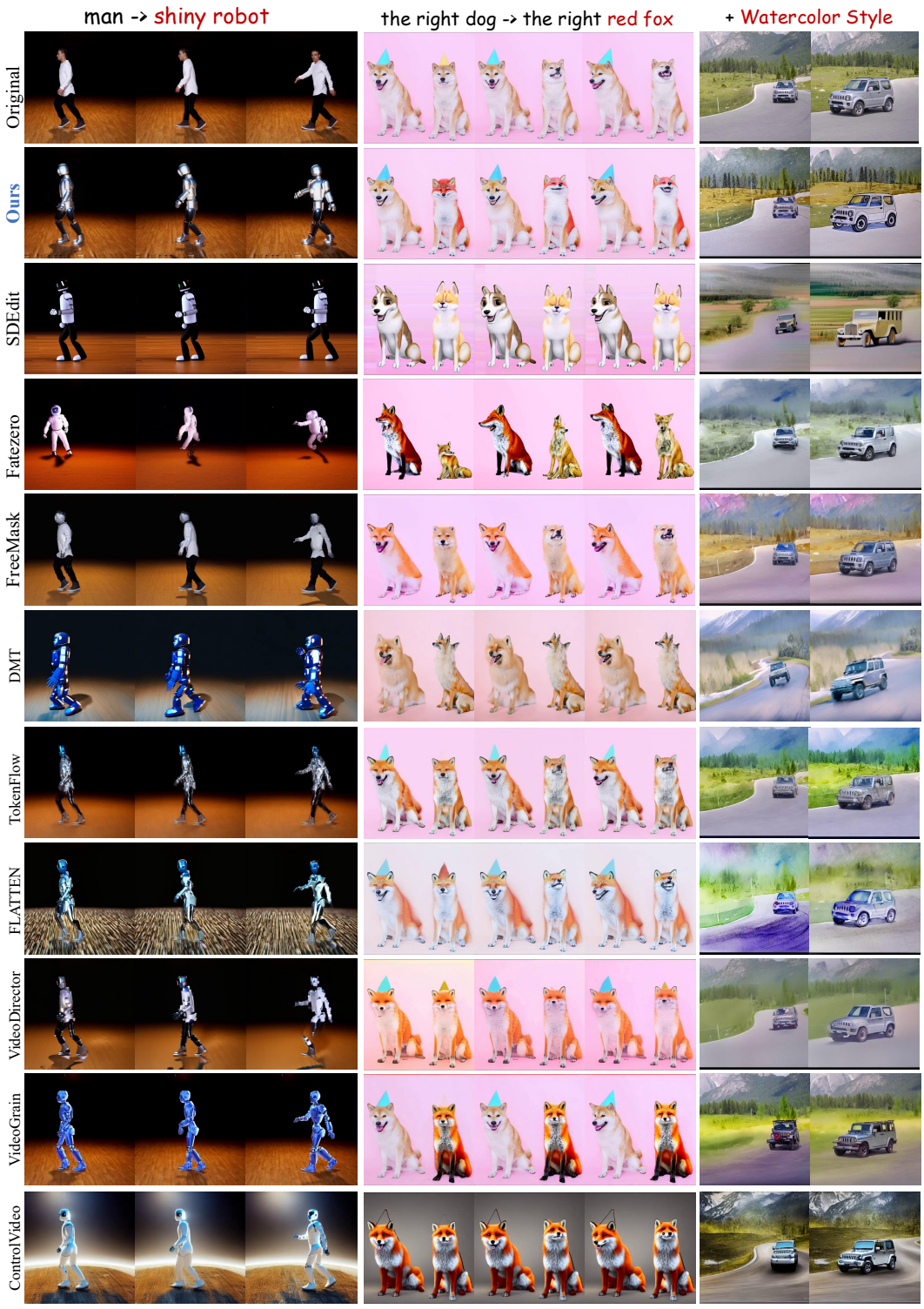


Figure 3: **Comparison.** Most methods based on attention-engineering and image diffusion models (FateZero [15], TokenFlow [17], VideoDirector [20]) suffer from flickering and fail in multi-object editing. While VideoGrain [19] enhances multi-object editing, it is inferior in structure consistency and motion detail fidelity (the second column). Attention-engineering-free approaches (FLATTEN [67], DMT [24], ControlVideo [18]) exhibit structural infidelity. FreeMask [16] improves temporal consistency but remains constrained by its 3D-Unet base model. Applying the image editing method SDEdit [50] directly to VideoDiTS compromises spatial-temporal fidelity. In comparison, our method achieves SOTA performance in fidelity, alignment, and temporal consistency. Refer to the supplementary material for more results.

Table 1: **Quantitative evaluation and user study results.**

Method	Consistency		Fidelity		Alignment	User Study			Computation Efficiency		
	CLIP-F↑	E_{warp}↓	M.PSNR↑	LPIPS↓	CLIP-T↑	Edit↑	Quality↑	Consistency↑	VRAM ↓	RAM↓	Latency↓
SDEdit [50]	0.9811	1.67	20.52	0.4090	27.46	66.57	80.45	85.66	1.01	1.13	**0.87**
FateZero [15]	0.9289	3.09	23.39	0.2634	26.08	58.87	50.63	56.89	2.32	21.44	3.40
FreeMask [16]	0.9699	2.00	29.92	0.2314	27.06	75.88	74.67	77.13	1.64	25.58	5.65
Tokenflow [17]	0.9583	1.48	29.97	0.2247	29.78	70.12	53.45	57.41	1.43	3.69	13.03
VideoDirector [20]	0.9555	2.44	28.97	0.3205	27.50	74.13	73.25	71.45	6.00	2.26	27.97
VideoGrain [19]	0.9695	2.68	30.70	0.2948	27.79	76.41	79.87	70.61	2.35	2.61	13.44
FLATTEN [67]	0.9510	4.89	15.91	0.3559	27.57	63.45	69.45	68.32	1.54	7.31	4.61
ControlVideo [18]	0.9533	3.10	10.08	0.4015	27.06	56.08	55.33	59.41	8.74	1.62	9.45
DMT [24]	0.9668	3.50	15.95	0.5096	25.34	62.66	68.36	69.88	5.64	3.32	24.40
DFVEdit	**0.9924**	**1.12**	**31.18**	**0.1886**	**30.84**	**87.65**	**84.56**	**86.98**	**0.95**	**0.86**	1.20
w/o ICA	0.9922	1.25	29.33	0.1920	31.02	86.45	84.33	86.56	0.94	0.78	1.19
w/o EmbedRF	0.9913	1.13	31.15	0.1889	29.25	86.04	83.15	86.13	0.95	0.85	1.20

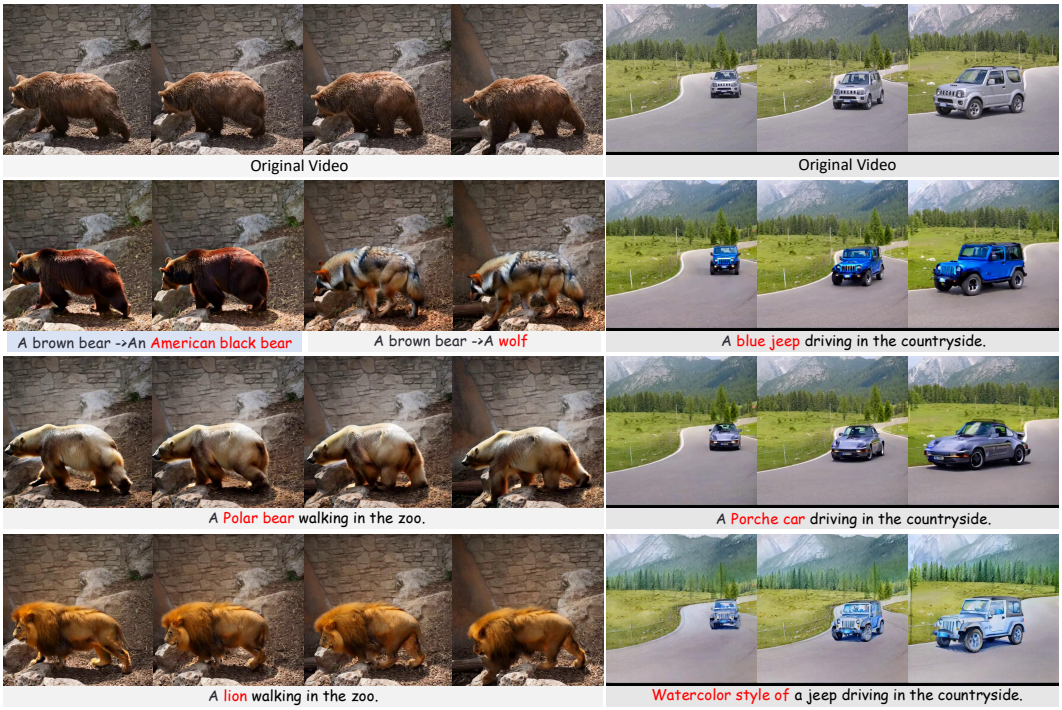


Figure 4: **Extensive qualitative results.** The extensive experiments take Wan2.1-14B [4] as the base model, demonstrating the generalization of DFVEdit for Video DiTs. See the supplementary material for more results.

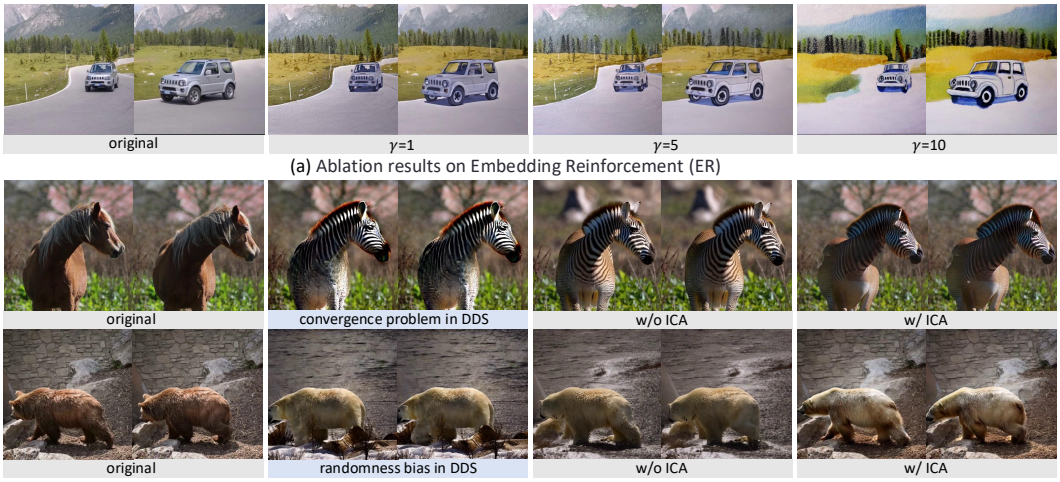
fail to preserve the details of unedited regions. For **multi-object editing** (second column), most methods struggle with editing accuracy; although VideoGrain [19] achieves success in multi-object editing using fine-grained SAM [52] masks, it falls short in maintaining motion detail fidelity (e.g., a mismatch between the fox and dog expressions). For **stylization** (third column), Freemask [16], which is based on a UNet-based video diffusion model, performs notably well, while other methods still show inconsistencies in color tone and structural details (refer to the supplementary material for video displays). Additionally, we extended FateZero [15] and KVEdit [7] directly to Cogvideo-5B [1] to compare editing quality and efficiency. Due to space limitations, please refer to the appendix for more detailed comparison results. Fig. 4 provides the extensive experiment results on Wan2.1-14B [4], which also demonstrates high editing quality with respect to structure fidelity, motion integrity, and prompt alignment. Wan [4] is combined with Flow Matching [30], while Cogvideox [1] is based on Score Matching [29]. As illustrated in both Fig. 4 and Fig. 3, DFVEdit achieves consistent editing quality across popular Video DiTs, whether based on Score Matching [29] or Flow Matching [30].

Quantitative evaluation. In Tab. 1, we compare with baselines using both automatic metrics and human evaluations, following [15, 70, 16, 17, 12]. Specifically, **CLIP-F** calculates inter-frame cosine similarity to assess structural consistency, while $\mathbf{E_{warp}}$ measures warping error [17] to evaluate motion fidelity. Additionally, **M.PSNR** computes the Masked Peak Signal-to-Noise Ratio between

source and target videos to gauge the fidelity of unedited regions, and **LPIPS** evaluates the Learned Perceptual Image Patch Similarity for overall structural fidelity. Moreover, **CLIP-T** quantifies the alignment between the target prompt and video through the CLIP Score [71]. Regarding user studies, we focus on Target Prompt Alignment (**Edit**), Overall Editing Quality including fidelity of unedited areas, minimal filtering and blurring (**Quality**), and Motion and Structural Consistency (**Consistency**). The results demonstrate that DFVEdit achieves superior spatial-temporal consistency, fidelity, and prompt alignment compared to other methods. Furthermore, to evaluate memory and computational efficiency, we measure Relative GPU Memory Consumption (**VRAM**), defined as the ratio of editing consumption on GPU relative to original inference consumption; Relative Inference Latency (**Latency**), which assesses the ratio of editing latency to inference latency; and Relative CPU Memory Consumption (**RAM**), measuring the ratio of editing consumption on CPU over original inference consumption. These metrics highlight the practical efficiency of DFVEdit. We also extend FateZero [15] and KVEdit [7] to CogVideoX-5B [1] to evaluate their efficiencies. Some findings are illustrated in Fig. 1(b), demonstrating that these methods, originally designed for image diffusion with attention engineering, incur significant computational overhead when applied to Video DiTs.

4.2 Ablation results

We evaluate the efficacy of CDFV, ICA, and ER in our ablation study. In Fig. 5(a), we vary the Embedding Reinforcement factor γ from 1 to 10. Without reinforcement ($\gamma = 1$), stylization effects are negligible. Stylization improves as γ increases but degrades with excessively high values. Empirically, $\gamma = 5$ optimizes stylization without compromising structural fidelity or visual quality. Fig. 5(c) shows that omitting Implicit Cross-Attention Guidance leads to unintended changes in unedited regions. Incorporating cross-attention mechanisms significantly enhances structural fidelity and overall quality. In Fig. 5(b), we replace CDFV with the stochastic latent refinement vector in DDS [31]. In this ablation, for 'horse' experiment, ICA and ER are kept, while for 'bear' they are omitted for a fair comparison. The results highlights the effectiveness of CDFV. For additional qualitative and quantitative comparison and ablation results, please refer to the Appendix.



(a) Ablation results on Embedding Reinforcement (ER)

(b) Ablation results on replacing CDFV with DDS vector.

(c) Ablation results on Implicit Cross Attention (ICA) Guidance

Figure 5: **Ablation.** (a)(c) demonstrate the effectiveness of ER and ICA. (b) highlights limitations of popular approximation-based latent refinement methods [31] in video editing, including: low convergence leading to unnatural changes and unpredictable convergence times; randomness bias resulting in unsatisfactory structural fidelity. Refer to the supplementary material for more results.

5 Conclusion

We present DFVEdit, an efficient and effective zero-shot video editing framework tailored for Video Diffusion Transformers. DFVEdit realizes video editing through the direct flow transformation of the clean source latent. We theoretically unify editing and sampling from the continuous flow perspective, propose CDFV to estimate the flow vector from the source video to the target video, and further enhance the editing quality with ICA guidance and ER mechanism. Extensive experiments demonstrate the efficacy of DFVEdit on Video DiTs.

References

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

[2] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[3] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[4] Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[5] Kunyu Feng, Yue Ma, Bingyuan Wang, Chenyang Qi, Haozhe Chen, Qifeng Chen, and Zeyu Wang. Dit4edit: Diffusion transformer for image editing. *arXiv preprint arXiv:2411.03286*, 2024.

[6] Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. *arXiv preprint arXiv:2412.08629*, 2024.

[7] Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. Kv-edit: Training-free image editing for precise background preservation. *arXiv preprint arXiv:2502.17363*, 2025.

[8] Yusuf Dalva, Kavana Venkatesh, and Pinar Yanardag. Fluxspace: Disentangled semantic editing in rectified flow transformers. *arXiv preprint arXiv:2412.09611*, 2024.

[9] Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. *arXiv preprint arXiv:2410.10792*, 2024.

[10] Guanlong Jiao, Biqing Huang, Kuan-Chieh Wang, and Renjie Liao. Uniedit-flow: Unleashing inversion and editing in the era of flow models. *arXiv preprint arXiv:2504.13109*, 2025.

[11] 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.

[12] 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.

[13] Chaehun Shin, Heeseung Kim, Che Hyun Lee, Sang-gil Lee, and Sungroh Yoon. Edit-a-video: Single video editing with object-aware consistency. In *Asian Conference on Machine Learning*, pages 1215–1230. PMLR, 2024.

[14] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8599–8608, 2024.

[15] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15932–15942, 2023.

[16] Lingling Cai, Kang Zhao, Hangjie Yuan, Yingya Zhang, Shiwei Zhang, and Kejie Huang. Freemask: Rethinking the importance of attention masks for zero-shot video editing. *arXiv preprint arXiv:2409.20500*, 2024.

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

[18] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023.

[19] Xiangpeng Yang, Linchao Zhu, Hehe Fan, and Yi Yang. Videograin: Modulating space-time attention for multi-grained video editing. In *The Thirteenth International Conference on Learning Representations*.

[20] Yukun Wang, Longguang Wang, Zhiyuan Ma, Qibin Hu, Kai Xu, and Yulan Guo. Videodirector: Precise video editing via text-to-video models. *arXiv preprint arXiv:2411.17592*, 2024.

[21] 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.

[22] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[23] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023.

[24] Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8466–8476. IEEE Computer Society, 2024.

[25] Max Ku, Cong Wei, Weiming Ren, Huan Yang, and Wenhu Chen. Anyv2v: A plug-and-play framework for any video-to-video editing tasks. *arXiv e-prints*, pages arXiv–2403, 2024.

[26] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023.

[27] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, et al. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024.

[28] Chenglin Yang, Celong Liu, Xueqing Deng, Dongwon Kim, Xing Mei, Xiaohui Shen, and Liang-Chieh Chen. 1.58-bit flux. *arXiv preprint arXiv:2412.18653*, 2024.

[29] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[30] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[31] Amir Hertz, Kfir Aberman, and Daniel Cohen-Or. Delta denoising score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2328–2337, 2023.

[32] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*.

[33] Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023.

[34] 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.

[35] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024.

[36] 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. *arXiv preprint arXiv:2412.20404*, 2024.

[37] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.

[38] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022.

[39] Daniel Garibi, Or Patashnik, Andrey Voynov, Hadar Averbuch-Elor, and Daniel Cohen-Or. Renoise: Real image inversion through iterative noising. In *European Conference on Computer Vision*, pages 395–413. Springer, 2024.

[40] Gilad Deutch, Rinon Gal, Daniel Garibi, Or Patashnik, and Daniel Cohen-Or. Turboedit: Text-based image editing using few-step diffusion models. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–12, 2024.

[41] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[42] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pages 87–103. Springer, 2024.

[43] Trong-Tung Nguyen, Quang Nguyen, Khoi Nguyen, Anh Tran, and Cuong Pham. Swiftedit: Lightning fast text-guided image editing via one-step diffusion. *arXiv preprint arXiv:2412.04301*, 2024.

[44] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025.

[45] Elia Peruzzo, Vidit Goel, Dejia Xu, Xingqian Xu, Yifan Jiang, Zhangyang Wang, Humphrey Shi, and Nicu Sebe. Vase: Object-centric appearance and shape manipulation of real videos. *arXiv preprint arXiv:2401.02473*, 2024.

[46] 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.

[47] Yuchao Gu, Yipin Zhou, Bichen Wu, Licheng Yu, Jia-Wei Liu, Rui Zhao, Jay Zhangjie Wu, David Junhao Zhang, Mike Zheng Shou, and Kevin Tang. Videoswap: Customized video subject swapping with interactive semantic point correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7621–7630, 2024.

[48] Bojia Zi, Shihao Zhao, Xianbiao Qi, Jianan Wang, Yukai Shi, Qianyu Chen, Bin Liang, Rong Xiao, Kam-Fai Wong, and Lei Zhang. Cococo: Improving text-guided video inpainting for better consistency, controllability and compatibility. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 11067–11076, 2025.

[49] 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. *Advances in Neural Information Processing Systems*, 36, 2024.

[50] 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. *arXiv preprint arXiv:2108.01073*, 2021.

[51] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023.

[52] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[54] Sunjae Yoon, Gwanhyeong Koo, Geonwoo Kim, and Chang D Yoo. Frag: Frequency adapting group for diffusion video editing. *arXiv preprint arXiv:2406.06044*, 2024.

[55] Chang Liu, Rui Li, Kaidong Zhang, Yunwei Lan, and Dong Liu. Stablev2v: Stablizing shape consistency in video-to-video editing. *arXiv preprint arXiv:2411.11045*, 2024.

[56] Max Ku, Cong Wei, Weiming Ren, Huan Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *Transactions on Machine Learning Research*.

[57] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*.

[58] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[59] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[60] On the theory of the brownian motion. *Physical review*, 36(5):823, 1930.

[61] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[62] Peter E Kloeden, Eckhard Platen, Peter E Kloeden, and Eckhard Platen. *Stochastic differential equations*. Springer, 1992.

[63] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[64] Ligong Han, Song Wen, Qi Chen, Zhixing Zhang, Kunpeng Song, Mengwei Ren, Ruijiang Gao, Anastasis Stathopoulos, Xiaoxiao He, Yuxiao Chen, et al. Proxedit: Improving tuning-free real image editing with proximal guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4291–4301, 2024.

[65] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022.

[66] 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.

[67] Yuren Cong, Mengmeng Xu, Shoufa Chen, Jiawei Ren, Yanping Xie, Juan-Manuel Perez-Rua, Bodo Rosenhahn, Tao Xiang, Sen He, et al. Flatten: optical flow-guided attention for consistent text-to-video editing. In *The Twelfth International Conference on Learning Representations*.

[68] 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.

[69] Pexels. Pexels free stock video clips and motion graphics. `https://www.pexels.com`. Accessed: 2025-05-15.

[70] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8599–8608, 2024.

[71] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Section 1

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: see the Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Section 3.2, Section 3.1 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Section 4 and the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: We have the public project webside, and the code will be released later.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4 and the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA] .

Justification: Not related to our work.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 4 and the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes] .

   Justification: have reviewed.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: there is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: the paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: the paper does not use existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: the paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: the paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: the paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: the core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.