Tsinghua University
July 24, 2026

# ElasticTTT: Prior-Preserving Test-Time Tuning for Video Editing

Yueyi Liu[1*] Chi Zhang[1*] Sen Cui[2] Miao Liu[1†]
College of AI, Tsinghua University[1],
Beijing Academy of Artificial Intelligence[2]

Project Page Code Dataset

**Abstract.**
Test-Time Tuning (TTT) on pretrained diffusion models has emerged as a powerful paradigm for video editing. However, there exists a foundational mismatch between the distribution-mapping nature of generative models and the single-point optimization of standard TTT. In this paper, we demonstrate that this mismatch triggers *Prior Collapse*, a degenerate state where the model discards the text conditions and spatial latents, collapsing generations to the source video, or entangling the features of distinct regions. To resolve this, we propose **ElasticTTT**, a novel framework that preserves the prior generative distribution and rescues generative elasticity. Specifically, we propose *Target Distribution Regularization* to prevent sharp memorization minima, *Contrastive CFG* to guide inference away from source biases, and *Asynchronous Noise Schedule* to preserve unedited regions. Extensive evaluations, supported by theoretical analysis, demonstrate that ElasticTTT successfully preserves the generative prior of the base model, achieving state-of-the-art performance on one-shot video editing.

## 1 Introduction

Advancements in denoising diffusion models [18, 62] have enabled significant progress in text-to-video generation [21, 32, 64, 74]. These powerful video generation backbones also open new opportunities for controllable video editing by leveraging learned generative priors [24, 50, 71]. However, user-provided source videos may fall outside the fixed training distribution of off-the-shelf models [2, 5, 17, 38, 70, 75, 78], particularly given the rapid emergence of new concepts across the internet, resulting in degraded editing performance. Recently, another line of research has explored Test-Time Tuning (TTT) [16, 24, 63, 65–67, 69], which aims to adapt the weights of off-the-shelf models to the domain of user-provided source videos.

Compared with video editing approaches that perform inference directly with pre-trained models, TTT is particularly well suited to challenging scenarios where the source video exhibits a large domain gap from the pretraining distribution. This is because TTT explicitly adapts the model to the specific content and dynamics of each source video, thereby improving the preservation of its appearance, structure, and motion. Besides serving as a practical paradigm for video editing,

[*] Equal contribution
[†] Corresponding author


**Figure 1.** Demonstration of ElasticTTT and the baseline TTT method on four different editing tasks. ElasticTTT obtains high quality editing results among all tasks, whereas the baseline method exhibits clear Conditioning Collapse and Spatial Entanglement.

TTT-based methods also function an effective data engine for synthesizing paired editing data to train large-scale feed-forward video editing models [70, 77].

Although TTT provides a natural mechanism for personalized user experiences [4, 12, 35, 58], it introduces an inherent tension with the diffusion process: TTT intentionally encourages instance-specific overfitting, while diffusion processes are inherently stochastic and intended to preserve distributional diversity. Notably, optimizing a highly parameterized diffusion model exclusively on this zero-variance singularity can easily lead to a failure mode that we term *prior collapse*. As illustrated in Figure 1, prior collapse manifests itself in two complementary failure phenomena. *Conditioning Collapse* arises when the conditional guidance becomes anchored to the source-conditioned dynamics during tuning, preventing effective semantic control and causing the model to disregard the target editing instruction. *Spatial Entanglement* arises when spatial representations become globally coupled during denoising, preventing localized control and causing unintended modifications outside the target region.

To circumvent these challenges, previous methods have proposed a spectrum of interventions. To mitigate Conditioning Collapse, prior works often rely on low-rank adaptation (LoRA) [13], prior-preservation losses [58], self-supervised objectives [66], or the isolated optimization of text and null-text embeddings [12, 49]. To resolve Spatial Entanglement, methods typically resort to invasive attention map modifications [36, 79], masked optimization phases [13], or the injection of external control signals like depth and optical flow [67]. However, these approaches are fundamentally sub-optimal, their complexity determines that they heavily rely on delicate architectural tuning, require auxiliary datasets, or remain dangerously sensitive to hyperparameter choices.

More importantly, while Test-Time Tuning (TTT) has shown immense promise in image editing by fine-tuning models directly on the source image, adapting TTT to video is remarkably scarce and fraught with difficulty. Video Diffusion Transformers [53, 64] possess highly complicated spatial-temporal priors, and naively transferring image-based TTT optimization techniques to video models

severely accelerates Prior Collapse. The network quickly overfits to the deterministic motion and appearance of the source video, losing its generative elasticity required for precise, accurate editing.

To resolve this fundamental optimization–sampling tension, we propose **ElasticTTT**, a principled framework designed to rescue generative elasticity and explicitly cure the symptoms of Prior Collapse. Rather than treating these failures as isolated artifacts, ElasticTTT intervenes across the entire generative pipeline: First, to prevent Conditioning Collapse during optimization, we introduce *Target Distribution Regularization*. By softly relaxing the rigid, single-point target with controlled stochasticity, we physically prevent the model from memorizing the source video, preserving its elastic denoising priors. However, as the tuned weights still exhibit a residual shift toward the source concept, we subsequently apply *Contrastive Classifier-Free Guidance* during inference to forcefully steer the trajectory toward the target condition. Finally, to resolve Spatial Entanglement, we introduce an Asynchronous Noise Schedule in the sampling process, which dynamically assign independent diffusion trajectories to the edited and anchored regions, clarifying latent representations and granting the model the localized freedom to render new concepts while safely preserving the unedited surroundings.

To validate the effectiveness of our framework, we conduct extensive experiments across a diverse range of editing tasks. Our results demonstrate that ElasticTTT successfully resolves the prior collapse issue, exhibiting robust generative elasticity, precise instruction adherence, and high-fidelity preservation of the source video. In both quantitative and qualitative evaluations, ElasticTTT consistently outperforms competitive baselines—spanning inference-only, test-time-tuning, and training-based approaches—ultimately achieving state-of-the-art performance in video editing.

## 2 Related work

### 2.1 Test-Time Tuning of Diffusion Models

Test-Time Tuning (TTT) [63, 65] updates a pre-trained diffusion model at inference time for tasks like customized and personalized generation [4, 12, 35, 58], stylized generation [8, 59, 68, 76], and test-time scaling [45]. For generative editing, TTT establishes a high-fidelity anchor by reconstructing a specific reference video [69] for subsequent modifications, but optimizing the model on a single reference naively often degenerates performance [58]. Previous works attempt to mitigate this issue via low-rank adaptation [12, 13], additional prior-preserving losses [58], only optimizing text embedding [12] or null-text embeddings [49], or introducing self-supervised signals [66]. However, these interventions do not fully resolve the issue, especially on highly parameterized text-to-video models [64], often resulting in a zero-sum compromise between following novel editing instructions and preserving details of the source video.

### 2.2 Video Editing

Video editing requires seamlessly integrating new semantic concepts or modifying existing features while preserving the source video's unedited structures. Existing methods generally fall into three paradigms. *Training-based methods* [2, 5, 17, 38, 70, 71, 75, 78] fine-tune video generation models on paired datasets, but suffer from severe data scarcity and limited generalization on new concepts, and heavy computational resources needed for training. *Training-free methods* [1, 10, 15, 26, 33, 34, 42, 72] bypass data requirements by leveraging pre-trained priors via inverse diffusion methods [47, 60], often including manipulating attention maps [36, 42, 48] or utilizing pretrained image editing models [13, 33, 42, 52], but inherently struggle to balance semantic alignment and reference preservation due to fixed model parameters and sampling trajectory length. *Test-Time Tuning (TTT) methods* [16, 24, 43, 69] bridge this gap by tuning the base model directly on the source video, improving source reconstruction without involving external datasets. However, adopting TTT for video editing directly encounters the prior collapse issue, to which we offer a systematic solution in our work. More details of the

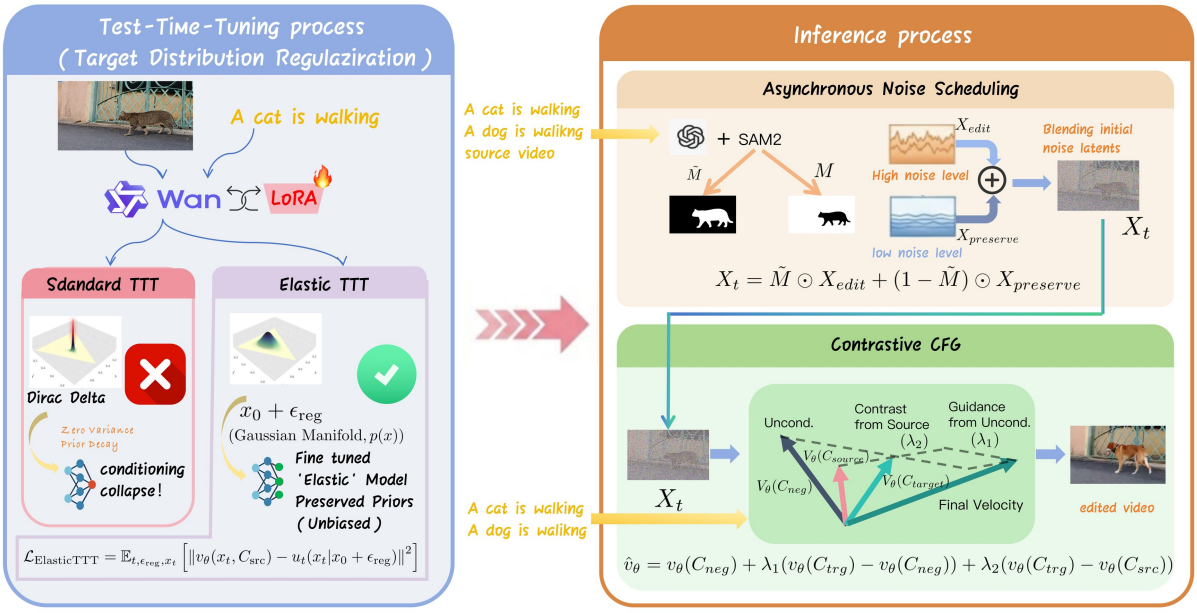


**Figure 2. The overall pipeline of ElasticTTT**. To resolve Conditioning Collapse, we soften the optimization target via TDR during training, and repel the sampling process from the source distribution via Contrastive CFG during sampling. To mitigate Spatial Entanglement, we perform Asynchronous Noise Scheduling to establish independent integration trajectories for distinct regions.

related works are shown in Supplementary.

## 3 Method

Given a source video $V_{\text{src}}$ and its corresponding descriptive text prompt $C_{\text{src}}$, the goal of video editing is to generate a target video $V_{\text{trg}}$ that aligns with a user-provided target prompt $C_{\text{trg}}$ while preserving the fundamental structural and temporal dynamics of $V_{\text{src}}$. Standard video editing models sample from a fixed conditional distribution $p_\theta(V_{\text{trg}} \mid V_{\text{src}}, C_{\text{trg}})$ with frozen parameters $\theta$, whereas test-time tuning approaches first adapt the model parameters on the source input via $\theta' = \mathcal{A}(\theta; V_{\text{src}}, C_{\text{src}})$, and then sample from the instance-adapted distribution $p_{\theta'}(V_{\text{trg}} \mid V_{\text{src}}, C_{\text{trg}})$, allowing stronger identity preservation and scene-specific consistency under the desired editing prompt.

Concretely, in the first stage, the model operates in a latent space and is optimized to reconstruct the source video latent $x_0$ by minimizing the MSE loss between the predicted velocity $v_\theta$ and the target conditional vector field $u_t$ given the noised latent $x_t$ at timestep $t$:

$$\mathcal{L}_{\text{TTT}} = \mathbb{E}_{t,x_t}\left[\|v_\theta(x_t, t, C_{\text{src}}) - u_t(x_t \mid x_0)\|^2\right] \tag{1}$$

Once the model is fine-tuned, inference is performed starting from a noisy latent, acquired by DDIM inversion [60], utilizing the target prompt $C_{\text{trg}}$ to guide the generation of $V_{\text{trg}}$.

### 3.1 The Optimization–Sampling Tension

While standard TTT captures the spatial-temporal features of the reference video, it introduces a fundamental mismatch with the generative nature of diffusion models. Diffusion models learn mappings between data distributions and rely on diverse samples to preserve generative flexibility. In TTT, however, the distribution of training data collapses to a single instance, effectively degenerating into a Dirac delta distribution, i.e., $p(x) = \delta(x - x_0)$.

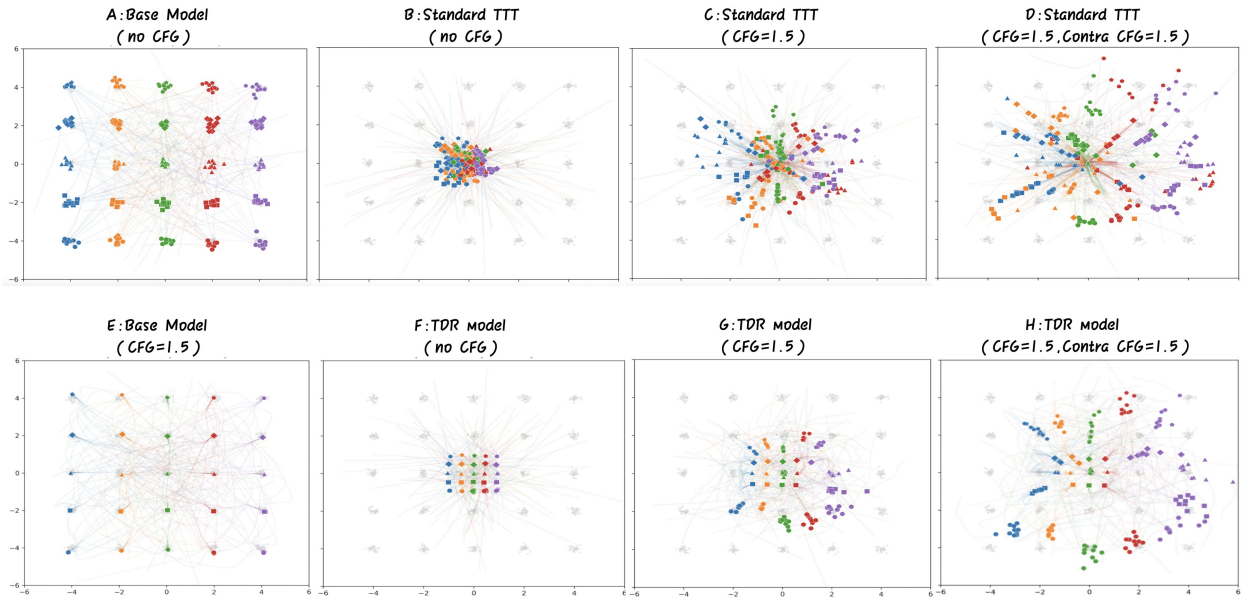


**Figure 3.** A 2D toy experiment, illustrating the effect of TDR and Contrastive CFG. In standard TTT, the generated distribution of the base model (A) collapses around the optimization target (B), and dispersing to chaos when CFG is applied (C, D). With TDR, the generated distribution remains its original structure (F), and Contrastive CFG further cancels out the introduced distribution shift (G, H).

Optimizing a highly parameterized diffusion model on such a single-point distribution can easily lead to a failure mode that we term ***prior collapse***: The model overfits to high-frequency pixel-level details of the source video, prioritizing exact reconstruction over generalizable condition-aware denoising. We conducted an illustrative experiment to examine this phenomenon, as shown in Figure 3: The base diffusion model is trained on a distribution with 25 classes (A), and TTT is performed on one sample of the center class. When we conduct inference with other class conditions, the model simply bypasses the conditions and generates a chaotic cluster around the memorized target distribution (B).

Formally, in video diffusion models, the network predicts the velocity field $v_\theta(x_t, t, C_{\text{src}})$ based on the noisy latent $x_t$ and the text condition $C_{\text{src}}$. Although the large pretrained model does not fully collapse the prior distribution, it still progressively learns to bypass meaningful variations in $x_t$ and $C_{\text{src}}$, instead hard-coding the high-frequency details of the source video into its parameters, degrading editing performance across two modalities:

- **Conditioning Collapse** By repeatedly updating on a static text prompt $C_{\text{src}}$, the model gradually weakens the text condition pathways. This causes the model to ignore the target prompt $C_{\text{trg}}$ during inference, losing its semantic elasticity, anchoring to the source concept rather than following new textual guidance (Figure 1 first row).
- **Spatial Entanglement** By increasingly bypassing the noisy input $x_t$ to memorize the global target, the model degrades its ability to extract semantically meaningful spatial representations from $x_t$. Consequently, distinct semantic objects lose their boundaries, causing severe visual corruption and entanglement when attempting to edit specific regions (Figure 1 second row).

Importantly, prior collapse is not equivalent to overfitting: TTT intentionally encourages controlled overfitting to capture source characteristics, while prior collapse represents a degenerate regime in which generative behavior collapses and editing controllability is lost.

To resolve this foundational mismatch, we propose **ElasticTTT**, a novel pipeline that preserves the

generative prior. We mitigate Conditioning Collapse during training by softening the optimization target into a continuous local distribution Section 3.2, and resolve the remaining conditioning bias induced through repelling the generation from the source condition through a contrastive classifier-free-guidance Section 3.3. We resolve Spatial Entanglement during inference by dynamically processing different regions with asynchronous noise levels Section 3.4. The following sections detail the formulations of these three methods.

### 3.2 Target Distribution Regularization: Escaping the Dirac Singularity

The root cause of Prior Collapse of the standard TTT lies in the rigid and deterministic nature of the Dirac delta distribution $p_{data}(x) = \delta(x - x_0)$ of the target data. To prevent the generative prior from collapsing into this sharp singularity, we introduce *Target Distribution Regularization* (TDR).

The core insight of TDR is to inject controlled stochasticity *strictly* into the optimization target, while keeping the input trajectory completely clean. Specifically, given a clean source video $x_0$ and a standard Gaussian noise sample $x_1 \sim \mathcal{N}(0, I)$, we construct the intermediate noisy latent $x_t$ exactly as in standard flow matching $x_t = tx_1 + (1-t)x_0$. However, when computing the target velocity for supervision, we replace the deterministic target $x_0$ with a perturbed version $\tilde{x}_0$:

$$\tilde{x}_0 = x_0 + \epsilon_{\text{reg}}, \quad \epsilon_{\text{reg}} \sim \mathcal{N}(0, \sigma_{\text{reg}}^2 I) \tag{2}$$

We then have the following updated optimization target $u_t(x_t|\tilde{x}_0) = x_1 - \tilde{x}_0$. Note that the model input $x_t$ is interpolated using the original, unperturbed $x_0$ and is completely independent of $\epsilon_{\text{reg}}$.

Crucially, this stochastic injection does not alter the fundamental optimization objective of the model. In the original flow-matching framework [41], the network is trained to predict the conditional expectation of the velocity field:

$$v^*(x_t, t) = \arg\min_v \mathbb{E}_{t,x_0,x_1}\left[\|v(x_t, t) - u_t(x_t \mid x_0)\|^2\right] = \mathbb{E}_{x_0|x_t}\left[u_t(x_t \mid x_0)\right] \tag{3}$$

Because standard flow matching paths construct the velocity field to be linear with respect to the target data (i.e. , $u_t(x_t|\tilde{x}_0) = x_1 - \tilde{x}_0 = x_1 - x_0 - \epsilon_{\text{reg}} - x_t = u_t(x_t|x_0) - \epsilon_{\text{reg}}$), training the network $v_\theta$ to predict this perturbed velocity field causes the new optimal vector field $\tilde{v}^*(x_t, t)$ to evaluate to:

$$\tilde{v}^*(x_t, t) = \mathbb{E}_{x_0,\epsilon_{\text{reg}}|x_t}\left[u_t(x_t \mid \tilde{x}_0)\right] = \mathbb{E}_{x_0,\epsilon_{\text{reg}}|x_t}\left[u_t(x_t \mid x_0) - \epsilon_{\text{reg}}\right] \tag{4}$$

$$= \mathbb{E}_{x_0|x_t}\left[u_t(x_t \mid x_0)\right] - \mathbb{E}_{\epsilon_{\text{reg}}}\left[\epsilon_{\text{reg}}\right] = v^*(x_t, t) \tag{5}$$

The injected noise $\epsilon_{\text{reg}}$ is sampled independently with zero mean, ensuring that the perturbation remains unbiased in expectation. Therefore, the global minimum of our regularized objective remains perfectly aligned with the original video's true vector field.

As shown in the toy experiment in Figure 3, instead of generating a collapsed distribution (B), the generation of the TDR model preserves the structures of the prior distributions (F), highlighting the significance of the introduced stochasticity. Although the expected target remains identical, the variance $\sigma_{\text{reg}}^2$ fundamentally alters the optimization dynamics. By replacing the strict Dirac delta point with a continuous local Gaussian manifold, we introduce a geometry-aware variance penalty into the loss landscape. The model thus no longer minimizes the loss by lazily memorizing an infinitely sharp, rigid spatial mapping to $x_0$. Instead, it learns to retain its elastic properties as a denoising model and seeks a broader, generalized minimum. Consequently, the model robustly follows target text prompts even after extensive fine-tuning steps, dramatically increasing editing flexibility and stabilizing the TTT process. A more mathematical explanation is shown in Supplementary Section A.

### 3.3 Contrastive Classifier-Free-Guidance:

In Figure 3 (F), after TDR, although the structure of the prior distribution is preserved, the generated points are still drawn closer to the source distribution trained in TTT. The same phenomenon occurs

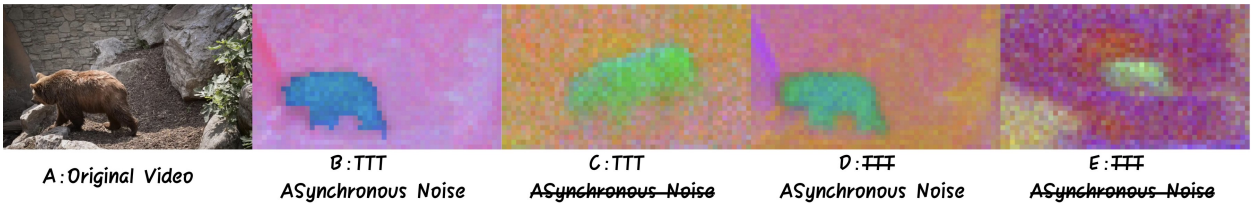


**Figure 4.** PCA [22] visualization of the DiT latent representations. When TTT is applied without Async-NS, the representation blurs and the model fails to capture the reference video's structure. Async-NS enables the model to extract clear, aligned representation

in video diffusion models: after TDR, the model no longer bypass $C_{\text{src}}$ and memorize the reference video entirely, yet the generated distribution still shifts toward the source distribution, showing a tendency of rendering the concepts introduced in the reference video.

To further cancel out the distribution shift and decouple the concepts of the source video which are planted into the model during training, we propose *Contrastive Classifier-Free-Guidance*, an enhanced Classifier-Free-Guidance (CFG) which actively utilizes the source condition as a negative constraint. The standard CFG [19] attempts to steer generation by creating a distribution shift between conditioned and unconditional predictions:

$$\hat{v}_\theta(x_t, t, C_{\text{trg}}) = v_\theta(x_t, t, C_{\text{neg}}) + \lambda \left(v_\theta(x_t, t, C_{\text{trg}}) - v_\theta(x_t, t, C_{\text{neg}})\right) \tag{6}$$

where $C_{\text{neg}}$ is typically an empty or generic negative prompt, and $\lambda$ is the guidance scale. To further suppress the embedded optimization bias, we expand this guidance into a tri-directional formulation. We explicitly contrast the target against the source, establishing a semantic opposition between them, by introducing a secondary negative constraint leveraging the original source prompt $C_{\text{src}}$:

$$\begin{aligned}\hat{v}_\theta(x_t, t, C_{\text{trg}}) = v_\theta(x_t, t, C_{\text{neg}}) &+ \lambda_1 \left(v_\theta(x_t, t, C_{\text{trg}}) - v_\theta(x_t, t, C_{\text{neg}})\right) \\ &+ \lambda_2 \left(v_\theta(x_t, t, C_{\text{trg}}) - v_\theta(x_t, t, C_{\text{src}})\right)\end{aligned} \tag{7}$$

where $\lambda_1$ controls the standard unconditional guidance and $\lambda_2$ governs the strength of the semantic contrast between the source and target conditions.

By contrasting the two conditions, Contrastive CFG isolates and amplifies the semantic delta between them, repelling the sampling destination away from the source distribution. As demonstrated in Figure 3, applying standard CFG (G)—which merely pulls the velocity toward the target condition—yields only a marginal shift in the generated distribution due to the residual gravitational pull of the memorized source. In contrast, our Contrastive CFG (H) actively repels the generation process from the source concept. This active contrast guarantees crisp semantic transitions and fully restores the model's generative elasticity.

### 3.4 Asynchronous Noise Scheduling

While TDR and Contrastive CFG cure Conditioning Collapse, the fine-tuned network remains vulnerable to Spatial Entanglement. Because the tuned network learns to bypass the noisy input $x_t$ and directly memorize $x_0$, it fails to extract localized, semantically rich spatial representations. As visualized in Figure 4 (C), during inference, the representation becomes blurred and misaligned with the source video, with the foreground and background heavily entangled.

Many existing approaches attempt to alleviate Spatial Entanglement by modifying attention maps [36, 79], using pre-edited images as guide [13, 33, 42] or intervening the optimization phase with masks [13, 66]. However, these complicated methods often compromise the global structural coherence. In contrast, we propose *Asynchronous Noise Scheduling* (Async-NS), an inference-time strategy that cleanly circumvents Spatial Entanglement by decoupling the integration trajectories of edited and anchored regions.

Our core insight is that temporal uniformity exacerbates Spatial Entanglement. Evaluating the entire latent at a single timestep $t$ allows the network to fall back on its memorized output. By explicitly assigning distinct noise levels and time embeddings to the edited versus anchored regions, we physically break the entanglement of representations. As evidenced in Figure 4 (B), this temporal asymmetry forces the network to process the regions distinctly, instantly restoring clear representational boundaries. Furthermore, this decoupling inherently aligns with the distinct generative needs of the regions: the edited area receives a high-noise regime ($T_\mathrm{e}$) for the flexibility to render new concepts, while the preserved region is restricted to a low-noise regime ($T_\mathrm{p}$) that carries the maximum information from the source video.

We define a frame-by-frame mask $M$, either from manual input for accuracy or from an automatic segmentation model for convenience, to mark the edited areas. $M$ is then downsampled to match the latent resolution and spatially smoothed into a continuous transition map $\tilde{M} \in [0,1]$ to naturalize the concept boundaries in latent space. We define two asynchronous noise schedules: $\{t_\mathrm{e}^i\}_{i=1}^N$ and $\{t_\mathrm{p}^i\}_{i=1}^N$, with $t_{(\cdot)}^N = T_{(\cdot)}$ and $t_{(\cdot)}^1 = 0$. The sampling starts from an initial latent $\hat{x}^N$, a noisy version of the source video latent with a mixed noise level, and at each step $i$, the asynchronous updates are spatially fused:

$$\begin{aligned} \hat{x}^N &= \tilde{M} \odot x_{T_\mathrm{e}} + (1-\tilde{M}) \odot x_{T_\mathrm{p}} \\ \hat{x}^{i-1} &= \tilde{M} \odot \left(\hat{x}^i - \hat{v}\Delta t_\mathrm{e}^i\right) + (1-\tilde{M}) \odot \left(\hat{x}^i - \hat{v}\Delta t_\mathrm{p}^i\right) \end{aligned} \tag{8}$$

where $\hat{v}$ denotes the final velocity by Contrastive CFG predicted with a spatially mixed timestep embedding of $t_\mathrm{e}^i$ and $t_\mathrm{p}^i$ as input:

$$\hat{v} = \hat{v}_\theta(\hat{x}^i, t^i, C_\mathrm{trg}), \quad t^i = \tilde{M} \odot t_\mathrm{e}^i + (1-\tilde{M}) \odot t_\mathrm{p}^i \tag{9}$$

By breaking the global temporal synchronization, Async-NS provides the model with a spatially clear, disentangled latent representation with distinct noise levels and timestep embeddings. Meanwhile, this formulation also grants the model localized freedom to aggressively overwrite target concepts along a longer integration path, while safely guiding unedited surroundings back to their exact physical state along a restricted, deterministic path.

# 4 Experiment

## 4.1 Experiment setup

To validate the effectiveness and robustness of ElasticTTT, we conducted extensive experiments and compared it with a wide range of video editing models. We utilize Wan2.1 1.3B [64] as the base model and conduct TTT with Low Rank Adaptation (LoRA) [23] for 100 steps. We set the TDR variance to $\sigma_\mathrm{reg} = 0.2$, the CFG guidance scales to $\lambda_1 = 6, \lambda_2 = 2$, and distinct noise regimes to $T_\mathrm{p} = 0.55, T_\mathrm{e} = 0.97$. The segmentation mask $M$ is automatically generated by a segmentation model Grounded-SAM2 [57] with a VLM Qwen3-VL-32B [3] to specify the edited concept. All experiments are done on a single Nvidia pro6000 GPU, where our full method introduces only a $\sim$7% inference-time overhead over vanilla TTT (approximately 5% from Contrastive CFG and 2% from Async-NS). More implementation details, along with a sensitivity analysis on these hyperparameters that verifies the robustness of our method, are provided in Supplementary Section B.

### 4.1.1 Evaluation Datasets

Prior works [15, 36, 39, 40, 48, 73] construct their testset by selecting text-video pairs from the Davis dataset [7, 54, 55]. However, either their data is not released [15, 27] or the selection contains only limited editing tasktypes [42]. Thus, we choose to select a testset following the common protocol on our own, including 125 text-video pairs. For comprehensive evaluation, our testset covers five

**Table 1. Quantitative comparison with prior video editing approaches.** Best results are highlighted in red, and second-best in blue. * refers that the method is re-implemented with our configurations.

| Methods | VBench↑ | CLIP-T↑ | VEBench↑ | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|---|---|---|
| *Other Methods* | | | | | | | |
| AnyV2V[33] | 4.80 | 30.21 | 0.53 | 3.51 | 5.89 | 2.30 | 4.31 |
| Ground-A-Video[26] | 4.89 | 28.51 | 0.13 | 3.57 | 4.49 | 2.42 | 3.90 |
| Token-flow[15] | 4.82 | 29.69 | 0.67 | 4.47 | 5.57 | 4.30 | 4.83 |
| Ditto [2] | 4.88 | 28.93 | 0.60 | 5.48 | 5.95 | 3.32 | 5.62 |
| Flow-align [30] | 4.93 | 27.72 | 0.61 | 5.28 | 5.44 | 7.23 | 5.44 |
| Uniedit [1] | 4.31 | 31.76 | 0.64 | 3.87 | 6.51 | 0.54 | 5.04 |
| *Test-Time-Tuning Methods* | | | | | | | |
| Tune-A-Video* [69] | 5.00 | 28.40 | 0.72 | 5.34 | 6.15 | 3.63 | 5.39 |
| VidTTA*[66] | 4.98 | 28.47 | 0.72 | 5.00 | 5.76 | 4.42 | 5.08 |
| ElasticTTT | 4.98 | 29.58 | 0.72 | 6.19 | 7.50 | 5.23 | 6.68 |

different types of tasks, specifically: subject editing, background editing, subject addition, color adjustment, and general editing. We define general editing as reconstructing all visual elements in the video while preserving only the general motion dynamics of the reference subject. This editing mode is particularly valuable when the goal is to retain the cinematographic composition (or camera movement) rather than the visual content. We will release our selection to promote fair and comprehensive benchmarking for future works.

#### 4.1.2 Prior Methods

We compare ElasticTTT against several strong video editing baselines: FlowAlign [30], UniEdit [1], Ditto [2], VidTTA [66], TokenFlow [15], AnyV2V [33], Tune-A-Video [69] and Ground-A-Video [26]. This selection represents a wide-range of conventional video editing paradigms, encompassing training-free methods [1, 15, 26, 30], first-frame-guided approaches [33], large-scale training-based models [2], and other Test-Time Tuning (TTT) methods [66, 69]. Notably, to ensure fair comparison with TTT-based baselines, we re-implement Tune-A-Video [69] and VidTTA [66] using the *exact same configurations* with ElasticTTT, including all settings of base model, training and inference.

#### 4.1.3 Evaluation Metrics

We evaluate ElasticTTT and the baseline models using a comprehensive suite of objective and subjective metrics.

*Automatic Metrics.* Following standard evaluation protocols [1, 2, 67, 69], we compute the CLIP similarity to measure semantic alignment between the edited video and the target prompt. In addition, we incorporate established metrics from video generation and editing benchmarks. We include metrics from VBench [25] to evaluate the overall visual quality and temporal consistency of the generated video. Specifically, we averaged 6 VBench metrics after normalizing them separately, to make a general assessment of the generated video. Also, we include the metric from VEBench [6] as a general indicator of video editing quality. These automatic metrics primarily serve as efficient and reproducible signals for model development, ablation analysis, and large-scale benchmarking.

*Judge-Based Evaluation.* Despite their efficiency, traditional quantitative metrics often fail to capture nuanced editing requirements, such as fine-grained preservation in unedited regions or complex instruction adherence. To address this limitation, we leverage the advanced visual reasoning

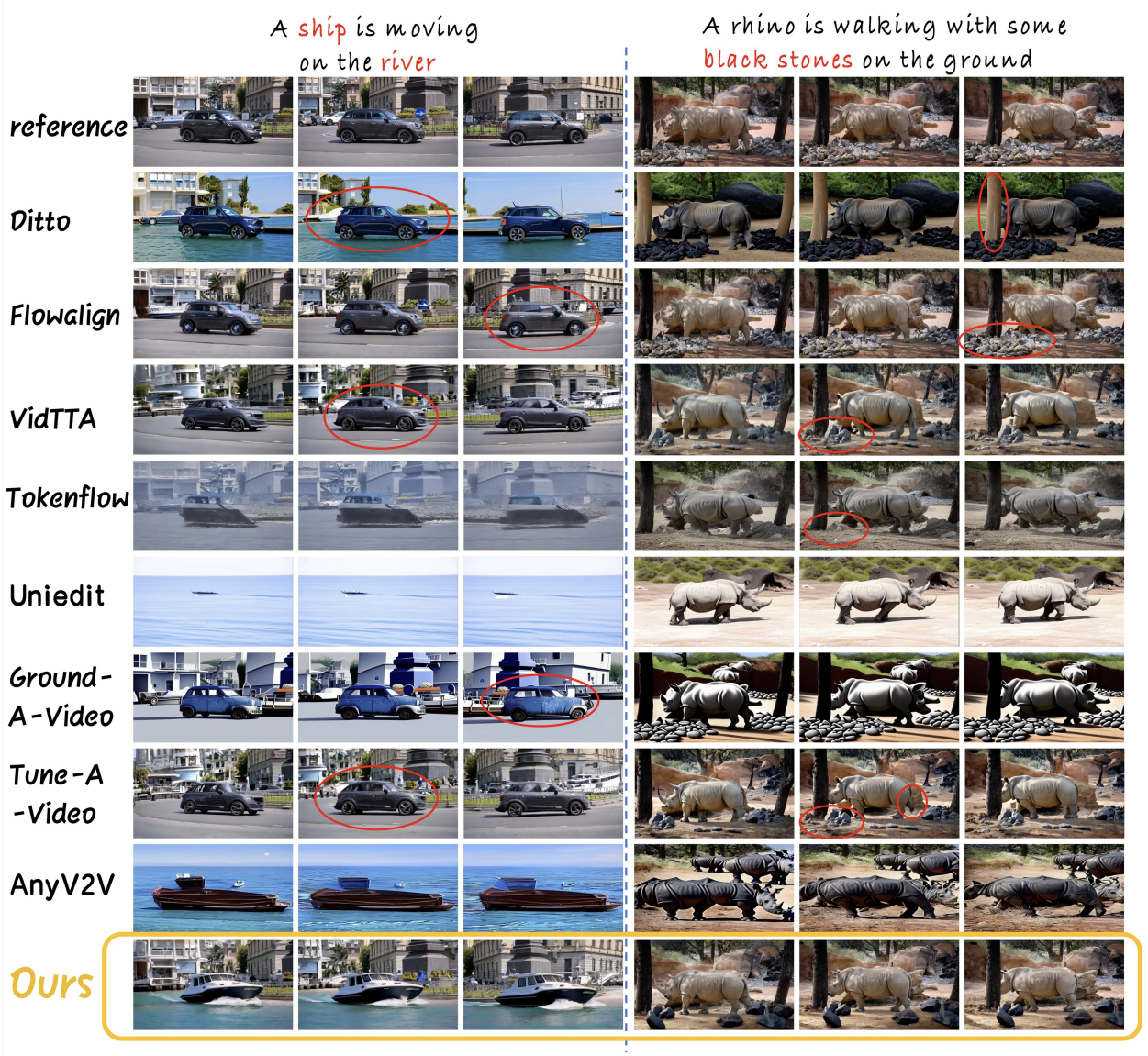


**Figure 5.** Visual comparisons with other methods. Our method strictly follows the editing instructions while successfully preserving the details in unedited regions.

capabilities of a pretrained large Vision-Language Model (VLM). Specifically, we prompt GPT-5 [51] to evaluate generated videos along four dimensions: (1) Video Quality (VQ) measures the quality of generated video, independent of the source video and editing instructions; (2) Instruction Adherence (IA) measures the instruction-following ability of the editing prompt; (3) Source Video Preservation (SP) measures the preservation of the source video in unedited areas; (4) an overall score (OVL), for general assessment.

*Human Study.* We further conduct a user study involving 10 participants, who independently rate each method along the same three dimensions: (1) Video Quality (VQ); (2) Instruction Adherence (IA); and (3) Source preservation (SP). We averaged the three scores to obtain an overall human evaluation result for each video. We randomly select one editing instruction for each video in our evaluation set to construct a subset for human evaluation. Participants are required to give a 1 to 5 rate on each criteria for each generated video. More details of evaluation metrics are shown in Supplementary Section C.

### 4.2 Comparison with Previous Methods

Table 1 summarizes the quantitative comparison between ElasticTTT and competitive baseline methods. In general, TTT-based methods outperform prevailing video editing approaches due to their ability to adapt to the specific content and temporal dynamics of each input video at test time. Moreover, ElasticTTT achieves the strongest performance among TTT-based methods, with particularly substantial gains on VLM-based evaluations, which provide a holistic and fine-grained assessment of video editing quality across multiple dimensions (VQ 6.19, IA 7.50, and OVL 6.68). Although certain baselines match or exceed ElasticTTT on isolated metrics, they suffer from significant trade-offs.

**Table 2. Human evaluation results.**Best results are highlighted in red, and second-best in blue.

| Methods | VQ ↑ | IA ↑ | SP ↑ | Avg.↑ |
|---|---|---|---|---|
| AnyV2V [33] | 2.50 | 2.61 | 2.48 | 2.54 |
| Ground-A-Video [26] | 1.95 | 2.15 | 2.37 | 2.16 |
| Token-flow [15] | 2.83 | 2.63 | 3.16 | 2.87 |
| Uniedit [1] | 2.36 | 2.94 | 2.08 | 2.46 |
| Ditto [2] | 3.85 | 3.20 | 3.07 | 3.37 |
| Flow-align [30] | 3.73 | 2.82 | 3.81 | 3.45 |
| ElasticTTT | 3.87 | 3.32 | 3.61 | 3.60 |

For example, Flow-align excels at preservation (SP 7.23) in unedited areas but struggles with editing elasticity and instruction adherence (IA 5.44). Conversely, Uniedit achieves strong semantic alignment with target instructions (evidenced by a high CLIP-T score of 31.76) but fails in preservation (SP 0.54), heavily distorting the original video. As shown in Figure 5, ElasticTTT achieves both precise editing adherence (ship and river, black stones) and source preservation (background building, rhino and trees), whereas other methods exhibit obvious compromises.

As shown in Table 2, the results of human evaluation provide strong support for our quantitative findings. ElasticTTT achieves the highest performance (Average 3.60), indicating a clear human preference over the baseline methods. It excels particularly in editing adherence (IA 3.32) and achieves the top score for video quality (VQ 3.87). Even in preservation, where Flow-align peaks, ElasticTTT maintains a strong second-best performance (SP 3.61) without the severe trade-offs observed in competing models. Additionally, a study provided in the Supplementary Section D reveals a strong alignment between human volunteer scores and VLM evaluations, demonstrating the rationality of our VLM-based evaluation protocol.

To examine whether ElasticTTT generalizes to larger base models, we further conduct the same experiment on Wan2.2-5B, comparing our full method against the vanilla TTT baseline under identical settings. As shown in Table 3, ElasticTTT yields consistent and even more pronounced improvements on the larger backbone: it raises the overall score from 4.91 to 7.00 (+2.09), surpassing the corresponding gain observed on Wan2.1-1.3B (+1.29, from 5.39 to 6.68). The improvement is most evident in instruction adherence (IA 5.09 → 7.75) and video quality (VQ 5.05 → 6.47), indicating that our target distribution regularization and Contrastive CFG effectively unlock the richer generative prior of the larger model, which would otherwise be suppressed by TTT memorization. Notably, Wan2.2-5B equipped with ElasticTTT achieves the best overall score among all configurations (OVL 7.00), suggesting that our method scales favorably with model capacity. We observe a mild decrease in source preservation (SP 4.40 → 3.68), which we attribute to the stronger editing elasticity restored in the larger model: it follows editing instructions more aggressively,

**Table 3. Generalization to a larger base model.**Best results are highlighted in red.

| Methods | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|
| Wan2.2-5B-baseTTT | 5.05 | 5.09 | 4.40 | 4.91 |
| Wan2.2-5B-ours | 6.47 | 7.75 | 3.68 | 7.00 |
| Wan2.1-1.3B-baseTTT | 5.34 | 6.15 | 3.63 | 5.39 |
| Wan2.1-1.3B-ours | 6.19 | 7.50 | 5.23 | 6.68 |

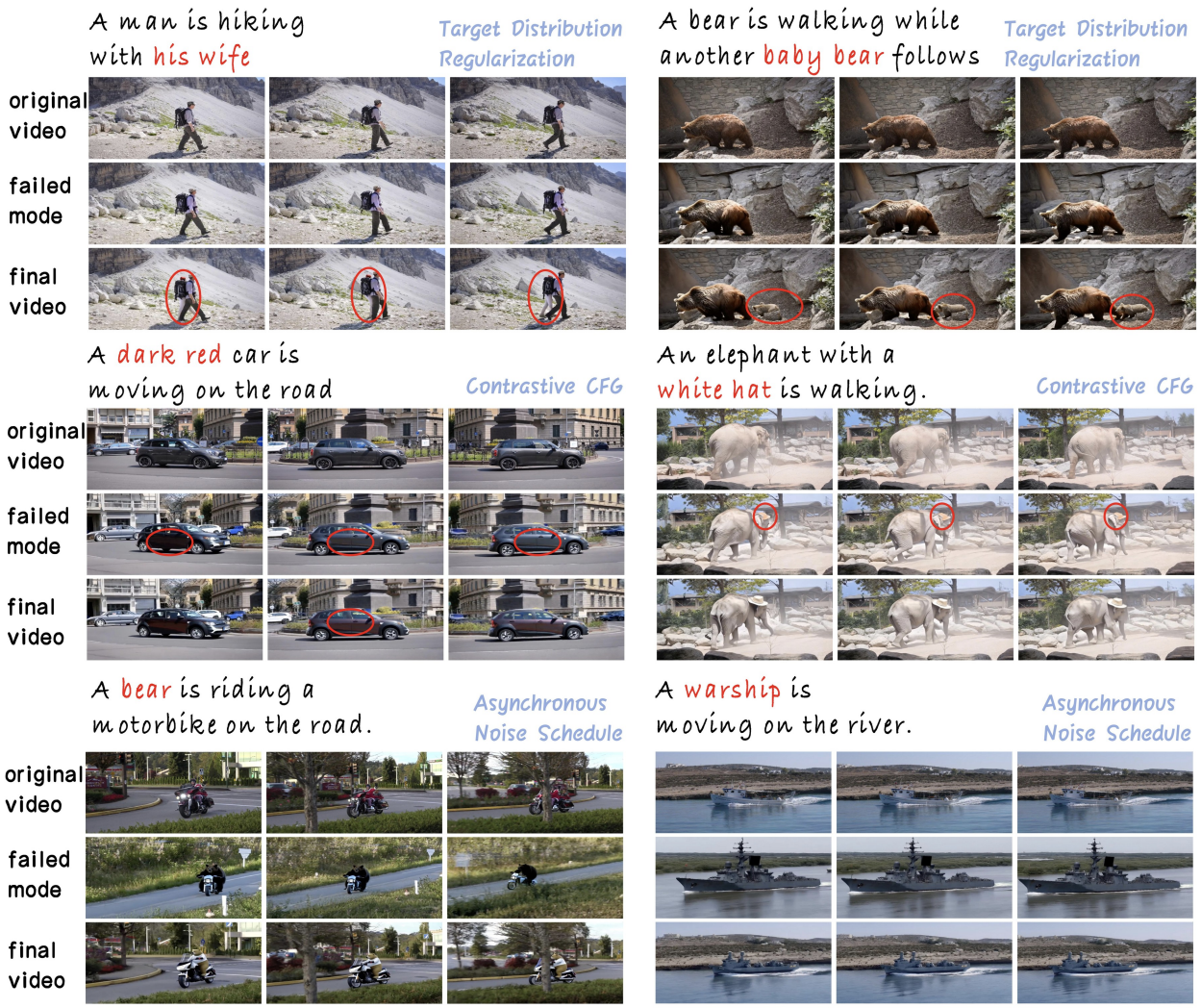


**Figure 6.** Visualized results of the ablation study.

trading marginal preservation for substantially better semantic alignment—a trade-off that clearly pays off in the overall quality.

### 4.3 Ablation Study

To assess individual contributions of our proposed designs, we conducted an ablation study systematically removing key components from the ElasticTTT pipeline. The results, presented both quantitatively in Table 4 and qualitatively in Figure 6, confirm that the full method achieves the optimal balance across metrics, and also exhibits the best performance in Figure 6.

Omitting TDR negatively affects video quality, editing adherence, and overall score. Although removing it marginally improves the preservation score, this comes at the expense of the model's ability to execute complex edits. As shown in the first row in Figure 6,

**Table 4. Quantitative results of the ablation study.** We exclude key components separately from the full model and report the resulting performance.Best results are highlighted in red.

| Methods | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|
| None (baseline) | 5.34 | 6.15 | 3.63 | 5.39 |
| w/o smooth transition | 6.07 | 7.37 | 5.09 | 6.56 |
| w/o Async-NS | 5.62 | 6.55 | 3.50 | 6.23 |
| w/o TDR | 6.13 | 7.45 | 5.36 | 6.64 |
| w/o contrastive CFG | 6.06 | 7.01 | 5.64 | 6.52 |
| Full (ElasticTTT) | 6.19 | 7.50 | 5.23 | 6.68 |

without TDR, the model merely reconstructs the source video and fails to add new concepts (wife, baby bear).

Removing Contrastive CFG leads to a noticeable decline in editing adherence and overall performance. Without this mechanism, the model struggles to fully follow the editing prompt, often resulting in an under-edited intermediate state between the reference and the target. As demonstrated in Figure 6 (second row), the absence of Contrastive CFG results in subtle, less distinguishable edits that lack the visual clarity of our full method.

Both Async-NS and the additional smooth transitions are critical for cohesive video generation. Removing Async-NS causes a drop in all quantitative metrics. Qualitatively, as seen in the third row of Figure 6, without Async-NS the model loses its ability to accurately isolate the target regions, incorrectly modifying the foreground and background simultaneously. Meanwhile, using smooth transition map $\tilde{M}$ instead of $M$ is verified to be beneficial, as it results in a minor but distinct improvement in the overall score.

We attribute this significant gap (OVL 6.68 vs. 4.71) to two primary factors. First, Async-NS introduces *asynchronous noise* inside the DiT forward process. By assigning different noise levels and timestep embeddings to the edited and preserved regions, it forms distinct regional representations during velocity prediction, thereby mitigating TTT-induced spatial entanglement. In contrast, post-hoc blending approaches only replace or mix latents after prediction, and thus cannot prevent entangled foreground/background features from being formed in the first place.

**Table 5. Comparison between Async-NS and re-implemented Latent Blending under identical settings.** Best results are highlighted in red.

| Methods | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|
| Latent Blending | 4.65 | 6.02 | 1.08 | 4.71 |
| Async-NS | 6.19 | 7.50 | 5.23 | 6.68 |

Second, such approaches rigidly hard-preserve the background, which proves counterproductive for realistic video editing: the unmasked region often requires soft physical adjustments, such as shadows, dust, boundary deformation, and contact effects. Forcibly stitching frozen background latents onto the edited content at mismatched noise levels produces visible seams and temporal artifacts, which explains the catastrophic preservation score of Latent Blending (SP 1.08) despite its seemingly conservative design. Async-NS instead suppresses spurious background drift while permitting these necessary interactions—a forward-pass desynchronization mechanism rather than a post-forward background replacement.

## 5 Conclusion

In this paper, we present **ElasticTTT**, a novel Test-Time Tuning based video editing framework. We first analyze the prior collapse phenomenon brought by the inherent tension between the rigid optimization target in TTT and the generative nature of diffusion models. By introducing three novel techniques: Target Distribution Regularization, Contrastive CFG, and Asynchronous Noise Scheduling, we resolve the Conditioning Collapse and Spatial Entanglement issues caused by prior collapse and achieve SOTA performance on quantitative results and human evaluation. Detailed ablation studies quantify the contribution of each key component in our framework. We believe our work not only represents an important step toward more reliable diffusion-based models, but also highlights promising directions for designing better personalized AI systems.

## References

[1] J. Bai, T. He, Y. Wang, J. Guo, H. Hu, Z. Liu, and J. Bian. Uniedit: A unified tuning-free framework for video motion and appearance editing, 2025.

[2] Q. Bai, Q. Wang, H. Ouyang, Y. Yu, H. Wang, W. Wang, K. L. Cheng, S. Ma, Y. Zeng, Z. Liu,

Y. Xu, Y. Shen, and Q. Chen. Scaling instruction-based video editing with a high-quality synthetic dataset. *arXiv preprint arXiv:2510.15742*, 2025.

[3] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[4] X. Bi, J. Lu, B. Liu, X. Cun, Y. Zhang, W. Li, and B. Xiao. Customttt: Motion and appearance customized video generation via test-time training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1871–1879, 2025.

[5] Y. Bian, Z. Zhang, X. Ju, M. Cao, L. Xie, Y. Shan, and Q. Xu. Videopainter: Any-length video inpainting and editing with plug-and-play context control, 2025.

[6] bingshuai liu, A. Wang, Z. Min, C. Lyu, L. Wang, and J. Su. VEBench: Towards comprehensive and automatic evaluation for text-guided video editing, 2025.

[7] S. Caelles, A. Montes, K.-K. Maninis, Y. Chen, L. Van Gool, F. Perazzi, and J. Pont-Tuset. The 2018 davis challenge on video object segmentation. *arXiv preprint arXiv:1803.00557*, 2018.

[8] J. Chung, S. Hyun, and J.-P. Heo. Style injection in diffusion: A training-free approach for adapting large-scale diffusion models for style transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8795–8805, 2024.

[9] J. Cohen. *Statistical power analysis for the behavioral sciences (2nd ed.* Statistical power analysis for the behavioral sciences, 1988.

[10] N. Cohen, V. Kulikov, M. Kleiner, I. Huberman-Spiegelglas, and T. Michaeli. Slicedit: Zero-shot video editing with text-to-image diffusion models using spatio-temporal slices. *arXiv preprint arXiv:2405.12211*, 2024.

[11] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

[12] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

[13] C. Gao, L. Ding, X. Cai, Z. Huang, Z. Wang, and T. Xue. Lora-edit: Controllable first-frame-guided video editing via mask-aware lora fine-tuning. *arXiv preprint arXiv:2506.10082*, 2025.

[14] C. F. Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. FA Perthes, 1877.

[15] M. Geyer, O. Bar-Tal, S. Bagon, and T. Dekel. Tokenflow: Consistent diffusion features for consistent video editing. In *The Twelfth International Conference on Learning Representations*, 2024.

[16] S. S. Harsha, A. Revanur, D. Agarwal, and S. Agrawal. Genvideo: One-shot target-image and shape aware video editing using t2i diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 7559–7568, June 2024.

[17] H. He, J. Wang, J. Zhang, Z. Xue, X. Bu, Q. Yang, S. Wen, and L. Xie. Openve-3m: A large-scale high-quality dataset for instruction-guided video editing. *arXiv preprint arXiv:2512.07826*, 2025.

[18] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[19] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[20] S. Hochreiter and J. Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

[21] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

[22] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

[23] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

[24] Y. Huang, W. Xiong, H. Zhang, C. Chen, J. Liu, M. Yan, and S. Chen. Dive: Taming dino for subject-driven video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16004–16014, 2025.

[25] Z. Huang, Y. He, J. Yu, F. Zhang, C. Si, Y. Jiang, Y. Zhang, T. Wu, Q. Jin, N. Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

[26] H. Jeong and J. C. Ye. Ground-a-video: Zero-shot grounded video editing using text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

[27] K. Kahatapitiya, A. Karjauv, D. Abati, F. Porikli, Y. M. Asano, and A. Habibian. Object-centric diffusion for efficient video editing. In *European Conference on Computer Vision*, pages 91–108. Springer, 2024.

[28] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[29] J. Kim, Y. Hong, J. Park, and J. C. Ye. Flowalign: Trajectory-regularized, inversion-free flow-based image editing. *arXiv preprint arXiv:2505.23145*, 2025.

[30] J. Kim, Y. Hong, J. Park, and J. C. Ye. Flowalign: Trajectory-regularized, inversion-free flow-based image editing. In *The Fourteenth International Conference on Learning Representations*, 2026.

[31] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[32] W. Kong, Q. Tian, Z. Zhang, R. Min, Z. Dai, J. Zhou, J. Xiong, X. Li, B. Wu, J. Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[33] M. Ku, C. Wei, W. Ren, H. Yang, and W. Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *Transactions on Machine Learning Research*, 2024. Reproducibility Certification.

[34] V. Kulikov, M. Kleiner, I. Huberman-Spiegelglas, and T. Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19721–19730, 2025.

[35] N. Kumari, B. Zhang, R. Zhang, E. Shechtman, and J.-Y. Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1931–1941, 2023.

[36] G. Kwon, J. Park, and J. C. Ye. Unified editing of panorama, 3d scenes, and videos through disentangled self-attention injection. *arXiv preprint arXiv:2405.16823*, 2024.

[37] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, F. Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[38] M. Li, L. Lin, Y. Liu, Y. Zhu, and Y. Li. Qffusion: Controllable portrait video editing via quadrant-grid attention learning. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[39] F. Liang, A. Kodaira, C. Xu, M. Tomizuka, K. Keutzer, and D. Marculescu. Looking backward: Streaming video-to-video translation with feature banks. In *The Thirteenth International Conference on Learning Representations*, 2025.

[40] F. Liang, B. Wu, J. Wang, L. Yu, K. Li, Y. Zhao, I. Misra, J.-B. Huang, P. Zhang, P. Vajda, et al. Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8207–8216, 2024.

[41] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

[42] C. Liu, R. Li, K. Zhang, Y. Lan, and D. Liu. Stablev2v: Stabilizing shape consistency in video-to-video editing. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[43] S. Liu, Y. Zhang, W. Li, Z. Lin, and J. 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.

[44] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[45] N. Ma, S. Tong, H. Jia, H. Hu, Y.-C. Su, M. Zhang, X. Yang, Y. Li, T. Jaakkola, X. Jia, et al. Inference-time scaling for diffusion models beyond scaling denoising steps. *arXiv preprint arXiv:2501.09732*, 2025.

[46] S. Mandt, M. D. Hoffman, and D. M. Blei. Stochastic gradient descent as approximate bayesian inference. *Journal of Machine Learning Research*, 18(134):1–35, 2017.

[47] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[48] T. H. S. Meral, H. Yesiltepe, C. Dunlop, and P. Yanardag. Motionflow: Attention-driven motion transfer in video diffusion models. *arXiv preprint arXiv:2412.05275*, 2024.

[49] R. Mokady, A. Hertz, K. Aberman, Y. Pritch, and D. Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6038–6047, 2023.

[50] C. Mou, M. Cao, X. Wang, Z. Zhang, Y. Shan, and J. Zhang. Revideo: Remake a video with motion and content control. *Advances in Neural Information Processing Systems*, 37:18481–18505, 2024.

[51] OpenAI. Gpt-5 system card, 2025.

[52] W. Ouyang, Y. Dong, L. Yang, J. Si, and X. Pan. I2vedit: First-frame-guided video editing via image-to-video diffusion models. *CoRR*, abs/2405.16537, 2024.

[53] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[54] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016.

[55] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[56] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[57] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[58] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.

[59] K. Sohn, N. Ruiz, K. Lee, D. C. Chin, I. Blok, H. Chang, J. Barber, L. Jiang, G. Entis, Y. Li, et al. Styledrop: Text-to-image generation in any style. *arXiv preprint arXiv:2306.00983*, 2023.

[60] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

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

[62] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[63] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020.

[64] Team Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang, J. Zeng, J. Wang, J. Zhang, J. Zhou, J. Wang, J. Chen, K. Zhu, K. Zhao, K. Yan, L. Huang, M. Feng, N. Zhang, P. Li, P. Wu, R. Chu, R. Feng, S. Zhang, S. Sun, T. Fang, T. Wang, T. Gui, T. Weng, T. Shen, W. Lin, W. Wang, W. Wang, W. Zhou, W. Wang, W. Shen, W. Yu, X. Shi, X. Huang, X. Xu, Y. Kou, Y. Lv, Y. Li, Y. Liu, Y. Wang, Y. Zhang, Y. Huang, Y. Li, Y. Wu, Y. Liu, Y. Pan, Y. Zheng, Y. Hong, Y. Shi, Y. Feng, Z. Jiang, Z. Han, Z.-F. Wu, and Z. Liu. Wan: Open and advanced large-scale video generative models, 2025.

[65] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

[66] J. Wang, Y. Chen, Y. He, X. Song, Y. Xin, D. Zhang, Z. Wan, B. Li, and R. Zhang. Low-cost test-time adaptation for robust video editing. *arXiv preprint arXiv:2507.21858*, 2025.

[67] Z. Wang and J. Tang. T3v2v: Test time training for domain adaptation in video-to-video editing. In *CVPR 2025 Workshop, AI for Creative Visual Content Generation Editing and Understanding*.

[68] Z. Wang, L. Zhao, and W. Xing. Stylediffusion: Controllable disentangled style transfer via diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7677–7689, 2023.

[69] J. Z. Wu, Y. Ge, X. Wang, S. W. Lei, Y. Gu, Y. Shi, W. Hsu, Y. Shan, X. Qie, and M. Z. 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.

[70] Y. Wu, L. Chen, R. Li, S. Wang, C. Xie, and L. Zhang. Insvie-1m: Effective instruction-based video editing with elaborate dataset construction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16692–16701, 2025.

[71] S. Yang, Z. Gu, L. Hou, X. Tao, P. Wan, X. Chen, and J. Liao. Mtv-inpaint: Multi-task long video inpainting. *CoRR*, abs/2503.11412, March 2025.

[72] X. Yang, L. Zhu, H. Fan, and Y. Yang. Eva: Zero-shot accurate attributes and multi-object video editing. *CoRR*, abs/2403.16111, 2024.

[73] X. Yang, L. Zhu, H. Fan, and Y. Yang. Videograin: Modulating space-time attention for multi-grained video editing. In *The Thirteenth International Conference on Learning Representations*, 2025.

[74] Z. Yang, J. Teng, W. Zheng, M. Ding, S. Huang, J. Xu, Y. Yang, W. Hong, X. Zhang, G. Feng, D. Yin, Yuxuan.Zhang, W. Wang, Y. Cheng, B. Xu, X. Gu, Y. Dong, and J. Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. In *The Thirteenth International Conference on Learning Representations*, 2025.

[75] S. Yu, D. Liu, Z. Ma, Y. Hong, Y. Zhou, H. Tan, J. Chai, and M. Bansal. Veggie: Instructional editing and reasoning video concepts with grounded generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15147–15158, 2025.

[76] Y. Zhang, N. Huang, F. Tang, H. Huang, C. Ma, W. Dong, and C. Xu. Inversion-based style transfer with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10146–10156, 2023.

[77] B. Zi and et al. Señorita-2m: A high-quality instruction-based dataset for general video editing by video specialists. *NeurIPS*, 2026.

[78] B. Zi, P. Ruan, M. Chen, X. Qi, S. Hao, S. Zhao, Y. Huang, B. Liang, R. Xiao, and K.-F. Wong. Se\˜ norita-2m: A high-quality instruction-based dataset for general video editing by video specialists. *arXiv preprint arXiv:2502.06734*, 2025.

[79] Z. Zuo, Z. Zhang, Y. Luo, Y. Zhao, H. Zhang, Y. Yang, and M. Wang. Cut-and-paste: Subject-driven video editing with attention control. *Neural Networks*, 181:106818, 2025.

This is the supplementary material for the paper "ElasticTTT:Prior-Preserving Test-Time Tuning for Video Editing". We organize the content as follows:



# A Mathematical Derivations

## A.1 Target Distribution Regularization

We demonstrate in Sec. 3.2 that Target Distribution Regularization (TDR) does not alter the optimization target of the diffusion model. However, it cannot fully explain why TDR preserves the general structure of the generated distribution. Here, we offer an analysis from the perspective of optimization dynamics. Specifically, we analyze the covariance of the stochastic gradients to demonstrate how standard TTT is prone to degenerate memorization, and how our proposed TDR implicitly injects a geometry-aware Gauss-Newton penalty that preserves the pre-trained generative prior.

### A.1.1 Setup and Standard TTT

Let the pre-trained network's prediction be denoted as $v_\theta(x_t, t, C) \in \mathbb{R}^d$, parameterized by $\theta \in \mathbb{R}^p$, where $x_t$ is the noisy latent, $t$ is the timestep, and $C$ represents the text conditioning. Let the true diffusion target be $u_t$. During standard TTT, the objective for a single stochastic step is the $L_2$ loss:

$$\mathcal{L}(\theta) = \frac{1}{2}\|v_\theta(x_t, t, C) - u_t\|^2 \tag{10}$$

Let the instantaneous error vector be $e_t(\theta) = v_\theta(x_t, t, C) - u_t \in \mathbb{R}^d$. We denote the Jacobian of the network output with respect to its parameters as $J_\theta = \nabla_\theta v_\theta(x_t, t, C) \in \mathbb{R}^{d\times p}$.

**Definition 1** (Stochastic Gradient and Covariance)**.** *The stochastic gradient for a single iteration of standard TTT is* $g(\theta) = \nabla_\theta \mathcal{L} = J_\theta^T e_t(\theta)$*. The expected (true) gradient is* $\bar{g} = \mathbb{E}_{t,\epsilon}[g(\theta)]$*. The covariance matrix of this stochastic gradient is defined as:*

$$\Sigma(\theta) = \mathbb{E}_{t,\epsilon}\left[(g(\theta) - \bar{g})(g(\theta) - \bar{g})^T\right] \tag{11}$$

**Proposition 1** (Prior Collapse of Standard TTT)**.** *Assume the highly overparameterized model is capable of perfect memorization of a target* $x_0$*. As the model perfectly overfits the spatial mapping, the instantaneous error vanishes* ($e_t(\theta) \to \mathbf{0}$)*, causing the gradient covariance to vanish:* $\Sigma(\theta) \to \mathbf{0}$*.*

*Proof.* By definition, $g(\theta) = J_\theta^T e_t(\theta)$. As the model perfectly fits the target distribution, $v_\theta(x_t, t, C) \to u_t$, yielding $e_t(\theta) \to \mathbf{0}$. Consequently, for all $t$ and $\epsilon$, $g(\theta) \to \mathbf{0} \Rightarrow \bar{g} \to \mathbf{0} \Rightarrow \Sigma(\theta) \to \mathbf{0}$. □

In the Stochastic Differential Equation (SDE) view of Stochastic Gradient Descent ($d\theta = -\eta\bar{g}dt + \sqrt{\eta\Sigma}dW$) [46], the term $\Sigma$ governs the exploration of the optimizer. When $\Sigma \to \mathbf{0}$, the optimizer loses all explorative ability and becomes irreversibly trapped in a sharp singularity where semantic conditioning $C$ and prior $x_t$ are bypassed in favor of rigid memorization.

### A.1.2 Target Distribution Regularization (TDR)

To prevent prior collapse, we introduce a zero-mean target regularization term $\epsilon_{\text{reg}} \sim \mathcal{N}(0, \sigma_{\text{reg}}^2 I)$. The regularized target becomes $\tilde{u}_t = u_t + \epsilon_{\text{reg}}$, while the input $x_t$ remains strictly unmodified.

**Proposition 2** (Unbiased Expected Gradient)**.** *The expected gradient under Target Distribution Regularization (TDR) remains strictly identical to the expected gradient of Standard TTT: $\bar{g}_{TDR} = \bar{g}$.*

*Proof.* The regularized stochastic gradient is given by:

$$g_{\text{TDR}}(\theta) = J_\theta^T(v_\theta(x_t, t, C) - (u_t + \epsilon_{\text{reg}})) = g(\theta) - J_\theta^T \epsilon_{\text{reg}} \tag{12}$$

Taking the expectation over $t, \epsilon$, and the independent noise $\epsilon_{\text{reg}}$:

$$\mathbb{E}[g_{\text{TDR}}(\theta)] = \mathbb{E}_{t,\epsilon}[g(\theta)] - \mathbb{E}_{t,\epsilon}\left[J_\theta^T\right] \mathbb{E}_{\epsilon_{\text{reg}}}[\epsilon_{\text{reg}}] \tag{13}$$

Since $\epsilon_{\text{reg}}$ is zero-mean, the second term vanishes, yielding $\bar{g}_{\text{TDR}} = \bar{g}$. □

**Proposition 3** (Geometry-Aware Covariance Injection)**.** *Under TDR, the gradient covariance matrix analytically decomposes into the standard covariance plus a Gauss-Newton penalty term proportional to $\sigma_{reg}^2$:*

$$\Sigma_{TDR}(\theta) = \Sigma(\theta) + \sigma_{reg}^2 \mathbb{E}_{t,\epsilon}\left[J_\theta^T J_\theta\right] \tag{14}$$

*Proof.* By definition, the new covariance is:

$$\Sigma_{\text{TDR}}(\theta) = \mathbb{E}\left[(g_{\text{TDR}}(\theta) - \bar{g})(g_{\text{TDR}}(\theta) - \bar{g})^T\right] \tag{15}$$

Substitute $g_{\text{TDR}}(\theta) = g(\theta) - J_\theta^T \epsilon_{\text{reg}}$ into the expectation:

$$\begin{aligned}\Sigma_{\text{TDR}}(\theta) &= \mathbb{E}\left[\left((g - \bar{g}) - J_\theta^T \epsilon_{\text{reg}}\right)\left((g - \bar{g}) - J_\theta^T \epsilon_{\text{reg}}\right)^T\right] \\ &= \mathbb{E}\left[(g - \bar{g})(g - \bar{g})^T\right] - \mathbb{E}\left[(g - \bar{g})\epsilon_{\text{reg}}^T J_\theta\right] \\ &\quad - \mathbb{E}\left[J_\theta^T \epsilon_{\text{reg}}(g - \bar{g})^T\right] + \mathbb{E}\left[J_\theta^T \epsilon_{\text{reg}} \epsilon_{\text{reg}}^T J_\theta\right]\end{aligned} \tag{16}$$

Because $\epsilon_{\text{reg}}$ is zero-mean and independent of the standard gradient components $g$ and $J_\theta$, the expectations of the two cross-terms evaluate strictly to zero:

$$\mathbb{E}\left[(g - \bar{g})\epsilon_{\text{reg}}^T J_\theta\right] = \mathbb{E}_{t,\epsilon}\left[(g - \bar{g})\right] \mathbb{E}_{\epsilon_{\text{reg}}}[\epsilon_{\text{reg}}^T] \mathbb{E}_{t,\epsilon}[J_\theta] = \mathbf{0} \tag{17}$$

Thus, the total expectation simplifies to the sum of the expectations of the two remaining terms. Recognizing that the first term is exactly the definition of the standard covariance $\Sigma(\theta)$:

$$\Sigma_{\text{TDR}}(\theta) = \Sigma(\theta) + \mathbb{E}_{t,\epsilon}\left[J_\theta^T \mathbb{E}_{\epsilon_{\text{reg}}}[\epsilon_{\text{reg}} \epsilon_{\text{reg}}^T] J_\theta\right] \tag{18}$$

Since the covariance of the injected noise is $\mathbb{E}[\epsilon_{\text{reg}} \epsilon_{\text{reg}}^T] = \sigma_{\text{reg}}^2 I$, the expression rigorously evaluates to:

$$\Sigma_{\text{TDR}}(\theta) = \Sigma(\theta) + \sigma_{\text{reg}}^2 \mathbb{E}_{t,\epsilon}\left[J_\theta^T J_\theta\right] \tag{19}$$

□

**Proposition 4** (Implicit Jacobian Penalty). *TDR injects optimizer noise proportional to the squared Frobenius norm of the Jacobian, mathematically measuring the "sharpness" [11, 20, 28] of the parameter space:*

$$Tr(\Sigma_{TDR}) = Tr(\Sigma) + \sigma_{reg}^2 \mathbb{E}_{t,\epsilon}\left[\|J_\theta\|_F^2\right] \tag{20}$$

*Proof.* Taking the Trace over both sides of Equation (19) yields. □

The term $\mathbb{E}[J_\theta^T J_\theta]$ is the uncentered covariance of the Jacobian, commonly recognized as the Gauss-Newton approximation [14] of the Hessian under $L_2$ losses. If the network attempts to form a degenerate, stiff spatial singularity (bypassing conditions and priors to perfectly memorize $x_0$), the Jacobian norm $\|J_\theta\|_F^2$ explodes.

Consequently, even as the empirical loss approaches zero ($e_t \to 0$ and $\Sigma \to \mathbf{0}$), the optimizer is subjected to persistent, geometry-aware noise: $\sigma_{\text{reg}}^2 \|J_\theta\|_F^2$. This exploding covariance violently ejects the optimizer from sharp minima via SDE Brownian motion. To converge, the network is physically forced to reach a parameter setting where $\|J_\theta\|_F^2$ remains small and stable. The original pre-trained generative prior, where the predicted velocity matches the ground truth across the entire noisy distribution, represents precisely this required flat configuration. Thus, TDR forbids prior collapse not by altering the global expected gradient, but by rendering collapsed states physically unreachable.

## A.2 Contrastive Classifier-Free Guidance

We claim in Sec. 3.3 that our Contrastive Classifier-Free Guidance (CFG) [19] actively repels the generated distribution from the source distribution, overcoming the distributional shift brought by the single-data optimization. This section provides a formal Bayesian analysis of this mechanism, providing a theoretical perspective. We formulate this mechanism as a dynamic, geometry-aware filter that mathematically neutralizes the distributional shift induced by TTT.

### A.2.1 The TTT Gravity Well

To elucidate the failure of standard CFG post-TTT, we analyze the optimization dynamics through the lens of Energy-Based Models (EBMs) [37, 61, 62]. In diffusion frameworks, the network estimates the score function [61] of a data distribution: $s_\theta(x_t, C) = \nabla_{x_t} \log p_\theta(x_t|C)$.

**Definition 2** (The Energy Landscape of TTT). *Let the pretrained base model describe an implicit probability distribution $p_{base}(x_t|C)$. TTT aggressively optimizes the network to reconstruct the source video trajectory $x_{src}$, effectively forcing the model to assign maximum likelihood to $x_{src}$ when conditioned on the source prompt $C_{src}$.*

**Proposition 5** (The Gradient Bias Field). *During prior collapse, TTT structurally shifts the base distribution by introducing a highly localized memorization peak. After TTT, the generated distribution shifts, and the score of the shifted distribution decomposes into the base score and an additive gradient bias field, $\nabla_{x_t} B(x_t)$.*

*Proof.* Because neural networks are universal function approximators, the overfitted probability distribution $p_{\theta^*}$ can be factored into the original pre-trained prior and a new, TTT-induced memorization peak:

$$p_{\theta^*}(x_t|C_{\text{src}}) = \frac{1}{Z} p_{\text{base}}(x_t|C_{\text{src}}) \cdot p_{\text{mem}}(x_t) \tag{21}$$

where $p_{\text{mem}}(x_t)$ approaches a Dirac delta function $\delta(x_t - x_{\text{src}})$ as the TTT loss converges, and $Z$ is a regularization term to guarantee $\int p_{\theta^*}(x_t|C_{\text{src}})\mathrm{d}x_t = 1$. The tuned score function is the log-likelihood

gradient of this factorization:

$$s_{\theta^*}(x_t, C_{\text{src}}) = \nabla_{x_t} \log p_{\text{base}}(x_t|C_{\text{src}}) + \nabla_{x_t} \log p_{\text{mem}}(x_t) \tag{22}$$

Defining the scalar field $B(x_t) = \log p_{\text{mem}}(x_t)$ yields the additive form:

$$s_{\theta^*}(x_t, C_{\text{src}}) = s_{\text{base}}(x_t, C_{\text{src}}) + \nabla_{x_t} B(x_t) \tag{23}$$

□

Mathematically, $B(x_t)$ acts as a deep, artificial energy well sculpted into the latent space. Its gradient, $\nabla_{x_t} B(x_t)$, acts as a spatial force vector aggressively pulling latents toward the source video $x_{\text{src}}$. We name this gradient bias $\nabla_{x_t} B(x_t)$ the *source attractor bias*.

### A.2.2 Semantic Leakage of the TTT Bias

Due to parameter sharing across conditions, this localized energy well unavoidably "bleeds" into other prompt conditions during inference.

**Definition 3** (Condition-Dependent TTT Bias)**.** *We define the tuned score function for an arbitrary prompt $C$ as the ideal base prior altered by a condition-dependent activation of the source attractor bias:*

$$s_{\theta^*}(x_t, C) = s_{base}(x_t, C) + \alpha(C)\nabla_{x_t} B(x_t) \tag{24}$$

*where $\alpha(C) \in [0, 1]$ parameterizes the activation strength.*

**Proposition 6** (Semantic Proportionality of Activation)**.** *The activation coefficient $\alpha(C)$ is directly correlated to the semantic similarity between the query prompt $C$ and the tuned source prompt $C_{src}$ within the text embedding space.*

**Justification.** During TTT, conditioning projections are updated specifically for the embedding of $C_{\text{src}}$. For a novel prompt $C$, the triggering of these updated weights depends on embedding alignment, well-approximated by cosine similarity: $\alpha(C) \approx \text{sim}(C, C_{\text{src}})$. This establishes three distinct activation regimes: **Source Prompt ($C_{\text{src}}$):** Similarity is maximized (sim = 1). Thus $\alpha(C_{\text{src}}) \approx 1$, fully activating the gravity well.
**Target Prompt ($C_{\text{trg}}$):** Target prompts share significant semantic overlap with the source (e.g., shared subjects). Hence, $0 < \alpha(C_{\text{trg}}) < 1$. This strict positivity inadvertently triggers a partial collapse into the TTT gravity well.
**Negative Prompt ($C_{\text{neg}}$):** Generic degradation prompts are semantically orthogonal to the source. Thus $\alpha(C_{\text{neg}}) \approx 0$, successfully bypassing the overfitted parameters.

### A.2.3 Active Cancellation via Contrastive CFG

To escape this gravity well and overcome the distributional shift, our Contrastive CFG defines the modified score as:

$$\begin{aligned} s_{\text{ContraCFG}}(x_t) = s_{\theta^*}(x_t, C_{\text{neg}}) &+ \lambda_1 \left(s_{\theta^*}(x_t, C_{\text{trg}}) - s_{\theta^*}(x_t, C_{\text{neg}})\right) \\ &+ \lambda_2 \left(s_{\theta^*}(x_t, C_{\text{trg}}) - s_{\theta^*}(x_t, C_{\text{src}})\right) \end{aligned} \tag{25}$$

**Proposition 7** (Standard CFG Amplifies TTT Bias)**.** *Standard Classifier-Free Guidance applied post-TTT inherently amplifies the source memorization, restricting semantic editing.*

*Proof.* Substituting Equation (24) into the standard CFG formula yields:

$$\begin{aligned} s_{\text{StdCFG}}(x_t) &= s_{\theta^*}(x_t, C_{\text{neg}}) + \lambda_1 \left(s_{\theta^*}(x_t, C_{\text{trg}}) - s_{\theta^*}(x_t, C_{\text{neg}})\right) \\ &= \underbrace{\Big\{ s_{\text{base}}(x_t, C_{\text{neg}}) + \lambda_1 \big(s_{\text{base}}(x_t, C_{\text{trg}}) - s_{\text{base}}(x_t, C_{\text{neg}})\big) \Big\}}_{s_{\text{base, StdCFG}}(x_t)} \\ &\quad + \Big\{ \alpha(C_{\text{neg}}) + \lambda_1 \big(\alpha(C_{\text{trg}}) - \alpha(C_{\text{neg}})\big) \Big\} \nabla_{x_t} B(x_t) \\ &\approx s_{\text{base, StdCFG}}(x_t) + \Big\{ 0 + \lambda_1 \big(\alpha(C_{\text{trg}}) - 0\big) \Big\} \nabla_{x_t} B(x_t) \\ &= s_{\text{base, StdCFG}}(x_t) + \lambda_1 \alpha(C_{\text{trg}}) \nabla_{x_t} B(x_t) \end{aligned} \tag{26}$$

Since $\alpha(C_{\text{neg}}) \approx 0$, the bias simplifies to $\lambda_1 \alpha(C_{\text{trg}}) \nabla_{x_t} B(x_t)$. Because $\lambda_1 \gg 1$, standard CFG aggressively scales the residual bias, misleading the generative trajectory toward the direction of the source attractor bias. □

**Proposition 8** (Contrastive Cancellation of TTT Bias)**.** *The contrastive guidance term injects a precisely scaled negative gradient that algebraically cancels the amplified TTT bias.*

*Proof.* Evaluating the contrastive guidance term (scaled by $\lambda_2$):

$$\begin{aligned} s_{\theta^*}(x_t, C_{\text{trg}}) - s_{\theta^*}(x_t, C_{\text{src}}) = &\left(s_{\text{base}}(x_t, C_{\text{trg}}) - s_{\text{base}}(x_t, C_{\text{src}})\right) \\ &+ \left(\alpha(C_{\text{trg}}) - \alpha(C_{\text{src}})\right) \nabla_{x_t} B(x_t) \end{aligned} \tag{27}$$

Because TTT strictly overfits $C_{\text{src}}$, we have $\alpha(C_{\text{src}}) > \alpha(C_{\text{trg}})$. Consequently, the bias scale is strictly negative. Let $\beta = \alpha(C_{\text{src}}) - \alpha(C_{\text{trg}}) > 0$, yielding:

$$s_{\theta^*}(x_t, C_{\text{trg}}) - s_{\theta^*}(x_t, C_{\text{src}}) = \left(s_{\text{base}}(x_t, C_{\text{trg}}) - s_{\text{base}}(x_t, C_{\text{src}})\right) - \beta \nabla_{x_t} B(x_t) \tag{28}$$

Substituting this directly back into the Contrastive CFG equation ( Equation (25)):

$$s_{\text{ContraCFG}}(x_t) = s_{\text{base, ContraCFG}}(x_t) + \left[\lambda_1 \alpha(C_{\text{trg}}) - \lambda_2 \beta\right] \nabla_{x_t} B(x_t) \tag{29}$$

□

While standard CFG burdens generation with a heavily amplified source-attractor bias, the contrastive term explicitly isolates the geometry of the TTT gravity well and subtracts it with magnitude $\lambda_2 \beta$. Through appropriate scaling of $\lambda_2$, the net bias coefficient $[\lambda_1 \alpha(C_{\text{trg}}) - \lambda_2 \beta]$ evaluates to zero, achieving exact cancellation of the TTT distributional shift, fully restoring semantic editing capabilities without degrading the structural fidelity

# B Details on Experiment Setup

## B.1 Implementation Details of Baseline Models

In our main experiment, we evaluate our proposed method, ElasticTTT, against recent state-of-the-art video editing approaches. To comprehensively assess performance, we categorize our baselines into two groups: those utilizing their official open-source configurations and those we re-implemented to guarantee a standardized comparison.

For several previous models—specifically AnyV2V [33], Ground-A-Video [26], Token-Flow [15], Ditto [2], and UniEdit [1]—we directly obtain results using their official open-source code and model weights. We strictly adhere to the default hyperparameters provided in their official implementations, keeping the output video lengths and spatial resolutions at their original settings without modification.

Regarding AnyV2V, the framework necessitates a dedicated image editing model to generate first-frame guidance; to maintain consistency with its original paper, we integrate InstructPix2Pix as the designated editor. However, a significant limitation of relying on these out-of-the-box methods is that their foundational settings, such as the chosen base models, vary drastically. This discrepancy makes direct comparisons between them and our method less conclusive.

To address this limitation and enable a truly fair comparison, we adopt select state-of-the-art training-free and test-time tuning editing methodologies and re-implement them on identical configurations to ElasticTTT. Specifically, we re-implement Flow-Align [29] to serve as a highly competitive training-free baseline that shares our exact experimental settings. Flow-Align [29] originally was a SOTA training-free image editing framework, where the methodology is model-independent and can be directly adapted to new architectures.

Although Tune-A-Video [69] and VidTTA [66] were originally designed for 3D-UNet architectures, their core contributions are architecture-agnostic and can be readily adapted to Diffusion Transformer (DiT) frameworks. To ensure an equitable evaluation, we port their primary mechanisms into the Wan2.1-1.3B [64] architecture. Furthermore, we align all key hyperparameters for these re-implemented models, such as the number of test-time tuning steps, learning rate, with those used in ElasticTTT to ensure fair comparison.

For all other evaluated baseline models, we strictly adhered to the default hyper-parameters provided in their official implementations.

### B.2 Implementation Details of ElasticTTT

Regarding our training configuration, we align our primary test-time tuning phase with the protocol established by LoRA-Edit [13], specifically fixing the number of optimization steps to 100. We utilized the AdamW [31, 44] optimizer with a learning rate set to 3e-5. We set the shift of the diffusion noise schedule to 3.0, and utilize 50 sampling steps during inference with an Euler sampler. All generated videos have 81 frames with 480p resolution, aligning with common protocol [64].

In our ablation study, the video samples demonstrating target distribution regularization were generated using 300 test-time tuning (TTT) steps, as we observed that this regularization mechanism performs more effectively with extended optimization. Conversely, the visual examples illustrating the contrastive CFG and AsyncNC ablation were obtained using 70 TTT steps. For the main visual comparisons against other state-of-the-art methods, all hyperparameters are kept identical to those used in our quantitative evaluation.

The methodology related hyperparameters introduced Sec.4.1 are determined empirically based on qualitative observations from a minimal validation set comprising only five video samples. Because we deliberately bypassed an exhaustive grid search or systematic hyperparameter optimization to save computational resources, the current configuration may be suboptimal. Consequently, we anticipate that our method possesses further headroom for performance improvement, which could be unlocked through a more rigorous hyperparameter search.

To further characterize how these hyperparameters affect performance, we evaluate the model across a diverse range of settings on our test set, as summarized in Table 6, with the default configuration shown in red.

Overall, ElasticTTT exhibits strong robustness to the TDR and CFG scales. Varying the TDR scale from 0.1 to 0.3 changes the overall score by less than 0.05 (6.68–6.73), while any non-zero setting outperforms disabling TDR entirely (OVL 6.59), confirming that the benefit stems from the regularization itself rather than a delicately tuned variance. Similarly, all tested CFG scale combinations stay within a narrow band (6.63–6.75), indicating that the contrastive guidance is effective across a wide range of strengths. Notably, two non-default settings (TDR scale 0.3 with OVL 6.73, and CFG scale 6-3 with OVL 6.75) slightly surpass our default configuration, which corroborates the aforementioned headroom left by our deliberately coarse hyperparameter selection.

**Table 6.** Overall scores (OVL) under different hyperparameter settings. The default configuration is shown in red.

| | | | | |
|---|---|---|---|---|
| **TDR scale** | 0 | 0.1 | 0.2 | 0.3 |
| **OVL** | 6.59 | 6.69 | 6.68 | 6.73 |
| **CFG scale** | 5-3 | 6-3 | 6-2 | 6-1 |
| **OVL** | 6.69 | 6.75 | 6.68 | 6.63 |
| **Async-NS** | 0.95-0.5 | 0.97-0.6 | 0.97-0.55 | 0.95-0.55 |
| **OVL** | 4.59 | 4.71 | 6.68 | 4.62 |

In contrast, the Async-NS noise regimes $(T_e, T_p)$ constitute the most sensitive hyperparameter: deviating from the default $(0.97, 0.55)$ causes the overall score to drop sharply from 6.68 to around 4.6. This is expected, as these two thresholds directly define the desynchronization between the edited and preserved regions—an overly high $T_p$ injects excessive noise into the regions to be preserved and corrupts the source structure, whereas a lower $T_e$ provides insufficient noise for the edited regions to break away from the memorized source content.

## C Details on Evaluation Metrics

**Table 7.** The elaborated results of six metrics on VBench [25]

| Methods | SC↑ | BC↑ | MS↑ | DD↑ | AQ↑ | IQ↑ |
|---|---|---|---|---|---|---|
| *Other Methods* | | | | | | |
| Flow-align | 0.900 | 0.910 | 0.971 | 0.960 | 0.519 | 67.056 |
| Ditto | 0.927 | 0.924 | 0.977 | 0.760 | 0.574 | 71.862 |
| Uniedit | 0.957 | 0.970 | 0.983 | 0.352 | 0.498 | 55.033 |
| AnyV2V | 0.850 | 0.906 | 0.937 | 0.928 | 0.526 | 65.215 |
| Ground-A-Video | 0.871 | 0.918 | 0.872 | 0.960 | 0.561 | 71.065 |
| Token-flow | 0.936 | 0.944 | 0.969 | 0.752 | 0.523 | 70.486 |
| *Test-Time-Tuning Methods* | | | | | | |
| Tune-A-Video | 0.937 | 0.924 | 0.973 | 0.960 | 0.530 | 67.886 |
| VidTTA | 0.931 | 0.917 | 0.972 | 0.960 | 0.526 | 67.900 |
| ElasticTTT | 0.934 | 0.920 | 0.971 | 0.969 | 0.534 | 66.554 |

### C.1 Automatic Metrics

We provide a detailed introduction to the automatic evaluation protocol adopted.

*CLIP-T.* To quantify the alignment between the generated visual content and the target text instructions, we measure the CLIP [56] text-image similarity (CLIP-T). This metric leverages the pre-trained CLIP [56] model to map both text and images into a shared latent space. For our evaluation, we uniformly sample 8 frames from each video sequence. The cross-modal similarity score is computed independently by calculating the cosine similarity between the embedding of each individual frame and the embedding of the edited prompt. The final reported CLIP-T value is the arithmetic mean of these 8 independent frame-level similarity scores.

*VEBench.* VEBench [6] is a subjective-aligned benchmark suite specifically designed for text-

driven video editing quality assessment. Traditional objective metrics often struggle to align with human perception; to overcome this, VEBench utilizes a dedicated multi-modal quality assessment network (VE-Bench QA). Rather than relying on generic vision-language models, this specialized network evaluates edited videos by simultaneously modeling three critical components: the inner connection and relevance between the source and edited videos (source-target relationship), the alignment between the edited video and the driving text prompt, and overall visual quality indicators such as aesthetics and distortion. To ensure strict reproducibility during our VEBench evaluation process, we fixed the random seed to 666.

*VBench.* VBench [25] is a comprehensive evaluation suite that dissects video generative performance into 16 hierarchical, disentangled, and fine-grained dimensions (such as temporal flickering, motion smoothness, subject consistency, and aesthetic quality). This allows for a highly objective and human-aligned assessment of video generation capabilities. In our experiments, we select six core aspects that are closely related to the editing task, and are directly measurable on data other than the standard VBench testset. Specifically, we calculate Subject Consistency (SC), Background Consistency (BC), Motion Smoothmess (MS), Dynamic Degree (DD), Aesthetic Quality (AQ), and Image Quality (IQ). Because the scoring scales and computational methodologies across these disentangled dimensions inherently vary, we systematically normalize the raw values of these six aspects. These normalized values are then summed to compute a single, unified overall editing score. A detailed breakdown of the specific scores across all evaluated VBench dimensions is provided in Table 7.

### C.2 Judge-Based Metrics

The above automatic metrics provide an overall rating of the generated videos. However, these metrics focus on the general information, and overlook fine-grained video details, as well as the subtle semantics in the editing prompt. To provide a nuanced, reasoning-based assessment of our editing results, we employ a Vision-Language Model (VLM) as an automated evaluator. Specifically, we utilize GPT-5 [51] to perform complex visual reasoning, allowing us to determine whether highly specific editing instructions were accurately executed. For this evaluation process, we uniformly sample six corresponding frames from both the original source video and the final edited video. These paired frame sequences are simultaneously fed into GPT-5. This uniform sampling strategy provides the VLM with a comprehensive temporal view of the spatial transformations between the unedited and edited states, without exceeding the model's context window.

The GPT-5 evaluation is systematically conducted across four distinct dimensions to ensure a thorough assessment. Instruction Alignment (IA) measures the accuracy with which the model executed the target textual edit, such as a local object swap or attribute modification. Source Preservation (SP) evaluates the model's ability to retain the unedited background, original subjects, and overall context of the source video without introducing unintended alterations. Visual Quality (VQ) assesses the perceptual fidelity of the edited frames, looking specifically for spatial artifacts or structural distortions introduced during the generative process. Finally, the Overall (OVL) dimension provides a holistic score that measures the overall editing performance, acting as a proxy for comprehensive human judgment. Notably, the VLM only assesses the video in one aspect per each call, guaranteeing that different evaluation dimensions are independently measured. The wordings of the custom reasoning prompts utilized for this VLM-based assessment are provided in Section G

## D Details on Human Evaluation

### D.1 Evaluation Setup

To conduct nuanced, human perceptual quality aligned evaluation, we conducted a comprehensive human evaluation study. To maintain a manageable cognitive load for our participants while ensuring

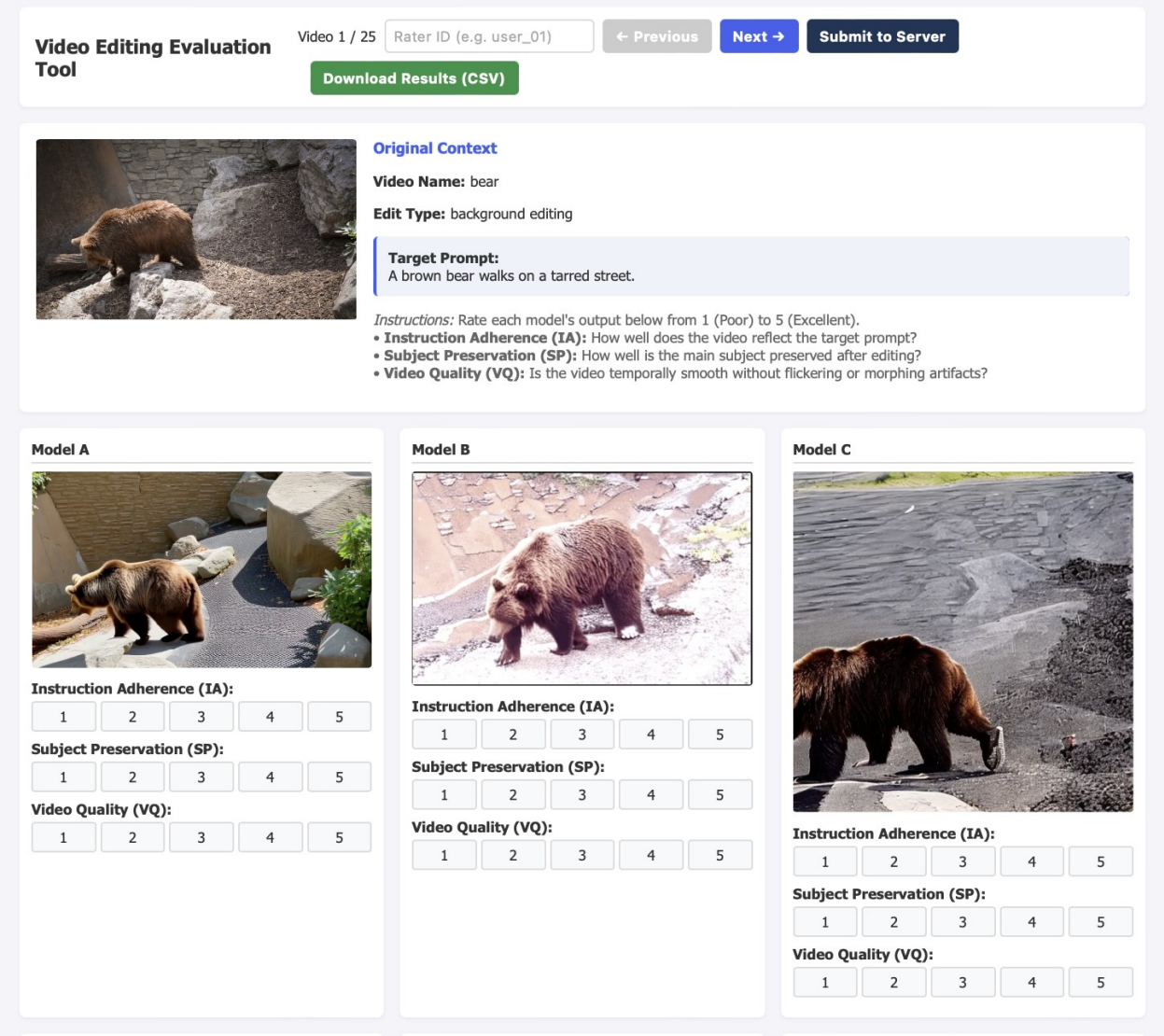


**Figure 7.** The user interface of our human evaluation website.

diverse coverage, we randomly selected a testset of 25 unique video-prompt combinations.

We invite 10 independent human evaluators in this assessment. For each of the 25 evaluation instances, participants are presented with the source video, the driving text prompt, and the edited videos generated by the evaluated methods. To ensure a strictly blind evaluation and mitigate any potential cognitive or brand bias, all model identities are completely anonymized, and the generated outputs are randomly shuffled and assigned temporary identifiers ranging from Model A to Model G for each individual trial.

The evaluators are tasked with rating each generated video on a range from 1 to 5, across three critical perceptual dimensions. Video Quality (VQ) assessed the overall temporal smoothness, aesthetic appeal, and absence of generative artifacts. Instruction Adherence (IA) measures how accurately the edited video reflected the specific semantic changes requested in the text prompt. Finally, Source Preservation (SP) evaluates the model's ability to retain the unedited background, subject identity, and structural layout of the original source video. A custom user interface is designed specifically for this evaluation platform, which facilitated simultaneous side-by-side comparisons, is illustrated in Figure 7.

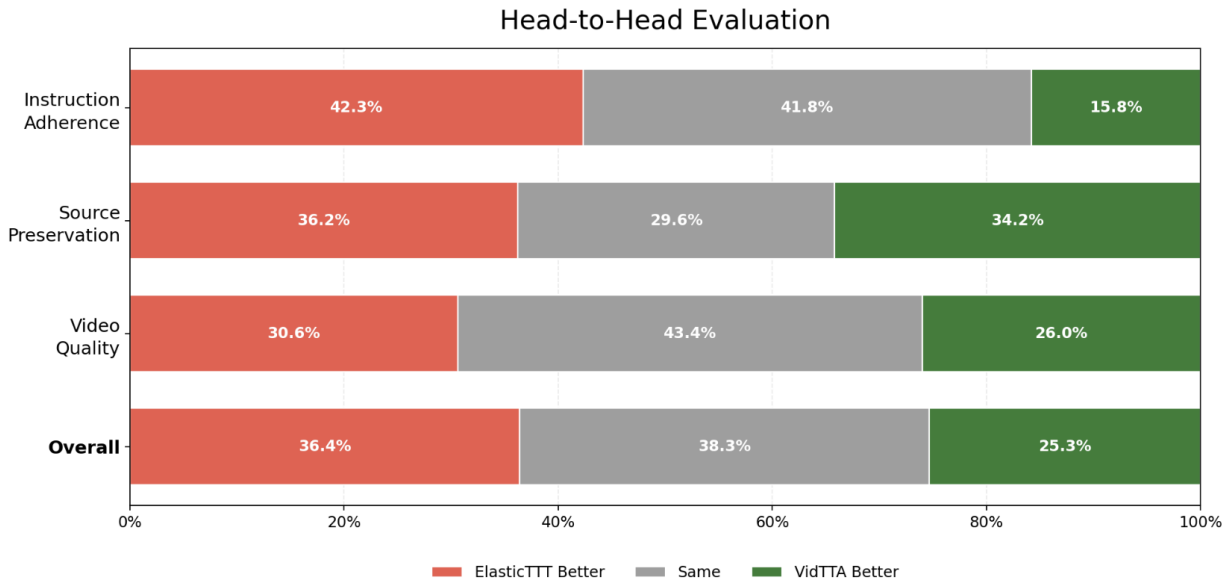


**Figure 8.** Head-to-head comparison results with the re-implemented VidTTA [66].

## D.2 Head-to-head Human Evaluation with VidTTA

As a highly competitive Test-Time Tuning (TTT) baseline that shares an identical foundational configuration with our proposed method, our re-implemented version of VidTTA [66] demonstrates remarkably strong performance on standard video editing tasks. Because the core optimization mechanisms of VidTTA closely parallel those of ElasticTTT, and the generation results demonstrate similar patterns, we deliberately exclude it from the broader multi-model human evaluation discussed previously, and explicitly conduct a *head-to-head* human evaluation.

Evaluators are presented with a direct side-by-side comparison of videos generated by both ElasticTTT and VidTTA, conditioned on the same source video and driving text prompt. The positioning of the two models' outputs is entirely randomized for each trial, and their identities remained strictly anonymized. For every video pair, participants were asked to compare the generative outputs and cast a single vote indicating whether the first video or the second is better, or if both videos are of equal quality. These comparative judgments are made holistically, factoring in Video

Quality, Instruction Adherence, and Source Preservation.

The quantitative results are illustrated in Figure 8. The aggregated preference data clearly indicates that ElasticTTT consistently outperforms the VidTTA baseline across all evaluation metrics, securing a vastly higher win rate. Notably, ElasticTTT achieves a significant 42.3% win rate over 15.8% in Instruction Adherence, and 36.4% over 25.3% in Overall score. This significant margin confirms that ElasticTTT successfully overcomes the inherent limitations of test-time tuning, yielding visual outputs that are demonstrably preferred by human observers.

**Table 8. Quantitative comparison with prior video editing approaches with 70 TTT steps** Best results are highlighted in red, and second-best in blue. * refers that the method is re-implemented with our configurations.

| Methods | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|
| *Other Methods* | | | | |
| AnyV2V[33] | 3.51 | 5.89 | 2.30 | 4.31 |
| Ground-A-Video[26] | 3.57 | 4.49 | 2.42 | 3.90 |
| Token-flow[15] | 4.47 | 5.57 | 4.30 | 4.83 |
| Ditto [2] | 5.48 | 5.95 | 3.32 | 5.62 |
| Flow-align [30] | 5.28 | 5.44 | 7.23 | 5.44 |
| Uniedit [1] | 3.87 | 6.51 | 0.54 | 5.04 |
| *Test-Time-Tuning Methods* | | | | |
| Tune-A-Video* [69] | 5.75 | 6.67 | 3.62 | 5.80 |
| VidTTA*[66] | 5.16 | 6.06 | 3.86 | 5.27 |
| ElasticTTT | 6.08 | 7.02 | 4.68 | 6.68 |

# E More Evaluation Results

## E.1 Fewer Steps Evaluation

Although we choose 100 TTT-steps as our final evaluation settings, we additionally evaluate ElasticTTT with 70 TTT-steps, comparing with the same group of baselines. Results are illustrated in Table 8. Experiments have shown that our method can achieve state-of-the-art results but also able to maintain high level editing quality at smaller TTT steps.

## E.2 More TTT Steps with TDR

Because the efficacy of Target Distribution Regularization (TDR) is deeply intertwined with the optimization dynamics of the Test-Time Tuning (TTT) process, we empirically evaluate the impact of TDR across varying optimization lengths. Figure 9 illustrates the overall editing performance as a function of the number of TTT steps. As our theoretical analysis predicted, standard TTT exhibits a degradation in editing capability as the step count increases. In contrast, incorporating TDR consistently enhances overall editing performance across all evaluated TTT step counts, with especially significant improvement on large TTT steps, actively preventing the optimization process from collapsing.

To further quantify this phenomenon, we conduct an ablation study extending the optimization to an aggressive 300 TTT steps. As detailed in Table 9, removing TDR leads to a drastically compromised Video Quality (VQ), Instruction Adherence (IA) and Overall (OVL) score, demonstrating the severe conditioning collapse. The increase in Source Preservation illustrates the overfitting brought by TTT without the controlled stochasticity. The 300 steps experiment exhibits a largely more significant result than the ablation study in Sec. 4.3, as the model overfits step-by-step, exacerbating the prior collapse phenomenon gradually through training.

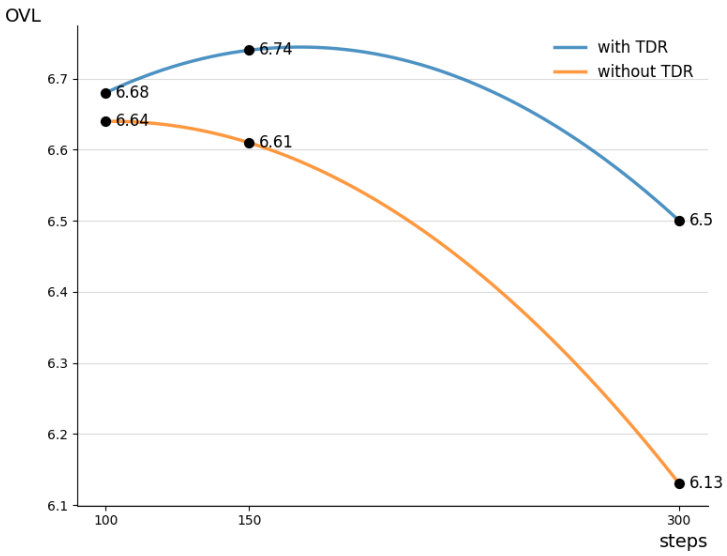


**Figure 9.** Comparison of Overall scores for different TTT steps, with and without TDR

**Table 9.** Quantitative results of TDR with 300 TTT steps.

| Methods | VQ↑ | IA↑ | SP↑ | OVL↑ |
|---|---|---|---|---|
| w/o TDR | 6.09 | 6.28 | 6.67 | 6.13 |
| Full (ElasticTTT) | 6.31 | 6.69 | 6.38 | 6.50 |

### E.3 Efficient Alignment-Editing Trade-off Via TDR

In sensitive applications, particularly those involving human faces, source preservation is a paramount concern; even minor structural alterations can result in severely unnatural artifacts or loss of identity. While adjusting the TTT step count serves as an effective mechanism to control the degree of editing and source preservation, an inherent trade-off exists: enforcing stricter alignment with the reference video inevitably degrades semantic alignment and overall editing quality. Incorporating our TDR module significantly mitigates this issue, enabling robust reference alignment with minimal semantic sacrifice. Empirically, when increasing the TTT steps from 100 to 300, employing TDR yields a highly efficient trade-off: a 1-point drop in OVL translates to a substantial 5-point gain in SP. In contrast, without TDR, the identical semantic cost yields a mere 2.7-point improvement in SP.

## F Correlation Analysis of VLM and Human Judgments

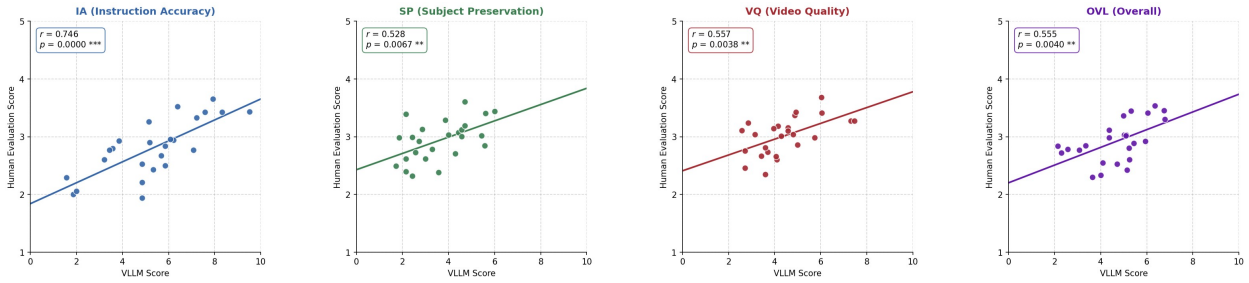


**Figure 10.** Correlation visualization between VLM score vs. Human Evaluation

**Table 10.** The $p$-values of two-sided Pearson correlation significance tests between human evaluation scores and VLM-given scores. Red represent **highly significant at the 0.01 level** and blue represent **statistically significant at the 0.05 level**.

| methods | VQ | IA | SP | OVL |
|---|---|---|---|---|
| Flow-align [29] | 0.0172 | 0.0001 | 0.0005 | 0.0025 |
| Ditto [2] | 0.0747 | 0.0000 | 0.0931 | 0.0000 |
| Uniedit [1] | 0.0116 | 0.0126 | 0.0006 | 0.0360 |
| AnyV2V [33] | 0.0544 | 0.0005 | 0.0000 | 0.0223 |
| Ground-A-Video [26] | 0.8595 | 0.0000 | 0.1202 | 0.3907 |
| Token-flow [15] | 0.0070 | 0.0000 | 0.0438 | 0.0000 |
| ElasticTTT | 0.0099 | 0.0220 | 0.0033 | 0.0038 |

**Table 11.** Pearson correlation coefficient ($r$) between VLM-given scores and human evaluation scores. Strong $r \geq 0.5$; Moderate $0.3 \leq r < 0.5$; Weak $r < 0.3$.

| Methods | VQ | IA | SP | OVL |
|---|---|---|---|---|
| Flow-Align [29] | 0.4041 | 0.7077 | 0.6310 | 0.5777 |
| Ditto [2] | 0.3084 | 0.8686 | 0.3146 | 0.7760 |
| UniEdit [1] | 0.5265 | 0.4945 | 0.6812 | 0.4219 |
| AnyV2V [33] | 0.3999 | 0.6388 | 0.7474 | 0.4595 |
| Ground-A-Video [26] | −0.1498 | 0.6103 | 0.2673 | 0.1795 |
| Token-Flow [15] | 0.5358 | 0.7119 | 0.3899 | 0.7405 |
| ElasticTTT | 0.5922 | 0.4245 | 0.5907 | 0.5915 |

To rigorously assess the agreement between our automated Vision-Language Model (VLM) evaluation and human perceptual judgments, we compute the Pearson correlation coefficient ($r$) across four core dimensions: Instruction Adherence (IA), Source Preservation (SP), Video Quality (VQ), and the Overall score (OVL). For a sample size of $n = 25$ video clips, the correlation coefficient between the VLM-assigned scores ($x_i$) and the human evaluation scores ($y_i$) is calculated as:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{30}$$

where $\bar{x}$ and $\bar{y}$ denote the respective sample means. This coefficient, bounded within $[-1, 1]$, quantifies the strength of the linear relationship between the two scoring paradigms.

To evaluate the statistical significance of these observed correlations, we formulate a two-sided hypothesis test. Let $\rho$ denote the true population correlation coefficient. We define our null hypothesis against the alternative hypothesis as follows:

$$H_0 : \rho = 0 \quad \text{vs.} \quad H_1 : \rho \neq 0 \tag{31}$$

To assess the evidence against $H_0$, which posits that any observed correlation is strictly due to random sampling variance, we compute the $t$-statistic:

$$t = r\sqrt{\frac{n-2}{1-r^2}} \tag{32}$$

Under the null hypothesis, this test statistic follows a Student's $t$-distribution with degrees of freedom $df = n - 2$. For our evaluation sample, $df = 23$. Based on this $t$-statistic, we calculate the corresponding two-sided $p$-value, representing the probability of observing a correlation as extreme as the computed $r$ entirely by chance (assuming $H_0$ holds). A sufficiently small $p$-value provides strong evidence to reject $H_0$ in favor of $H_1$, indicating a statistically significant linear relationship. The specific $p-values$ and significant levels are demonstrated in Table 10. The specific Pearson correlation coefficients for each of the seven evaluated models are reported in Table 11, with categorization thresholds adopted from Cohen's established guidelines [9].

Furthermore, we visualize this alignment in Figure 10. Each data point in the scatter plot represents an individual editing task, with its spatial coordinates corresponding to the mean VLM and human evaluation scores averaged across all seven models. In the figure caption, $r$ denotes the correlation coefficient and $p$ denotes the corresponding $p$-value for statistical significance testing. An asterisk (*) indicates that the correlation is statistically significant, typically with $p < 0.05$. More asterisks (*) represent a higher significance.

Ultimately, these quantitative analyses reveal a strong, positive, and statistically significant linear correlation between VLM scoring and human evaluation in the vast majority of cases. This robust alignment demonstrates that our VLM-based metrics serve as a highly reliable and scalable proxy for human visual judgment.

## G Prompt Design for VLM Judgment

To ensure the reproducibility of the results, we list the prompt for GPT-5 evaluation below:

```
"You will be given two frame sequences:
- Sequence A: original video frames
- Sequence B: edited video frames
You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based on the actual situation
in the video.

Scoring rubric:
1) Edit fulfillment (highest weight):
does edited video follow edited prompt vs original prompt?
2) Content preservation:
keep unrelated content/identity/background when they should remain.
3) Temporal and visual consistency:
stable motion, no obvious artifacts/flicker.

Return strict JSON only with keys:
{"score_0_10": number, "reason": string, "confidence": number}
Where score_0_10 in [0,10], confidence in [0,1], reason <= 80 words."
```

**Listing 1.** VLM prompt for Overall Score

```
"You will be given two frame sequences:
- Sequence A: original video frames
- Sequence B: edited video frames
You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video.
```

```
"You are scoring a edited video quality from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video."
        "Scoring rubric:"
        "1) image quality:
        the image from the edited video is of high quality,
        should be sharp and clear."
        "2) Aesthetic score:
        the edited video is aesthetically pleasing,
        the motion is smooth and natural."
        "3) Background consistency:
        the background is consistent with the original video,
        no obvious artifacts/flicker."
        "4) subject consistency:
        the object is consistent with the original video,
        no obvious artifacts/flicker."
        "5) motion smoothness:
        the motion is smooth and natural,
        no obvious artifacts/flicker."
        "Return strict JSON only with keys:"
        '{"score_0_10": number, "reason": string, "confidence": number}'
        "Where score_0_10 in [0,10], confidence in [0,1],
        reason <= 80 words."

Return strict JSON only with keys:
{"score_0_10": number, "reason": string, "confidence": number}
Where score_0_10 in [0,10], confidence in [0,1],
reason <= 80 words."
```

**Listing 2.** VLM prompt for Video Quality

```
"You will be given two frame sequences:
- Sequence A: original video frames
- Sequence B: edited video frames
You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video.

"You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video."
        "Scoring rubric:"
        "1) Semantic alignment:
        does edited video follow edited prompt vs original prompt?"

        "Return strict JSON only with keys:"
        '{"score_0_10": number, "reason": string, "confidence": number}'
        "Where score_0_10 in [0,10], confidence in [0,1],
        reason <= 80 words."
Return strict JSON only with keys:
{"score_0_10": number, "reason": string, "confidence": number}
Where score_0_10 in [0,10], confidence in [0,1], reason <= 80 words."
```

**Listing 3.** VLM prompt for Video

```
"You will be given two frame sequences:
- Sequence A: original video frames
- Sequence B: edited video frames
You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video.

"You are scoring a video editing result from 0 to 10.
Please assign scores more dispersedly based
on the actual situation in the video."
        "Scoring rubric:"
        "1) How much the details in the background
        (the place that should not be edited) are consistent
        with the original video?"

        '{"score_0_10": number, "reason": string, "confidence": number}'
        "Where score_0_10 in [0,10], confidence in [0,1],
        reason <= 80 words."
```

**Listing 4.** VLM prompt for Source Preservation

## H Demonstration of Asynchronous Noise Scheduling Masks

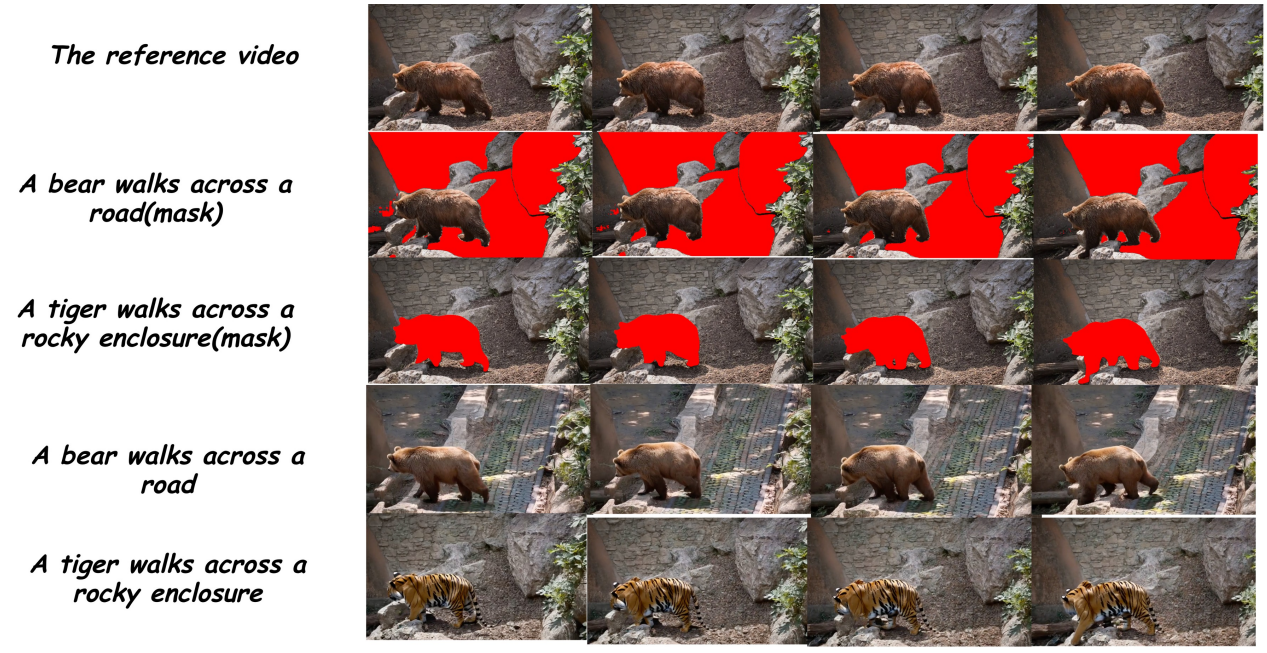


**Figure 11.** Visual illustration of our mask area and the final edited video.

In Figure 11, we illustrate the mask used for Asynchronous Noise Scheduling (Async-NS) generated by Grounded-SAM2 [57].

We showcase two different editing scenarios of the same reference video to further demonstrate the relationship between editing prompt and mask area. The second and third row show the masked areas and the fourth and fifth row are our final editing results.

# I Limitations and Future Work

## I.1 Limitation

Although our model can achieve state-of-the-art performance in all editing tasks, it still suffers from a few limitations. Firstly, the TTT nature of our model leads to more computation cost compared to those training-free methods. Secondly, in some situation, we found that over-specification of a certain area in the video can also limit the editing capabilities of other areas. We infer that this phenomenon may be caused by the over-optimization of the self-attention layers during the Test-Time-Tuning process.

## I.2 Future Work

Currently, our Asynchronous Noise Scheduling relies on explicit masks generated by external models like Grounded-SAM2. Future work could eliminate this dependency by dynamically extracting spatial masks directly from the diffusion model's internal cross-attention maps during the initial TTT steps, creating a fully end-to-end framework.

Also, we aim to extend ElasticTTT to handle long-context or multi-shot videos. By introducing temporal chunking or sliding-window TTT, we can ensure temporal consistency across hundreds of frames without exceeding standard VRAM constraints.

# J More Demonstrations

In Figure 12, we further show two challenging cases with increased motion and complex postures. In Figure 13, we demonstrate two more cases comparing to other video editing methods.

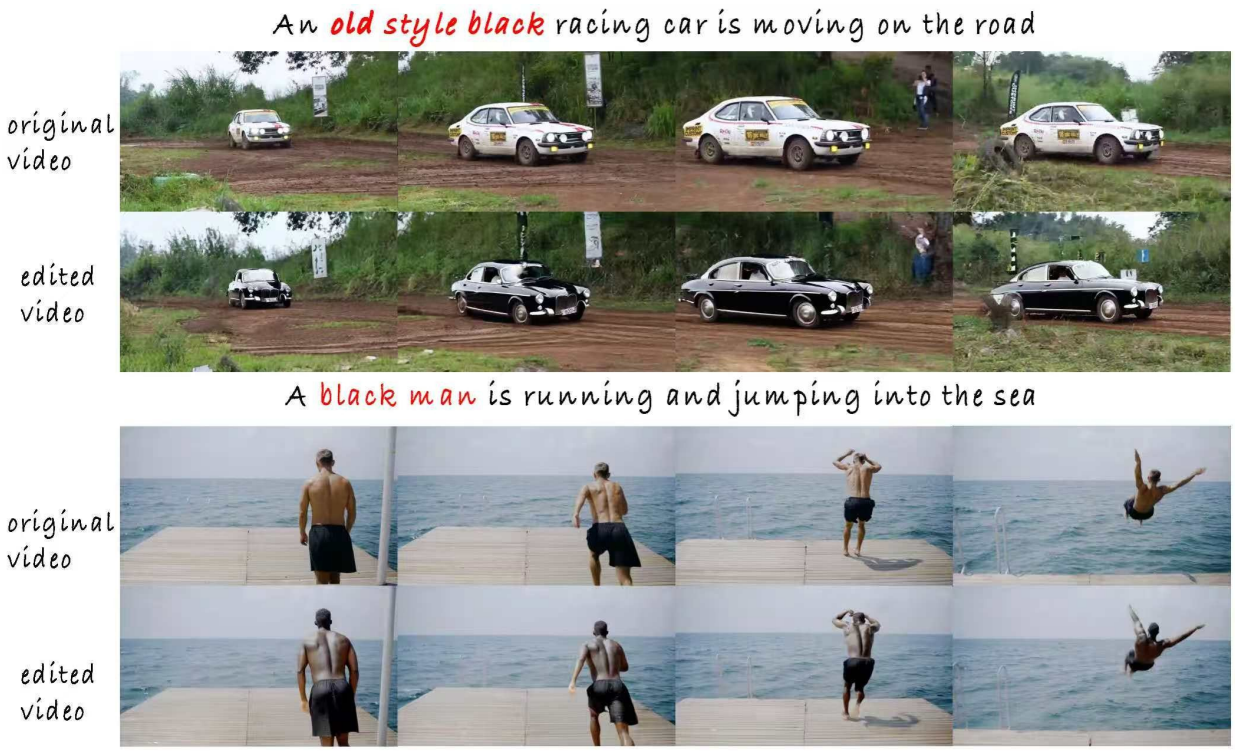


**Figure 12.** Two more challenging cases.

# K Social Impact and Ethical Concerns

## K.1 Social Impact

### K.1.1 Positive Social Impact

ElasticTTT serves as an accessible and highly effective framework that democratizes video editing for a broad user base. By enabling intuitive modifications solely through natural language instructions, it eliminates the steep learning curve traditionally associated with professional-grade software such as DaVinci Resolve. Furthermore, our model offers substantial value to professional creators by automating routine editing tasks, thereby significantly accelerating workflows and reducing both temporal and financial production costs.

### K.1.2 Potential Risk

Although our proposed ElasticTTT democratizes high-quality video editing, it inherently carries risks related to malicious video synthesis. The ability to precisely edit specific regions of a video could be misused to create convincing counterfeit content, such as placing individuals in fabricated scenarios without their consent. This contributes to the broader societal challenge of deepfakes and misinformation propagation. We strictly condemn the use of our method for generating deceptive content. To safeguard against such misuse, we suggest that downstream applications of our framework incorporate invisible watermarking and pair the technology with advanced synthetic media detection algorithms.

## K.2 Ethical Concerns for Human Evaluation

Our human evaluation framework is designed in strict compliance with ethical research standards. First, all evaluators engaged in the assessment strictly on a voluntary basis, retaining the unconditional right to opt out at any stage without facing any negative repercussions. Second, we enforced a rigorous data privacy policy; all collected feedback was entirely anonymized, ensuring that no personally

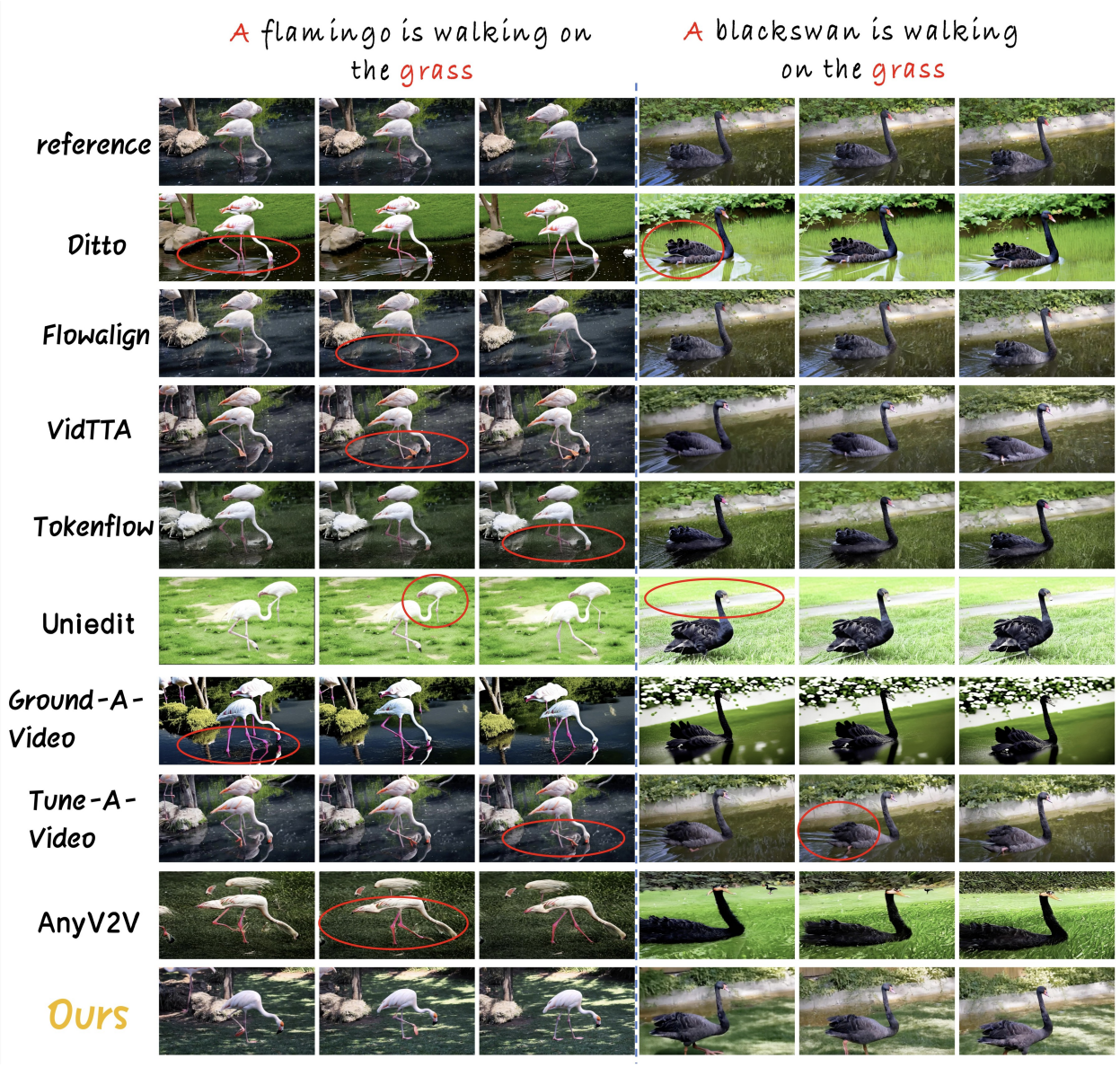


**Figure 13.** Visual comparisons with other methods. Our method strictly follows the editing instructions while successfully preserving the details in unedited regions.

identifiable details were ever recorded. Furthermore, the study is purely observational. The core task merely requires human subjects to review standard synthesized videos, which completely shields them from physical hazards, psychological stress, or any offensive visual stimuli. Consequently, our methodology avoids any invasive interventions that might compromise participant well-being.

### K.3 Safeguard

It is quite essential to implement proper safeguards when developing user-friendly video editing software or platform. To ensure a thoroughly vetted and secure assessment environment for our human evaluation, we strictly sampled videos from our curated dataset, which is constructed based on the renowned DAVIS benchmark. This deliberate selection process guarantees that all visual content is benign, thereby actively preventing the inadvertent inclusion of inappropriate, sensitive, or unvetted media commonly found on the internet.