TC-UAP: Towards Video Protection Against Generative AI

Yuxin Huang[1] **Mingming Gong**[2,5 †] **Wanyu Wang**[3] **Jing Zhang**[4] **Tongliang Liu**[1,5 †]
[1]Sydney AI Centre, The University of Sydney [2]University of Melbourne
[3]City University of Hong Kong [4]Wuhan University
[5]Mohamed bin Zayed University of Artificial Intelligence

Abstract

Recent diffusion-based video generation models have enabled high-quality personalized video customization through both *tuning-based pipelines*, which fine-tune a video diffusion model, and *reference-based pipelines* such as image-to-video generation. However, these capabilities raise serious concerns about personal privacy, identity ownership and intellectual property protection. Existing anti-customization works focus on protecting images, while protection for videos against both reference- and tuning-based customization remains largely underexplored. Protecting videos in this setting raises three challenges: (i) Image-level perturbations, optimized frame by frame, cannot survive temporal compression by 3D video VAE. (ii) A video-level perturbation optimized on a single video is vulnerable to temporal editing and fails to protect unseen videos. (iii) Temporally inconsistent perturbations are not robust to temporal attacks. To address these challenges, we propose Temporally Consistent Universal Adversarial Perturbations (TC-UAP), the first protection method against both reference- and tuning-based video customization. TC-UAP optimizes an identity-level multi-frame UAP over sliding windows from multiple videos, accounting for local temporal dependencies induced by temporal compression in video VAE and enabling a single perturbation to protect unseen videos of varying lengths. Moreover, we introduce intrinsic temporal modeling and an extrinsic surrogate temporal-attack loss, which make the perturbation temporally consistent and robust to unseen temporal attacks. Empirically, quantitative and qualitative results show that TC-UAP achieves the strongest identity protection compared with existing methods under both reference- and tuning-based video customization, and remains robust under multiple unseen temporal attacks. The code and additional results are available on our project page.

1 Introduction

Recent advances in diffusion models have revolutionized video creation, enabling the generation of high-quality videos. Both open-source [30, 81, 66, 17] and commercial models [58, 89] have shown increasingly impressive results. These models enable users to specify the subject of the generated video, a capability commonly known as video customization. Such customization typically follows two approaches. *Tuning-based methods* [1] fine-tune a video diffusion model on a small set of video clips, for example, using Low-Rank Adaptation (LoRA) [23], and can then generate diverse new videos. *Reference-based methods* [9, 16, 19, 82], such as image-to-video (I2V) models, take a reference image and a text prompt as input, offering pixel-level control.

However, these models raise serious concerns about personal privacy, identity ownership and intellectual property (IP) protection [28, 34, 33, 85, 22, 13, 71, 79] in video content. From public video clips of a target person, a malicious user can exploit current models to generate misleading or harmful content [49, 78, 87, 47, 40, 39] about that person through either approach: fine-tuning a video diffusion model on the whole set through tuning-based customization, or extracting a single

†Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

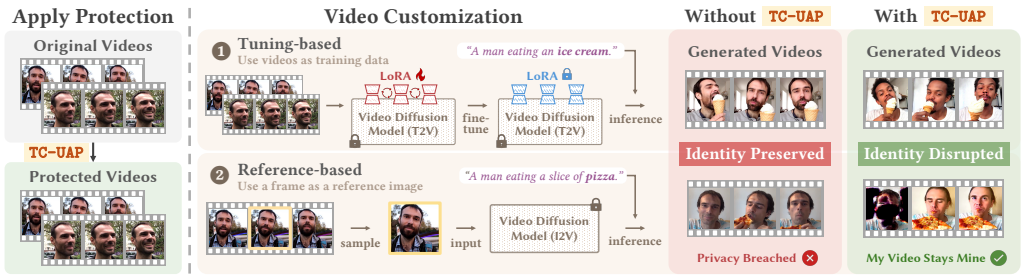


Figure 1: Protecting personal videos against unauthorized video customization. Once posted online, a user's videos may be collected and exploited by either tuning-based or reference-based customization. Our TC-UAP proactively protects the videos by applying imperceptible temporally consistent perturbations, making both customization methods fail to generate usable results.

frame from any of these videos as the reference image for reference-based customization. These threats may lead to fake news and violations of portrait rights. Protecting video content from being misused for unauthorized customization thus becomes a pressing concern.

Existing anti-customization works have mostly focused on adversarial perturbation-based image-level protection. Early efforts mainly protect *images* against tuning-based customization of text-to-image (T2I) models [38, 60, 64, 37, 44, 69, 77, 86, 24, 35, 36, 5, 46, 90, 43] and reference-based image-to-image (I2I) editing [57, 2, 31, 8]. More recent works have further extended image protection to video generation scenarios, where the protected content is still a reference image used by image-to-video (I2V) pipelines [53, 15, 65, 10, 45]. *In contrast, protecting videos has received much less attention, despite the growing prevalence of video-based content across online platforms.* This limitation is particularly concerning because publicly available videos can be exploited as both *references* and *training data* for the two kinds of unauthorized customization. However, protecting videos against both reference- and tuning-based customization remains largely unexplored, highlighting the need for a unified video protection method against both types of threats.

To this end, we explore protecting videos from reference- and tuning-based customization, and identify three temporal challenges (the complete analyses are provided in Section 3), namely, (i) *temporal compression*, (ii) *temporal overfitting*, and (iii) *temporal inconsistency*:

- (i) *Image-level perturbations, optimized frame by frame, cannot survive temporal compression in 3D video VAE.* Image-level protection can be applied to videos by optimizing perturbations frame by frame. While such perturbations remain applicable to reference-based I2V customization, they are insufficient for tuning-based video customization, where a 3D causal video VAE encodes the entire video into compact latent representations [81, 30, 66, 17]. This *temporal compression* substantially weakens frame-wise perturbations, causing them to fail after video encoding, as illustrated in Fig. 2. Therefore, protecting videos against both threats requires a video-level perturbation design.
- (ii) *A perturbation optimized on a single video is vulnerable to temporal editing and fails to generalize to unseen videos.* Optimizing a perturbation for a single video tends to overfit to the original temporal sequence. When the protected video undergoes temporal editing (e.g., trimming into a shorter clip), the protection may significantly degrade, as shown in Fig. 3. Moreover, such a video-specific perturbation cannot be directly transferred to unseen videos. These limitations pose challenges to temporal flexibility, identity-level generalization, and protection efficiency.
- (iii) *Temporally inconsistent perturbations are not robust to malicious temporal attacks.* An attacker can exploit cross-frame redundancy across video frames to perform temporal attacks (e.g., frame interpolation and temporal averaging), which can substantially attenuate temporally inconsistent perturbations, as shown in Fig. 4. This highlights the necessity of modeling temporal consistency of perturbations to ensure robust protection against temporal attacks.

To address these challenges, we propose Temporally Consistent Universal Adversarial Perturbations (dubbed TC-UAP), the first video protection method against both reference- and tuning-based video customization, as shown in Fig. 1. Specifically, we model the protection signal as an identity-level multi-frame UAP comprising multiple temporally indexed perturbation frames, thereby providing greater temporal expressiveness for disrupting the video VAE latent representations shared by both customization pipelines. Such a multi-frame UAP can be temporally repeated and applied to videos of different lengths, enabling it to be reused across clips without per-video optimization. Moreover, to make the perturbation survive the *temporal compression* of 3D video VAEs while avoiding *temporal*

overfitting, we adopt a *sliding-window optimization* strategy over multiple videos of the same identity. The window length is chosen to cover the dominant portion of the temporal receptive field, allowing each window to capture sufficient temporal dependencies induced by temporal compression. By sampling windows from different videos and temporal positions, this strategy exposes the UAP to diverse temporal contexts and prevents it from becoming tied to the complete temporal sequence of any training video. In addition, although the multi-frame formulation improves temporal expressiveness, it does not by itself guarantee temporal consistency. We therefore introduce *intrinsic temporal modeling* and an *extrinsic surrogate temporal-attack loss* to regularize cross-frame consistency and preserve protection effectiveness under temporal attacks. As a result, once optimized, the same UAP can be efficiently reused to protect unseen videos of the same identity while remaining robust against unseen temporal attacks.

Empirically, we conduct extensive experiments to evaluate `TC-UAP`. Both the quantitative and qualitative results show that our method achieves the strongest identity protection compared with existing methods under both reference- and tuning-based video customization, and remains robust under diverse unseen temporal attacks. Our contributions are summarized as follows:

- We are the first to formulate and study the problem of unified video protection against both reference- and tuning-based customization. We identify three key temporal challenges: (i) *temporal compression*, (ii) *temporal overfitting*, and (iii) *temporal inconsistency*.
- We propose `TC-UAP`, a unified video protection method that models the protection signal as an identity-level multi-frame UAP, thereby enhancing temporal expressiveness and enabling efficient reuse across videos of different lengths. We adopt a sliding-window optimization strategy to mitigate temporal overfitting, while intrinsic temporal modeling and a surrogate temporal-attack loss improve robustness against unseen temporal attacks.
- Extensive quantitative and qualitative experiments demonstrate that `TC-UAP` provides effective identity protection against both reference- and tuning-based video customization, generalizes to unseen videos of the same identity, and remains robust against a range of unseen temporal attacks.

2 Preliminaries

2.1 Problem Formulation

Consider a set of M videos $\mathcal{V}_y = \{V_i\}_{i=1}^{M}$ of an identity y. Each video is denoted as $V_i = \{v_i^t\}_{t=0}^{T_i-1} \in \mathbb{R}^{T_i \times C \times H \times W}$, where T_i is the number of frames, and C, H, and W denote the channel, height, and width of each frame v_i^t. An *attacker* may use these videos for either *tuning-based* or *reference-based* customization, which we introduce in Section 2.2. In both cases, the attacker aims to synthesize new videos that reproduce the target identity y.

The *goal of the protector* is to construct protected videos that remain nearly indistinguishable from the originals, while preventing both reference- and tuning-based customization from generating usable videos that faithfully preserve the target identity y. Following prior image protection works against diffusion-based customization [57, 69, 64, 37, 38], we adopt a white-box setting in which the protector has access to the VAE encoder.

Following the common perturbation-based protection paradigm [38, 60, 64, 53, 15, 65, 10, 45], the protection goal can be formulated as adding imperceptible perturbations $\delta = \{\delta_i\}_{i=1}^{M}$ to $\mathcal{V}_y$ to obtain a protected video set $\mathcal{V}'_y = \{V'_i\}_{i=1}^{M}$. For each V_i, the protector generates a perturbation δ_i, obtaining:

$$V'_i = V_i + \delta_i, \quad \text{s.t.} \quad ||\delta_i||_\infty \leq \eta, \tag{1}$$

where η denotes the perturbation budget. The perturbation optimization objective is defined as:

$$\delta^\star = \arg \min_{||\delta||_\infty \leq \eta} \mathcal{L}_{\text{id}}(\mathcal{V}_y, \delta), \tag{2}$$

where $\mathcal{L}_{\text{id}}$ is a surrogate objective, and minimizing it induces the customization failure defined above.

2.2 Video Customization

Diffusion Models. Diffusion Models (DMs) are generative models [61, 20, 42, 41] that learn a target data distribution. Given a clean sample z_0, the forward process produces a noisy sample at step τ as $z_\tau = \sqrt{\bar{\alpha}_\tau}\, z_0 + \sqrt{1-\bar{\alpha}_\tau}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$, where $\{\bar{\alpha}_\tau\}$ is a fixed noise schedule. A

denoising network ϵ_ω is trained to predict ϵ from z_τ under a conditioning signal c (e.g., text, image, etc.) encoded by a corresponding encoder $\mathcal{E}_{\text{text}}$:

$$\mathcal{L}_{\text{DM}}(\omega) = \mathbb{E}_{z_0,\,\tau,\,\epsilon,\,c}\big[\,\big\|\epsilon - \epsilon_\omega(z_\tau,\,\tau,\,\mathcal{E}_{\text{text}}(c))\big\|_2^2\,\big]. \tag{3}$$

Video VAE. Latent diffusion models (LDMs) [55] operate in a compressed latent space encoded by an encoder $\mathcal{E}$. For video diffusion, it is instantiated as a video VAE that operates along the spatial and temporal dimensions. Given a video $V_i \in \mathbb{R}^{T_i \times C \times H \times W}$, the encoder produces a latent:

$$z_i = \mathcal{E}(V_i) \in \mathbb{R}^{\tilde{V}_i \times \tilde{C} \times \tilde{H} \times \tilde{W}}, \qquad \tilde{V}_i = \tfrac{T_i - 1}{r_t} + 1,\;\; \tilde{H} = \tfrac{H}{r_h},\;\; \tilde{W} = \tfrac{W}{r_w}, \tag{4}$$

which is downsampled by r_h and r_w along the spatial dimension and r_t along the temporal dimension, with $\tilde{C}$ latent channels. We use $t \in \{0, \ldots, T_i - 1\}$ as the frame index in the video, and $p \in \{0, \ldots, \tilde{V}_i - 1\}$ as the temporal position in the latent space. The decoder $\mathcal{D}$ reconstructs $\hat{V}_i = \mathcal{D}(z_i)$.

Tuning-based Video Customization. Tuning-based methods fine-tune a pretrained denoising network ϵ_ω on a set of video-text pairs $\{(V_i, c_i)\}$ with the latents $z_i = \mathcal{E}(V_i)$. A lightweight adapter (e.g., LoRA [23]) parameterized by $\Delta\omega$ is optimized with the denoising objective:

$$\mathcal{L}_{\text{tune}}(\Delta\omega) = \mathbb{E}_{(V_i,c_i)\sim\{(V_i,c_i)\},\,\tau,\,\epsilon}\big[\,\big\|\epsilon - \epsilon_{\omega+\Delta\omega}(z_{i,\tau},\,\tau,\,\mathcal{E}_{\text{text}}(c_i))\big\|_2^2\,\big], \tag{5}$$

where $z_{i,\tau}$ is the noisy latent at step τ. Given the adapted model $\epsilon_{\omega+\Delta\omega}$ and a new prompt $c^\star$, a new video is generated by iterative denoising in the latent space, followed by VAE decoding.

Reference-based Video Customization. Reference-based methods take a reference image I_{ref} and a prompt c, and generate a video through a pretrained image-to-video (I2V) model g_ϕ. The reference is first encoded as $z_{\text{ref}} = \mathcal{E}(I_{\text{ref}})$ and injected into the denoising trajectory in one of two common ways:

$$\epsilon_\phi(z_\tau,\,\tau,\,\mathcal{E}_{\text{text}}(c),\,\mathcal{A}(z_{\text{ref}})) \quad \text{or} \quad \epsilon_\phi([\,z_\tau \,\|\, z_{\text{ref}}\,],\,\tau,\,\mathcal{E}_{\text{text}}(c))\,, \tag{6}$$

where $\mathcal{A}(\cdot)$ denotes a cross-attention branch and $[\,\cdot \,\|\, \cdot\,]$ denotes channel-wise concatenation. The denoised latent is finally decoded back to a video via VAE decoding.

3 Rethinking Video Protection

In this section, we rethink video protection from the perspective of video temporal structure. We first consider a general protection objective against both reference- and tuning-based customization, and then show that naively transferring image protection methods to videos leads to several failure modes. These observations reveal three video-specific temporal challenges that motivate our method design.

3.1 Video VAE as the Shared Vulnerability

Diffusion-based video customization typically starts by encoding videos or images into latent representations using a 3D video VAE, which compresses high-dimensional data (e.g., raw pixels) into compact latents for computational efficiency. Reference-based methods [18, 66, 3] encode the reference image before generation, while tuning-based methods [1, 23] encode training videos before fine-tuning. This makes the VAE latent a shared representation in both customization pipelines. We thus optimize the protective perturbation to disrupt the VAE latent, so that identity information is corrupted before being used by downstream customization.

To analyze protection effectiveness and identify video-specific challenges, we use the **VAE reconstruction** result in this section. An accurate reconstruction indicates that the latent contains usable information, and a corrupted one indicates that this information has been disrupted. All experiments in this section are conducted using the 3D causal video VAEs of LTX-2.3 [18] and Wan2.2 [66].

3.2 Temporal Challenges in Video Protection

Unlike existing image protection methods for T2I and I2I generation, and even I2V generation where protection is usually applied to a single image reference, protecting complete videos requires addressing additional challenges arising from temporal structure. We identify three key temporal challenges that hinder effective full-video protection: *temporal compression*, *temporal overfitting*, and *temporal inconsistency*.

Temporal compression: image-level perturbations cannot survive *temporal compression* in 3D video VAE. A straightforward way to protect videos is to extend image-level protection in a



Figure 2: Image-level protection targets the video VAE. (a) Clean image. (b) Protected image. (c) Clean video, *where the numbers indicate frame indices*. (d) Protected video obtained by applying image protection to each frame. Images are treated as single-frame videos, and all inputs are reconstructed by the same video VAE. The top row shows the inputs to the video VAE, and the bottom row shows the corresponding VAE reconstructions.



Figure 3: Temporal overfitting. (a) A perturbation optimized on a 25-frame video provides strong protection for the original video. (b) When the protected video is temporally cropped to its last 9 frames, the protection largely disappears. (c) Applying the same perturbation to an unseen 25-frame video fails to protect it.

frame-wise manner, i.e., by optimizing an image-level perturbation against video-based customization independently for each frame. We first consider this frame-wise image-level setting. Given a frame image v, an image-level perturbation ξ can be optimized by maximizing the distance between the VAE latent representations of the clean and perturbed inputs:

$$\xi^\star = \arg\max_{\|\xi\|_\infty \le \eta} \left\| \mathcal{E}(v+\xi) - \mathcal{E}(v) \right\|_2^2, \tag{7}$$

where $\mathcal{E}$ is the encoder of the 3D causal video VAE. As shown in Fig. 2(a-b), the *clean image* is accurately reconstructed, whereas the *protected image* is reconstructed as a near-black image, indicating that single-image protection is effective. *However, this effectiveness does not carry over to videos.* From the comparison between Fig. 2(c-d), when the same image-level objective is optimized and applied frame by frame for the video, although the reconstructed video is initially disrupted, the reconstruction *gradually recovers* in later frames.

We attribute this phenomenon to the **temporal compression** mechanism of 3D **causal** video VAEs:

- **Temporal compression weakens image-level perturbations.** The video VAE encoder compresses information from multiple frames into each temporal latent position. Since image-level perturbations are optimized independently for each frame, even if they effectively disrupt the corresponding single-frame latent representation, *their effect can be weakened after being aggregated with temporally compressed information from neighboring frames*. In other words, a perturbation that enlarges the latent distance for an individual frame does not necessarily enlarge the distance between the temporally compressed latent representations of the clean and protected videos. Further analysis of this temporal compression mechanism is provided in Appendix B.1.1.
- **Causal temporal modeling makes the degradation more severe in later frames.** Due to causal temporal modeling, such as causal temporal convolution [66, 18], the latent representation at each temporal position can only aggregate information from the current and previous frames, while future frames are inaccessible. As a result, later frames aggregate temporally compressed information from more previous frames, leading to stronger perturbation degradation. In contrast, earlier frames involve a shorter temporal history and are therefore less affected. Further analysis of the temporal receptive field of 3D causal video VAEs and comparisons with non-causal modeling are provided in Appendix B.1.1.

This suggests that video protection should optimize perturbations over multiple frames jointly to account for temporal dependencies caused by temporal compression and causal temporal modeling.



Figure 4: Attack-strength retention under 7 temporal attacks (named in the left column, formulations in Appendix B.1.2). Temporal consistency is evaluated by cosine similarity between adjacent frames.

Temporal overfitting: a perturbation optimized on a single video is vulnerable to *temporal editing* and *fails to generalize* to unseen videos. A natural next step is therefore to optimize perturbations at the video level. However, simply moving from image-level perturbations in a frame-wise manner to video-level perturbations for the full video is not enough. Specifically, a full video-level perturbation can easily become temporally inflexible: it may be vulnerable to *temporal edits* such as trimming or length changes, fail to generalize across different clips, and incur optimization costs that scale with the full video length. To examine this, we consider a straightforward video-level baseline that applies Projected Gradient Descent (PGD) [48] by directly optimizing a video-level perturbation $\xi \in \mathbb{R}^{T\times C\times H\times W}$ with the same shape as the input video:

$$\xi^\star \;=\; \arg\max_{\|\xi\|_\infty \le \eta} \big\| \mathcal{E}(V+\xi) - \mathcal{E}(V) \big\|_2^2. \tag{8}$$

As shown in Fig. 3(a), this baseline provides strong protection across all frames, showing that video-level optimization can disrupt video VAE reconstruction. However, the optimized perturbation overfits to the specific temporal sequence of the video used for optimization, leading to poor generalizable protection in two aspects:

- **Vulnerability to temporal editing of the same video.** When the protected video is temporally cropped (e.g., to its last 9 frames), the protection largely disappears, as shown in Fig. 3(b). This failure indicates that optimizing a perturbation over the complete video can cause *temporal overfitting* to the specific temporal sequence, making the protection sensitive to even common temporal edits such as changing the temporal span or starting position.
- **Failure to generalize to unseen videos.** Applying the same perturbation to an unseen 25-frame video fails to disrupt reconstruction, as shown in Fig. 3(c). This indicates that per-video PGD is not reusable across different video content, and a new perturbation must be optimized for each video.

These results suggest that effective video protection should avoid overfitting to a specific temporal sequence, thereby remaining robust to temporal cropping or length changes. Ideally, it should also generalize to unseen clips, reducing the need to optimize a new perturbation for every full video and thus enabling practical deployment.

Temporal inconsistency: temporally inconsistent perturbations are not robust to temporal attacks. Before using a protected video for customization, an attacker can apply *temporal attacks*, such as frame interpolation and frame averaging, *to weaken the protection by attenuating the perturbation in pixel space* before it is encoded by the video VAE. We observe that when each frame of the video perturbation is optimized independently, the perturbation becomes highly vulnerable to such temporal attacks. We refer to such a perturbation as **temporally inconsistent**. To quantify how well a perturbation survives a temporal attack $\mathcal{T}$, we measure the attack-strength retention. As shown in Fig. 4, a temporally inconsistent perturbation, which sits at the left end of the consistency axis, is almost completely erased by every attack, as its retention drops to nearly zero. The reason is that per-frame independent perturbations are temporally uncorrelated. Frame-averaging operations act as low-pass filters along the temporal dimension, attenuating the adversarial perturbation in pixel space. This suggests that δ_y is encouraged to be temporally consistent to be robust against such attacks. More details on the definition of temporal attacks and experiments are provided in Appendix B.1.2.

4 `TC-UAP`: Temporally Consistent Universal Adversarial Perturbations

In this section, we propose `TC-UAP`, an identity-level Temporally Consistent Universal Adversarial Perturbation for video protection against both tuning-based and reference-based customization.

4.1 Identity-Level Multi-Frame UAP

Identity-level multi-frame formulation. To enable practical and efficient protection, we seek to generate adversarial perturbations that can accommodate videos of varying lengths and generalize to unseen videos of the same identity without requiring per-video optimization. Accordingly, we model the protection signal as an identity-level universal adversarial perturbation (UAP). For identity y, given a training video set $\mathcal{V}_y^{\mathrm{Tr}} = \{V_i\}_{i=1}^{M_{\mathrm{Tr}}} \subseteq \mathcal{V}_y$, we learn a UAP δ_y shared across videos of that identity. To provide sufficient temporal expressiveness beyond conventional one-frame UAPs[1] [50], we instantiate $\delta_y \in \mathbb{R}^{N\times C\times H\times W}$ as an N-frame UAP with $\|\delta_y\|_\infty \le \eta$, where η is the perturbation budget and δ_y^n denotes its n-th perturbation frame. To protect a video V_i of length T_i, we repeat δ_y along the temporal dimension until it covers all T_i frames:

$$\delta_{T_i} = \mathrm{Repeat}_{T_i}(\delta_y) = \underbrace{\{\delta_y^1, \dots, \delta_y^N, \delta_y^1, \dots, \delta_y^N, \delta_y^1, \dots\}}_{\text{repeated and cropped to } T_i \text{ frames}}, \tag{9}$$

and obtain the protected video as $V_i' = V_i + \delta_{T_i}$. This construction allows the fixed-length multi-frame UAP to be applied to videos of varying lengths.

Video-level latent disruption objective. To provide unified protection against both customization pipelines, we target video VAE latent representations, which constitute a common intermediate representation in reference- and tuning-based customization. To mitigate the weakening effect of *temporal compression* in 3D video VAEs, we optimize the perturbation jointly at the video level rather than frame by frame. Therefore, an ideal full-video latent-disruption objective can be formulated as:

$$\mathcal{L}_{\mathrm{lat}}^{\mathrm{full}}(\delta_y) = -\,\mathbb{E}_{V_i\sim\mathcal{V}_y^{\mathrm{Tr}}} \left\|\mathcal{E}\big(V_i + \mathrm{Repeat}_{T_i}(\delta_y)\big) - \mathcal{E}(V_i)\right\|_2^2, \tag{10}$$

where $\mathcal{E}$ denotes the video VAE encoder. Minimizing this objective encourages the multi-frame UAP to induce a large distance between the latent representations of the clean and protected videos.

However, directly optimizing Eq. (10) over full-length videos incurs substantial *memory overhead* and may cause the perturbation to *overfit to the temporal sequences* of the training videos, including their full durations, fixed sequence boundaries, and fixed alignment between the causal sequence boundary and the starting phase of the repeated multi-frame UAP. Temporal edits such as starting-point shifts, cropping, and length variations alter these properties and may weaken the protection.

Sliding-window proxy. We therefore use sliding windows as a proxy for the full-video objective. As shown in Fig. 8(b) left column, under causal temporal modeling, each temporal latent position primarily depends on a local span of the current and preceding input frames. Accordingly, different latent positions within a window involve different effective history lengths, which gradually increase until the temporal receptive field is largely saturated. Motivated by this observation, we choose the window length L to cover at least the dominant portion of the temporal receptive field of the 3D causal video VAE. Although shorter than the full video, each L-length window captures sufficient temporal dependencies to account for the effects of temporal compression.

Formally, for a training video $V_i \in \mathcal{V}_y^{\mathrm{Tr}}$ of length T_i and its protected version $V_i' = V_i + \mathrm{Repeat}_{T_i}(\delta_y)$, we sample a starting temporal position $k \in \{0, \dots, T_i - L\}$ and extract $X = \{v_i^j\}_{j=k}^{k+L-1}$ and $X' = \{v_i'^j\}_{j=k}^{k+L-1}$. The practical latent-disruption objective is then defined as:

$$\mathcal{L}_{\mathrm{lat}}(\delta_y) = -\,\mathbb{E}_{V_i\sim\mathcal{V}_y^{\mathrm{Tr}},\,k} \left\|\mathcal{E}(X') - \mathcal{E}(X)\right\|_2^2. \tag{11}$$

Intuitively, this sliding-window objective serves as an effective proxy for the full-video objective. For a video longer than the optimization window, each latent position still depends primarily on a local temporal context whose dominant receptive field can be covered by the L-length window. When the multi-frame UAP is temporally repeated across such a video, latent features at later temporal positions remain influenced by the adversarial perturbations within their corresponding receptive fields, similarly to how latent features are perturbed during window-based optimization. Consequently, perturbation patterns learned within fixed-length windows can be extended to longer videos through temporal repetition while remaining effective under longer-sequence encoding.

[1] Empirically, we find that a one-frame UAP, when repeated across video frames, fails to provide sufficiently strong video-level protection under the video VAE latent-disruption objective. Detailed results are provided in Fig. 10.

4.2 Temporal Consistency Design

Although the multi-frame formulation provides greater temporal expressiveness, a multi-frame UAP is not necessarily temporally consistent when its frames are independently parameterized, leaving the resulting perturbation vulnerable to temporal attacks. To address this issue, we improve temporal robustness from two complementary aspects: an extrinsic surrogate temporal-attack loss and intrinsic temporal consistency modeling.

Extrinsic surrogate temporal-attack loss. We first introduce a surrogate temporal-attack loss that directly optimizes the multi-frame UAP to remain effective after temporal attack:

$$\mathcal{L}_{\text{temporal}}(\delta_y) = -\,\mathbb{E}_{X,\,X'} \left\| \mathcal{E}(\mathcal{T}(X')) - \mathcal{E}(\mathcal{T}(X)) \right\|_2^2, \tag{12}$$

where $\mathcal{T}$ is a surrogate temporal attack, such as frame averaging, which averages each pair of adjacent frames with weights $0.5/0.5$. Specifically, this objective encourages the multi-frame UAP to still preserve a large VAE latent distance after the surrogate temporal attack, thus explicitly enhancing its robustness. However, such attack-aware optimization may become tied to the specific surrogate attack and fail to generalize to unseen temporal attacks.

Intrinsic temporal consistency modeling. As illustrated in Fig. 4, perturbations with stronger temporal consistency retain their attack strength more effectively across different temporal attacks. Motivated by this observation, we introduce an *attack-agnostic temporal-consistency prior* into the multi-frame UAP parameterization to improve generalization to unseen temporal attacks. The parameterization is designed to balance temporal consistency with perturbation expressiveness, as overly strong temporal constraints may limit the expressiveness required for video protection, whereas weak temporal correlation may leave different frames nearly independent, making stable temporal consistency difficult to achieve. Based on this trade-off, we use a reparameterization based on the average of previous frames. Specifically, we introduce a parameter $\theta_y \in \mathbb{R}^{N\times C\times H\times W}$ and define the reparameterization $\delta_y = R(\theta_y)$ as:

$$\delta_y^1 \;=\; \theta_y^1, \qquad \delta_y^n \;=\; \frac{1}{n-1}\sum_{i=1}^{n-1}\delta_y^i \;+\; \theta_y^n, \quad \text{for } n \geq 2, \tag{13}$$

with inverse $\theta_y = R^{-1}(\delta_y)$, where δ_y^n and θ_y^n denote the n-th frame of δ_y and θ_y, respectively.

In this formulation, each UAP frame consists of two terms: the average of all previous UAP frames and a learnable frame-specific term. The first term makes δ_y^n depend on perturbations at earlier temporal indices, thereby introducing temporal correlation across perturbation frames. The second term θ_y^n retains a learnable component specific to the n-th frame, thereby preserving the expressiveness of the multi-frame UAP. This attack-agnostic temporal structure complements the surrogate temporal-attack loss and improves robustness against unseen temporal attacks.

4.3 Optimization for `TC-UAP`

Overall objective. Combining the untargeted latent distance loss in Eq. (11) with the surrogate temporal attack loss in Eq. (12), the total training objective is:

$$\mathcal{L}_{\text{id}}(\delta_y) \;=\; \mathcal{L}_{\text{lat}}(\delta_y) \;+\; \lambda\,\mathcal{L}_{\text{temporal}}(\delta_y), \quad \delta_y = R(\theta_y), \tag{14}$$

where λ controls the contribution of the surrogate temporal-attack loss. Minimizing $\mathcal{L}_{\text{id}}$ encourages the multi-frame UAP to induce a large VAE latent distance both before and after the surrogate temporal attack, thereby jointly optimizing protection effectiveness and temporal robustness.

Sliding-window-based Projected Gradient Descent. For each identity y, the UAP is optimized over a training video set $\mathcal{V}_y^{\text{Tr}} = \{V_i\}_{i=1}^{M_{\text{Tr}}}$. At each training step, we sample a video $V_i \in \mathcal{V}_y^{\text{Tr}}$ with length T_i, and construct the protected video $V_i' = V_i + \text{Repeat}_{T_i}(\delta_y)$ using Eq. (9). We then uniformly sample a starting frame index $k \in \{0, \ldots, T_i - L\}$, and extract the clean and protected windows from the same temporal position, i.e., $X = \{v_i^j\}_{j=k}^{k+L-1}$ and $X' = \{v_i'^j\}_{j=k}^{k+L-1}$. The total loss is computed according to Eq. (14). We optimize θ_y by back-propagating gradients through $\delta_y = R(\theta_y)$. After each update, the UAP $\delta_y = R(\theta_y)$ is projected in δ_y-space to satisfy the ℓ_∞ budget:

$$\delta_y \leftarrow \Pi_{\|\delta_y\|_\infty \leq \eta}\big(R(\theta_y)\big), \qquad \theta_y \leftarrow R^{-1}(\delta_y), \tag{15}$$

where $\Pi_{\|\delta_y\|_\infty \leq \eta}$ denotes element-wise projection onto the ℓ_∞ ball of radius η, implemented by clamping each perturbation value to $[-\eta, \eta]$. *Algorithm 1 summarizes the optimization procedure.*

Table 1: Results of protection quality evaluated under tuning-based and reference-based settings. **Bold** indicates the best result and underline the second-best. On the *Ours* row, green percentages report the relative improvement of our method against the strongest baseline.

Dataset	Methods	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
HDTF	Clean	39.64	0.973	0.016	0.000	0.598	0.609	41.69	0.979	0.011	0.017	0.607	0.624
	PhotoGuard	29.44	0.618	0.590	0.000	0.572	0.585	30.64	0.688	0.451	0.020	0.556	0.573
	Mist	31.54	0.770	0.347	0.000	0.544	0.557	31.25	0.743	0.374	0.003	0.527	0.544
	IDProtector	29.15	0.757	0.397	0.000	0.430	0.459	29.33	0.763	0.401	0.003	0.179	0.191
	Ours	**13.86** ↑52.45%	**0.368** ↑40.45%	**0.657** ↑11.36%	**0.027** –	**0.077** ↑82.09%	**0.080** ↑82.57%	**10.08** ↑65.63%	**0.153** ↑77.76%	**0.736** ↑63.19%	**0.321** ↑1505.00%	**0.129** ↑27.93%	**0.141** ↑26.18%
CelebV-HQ	Clean	39.47	0.972	0.019	0.000	0.630	0.650	42.27	0.981	0.011	0.013	0.611	0.637
	PhotoGuard	29.26	0.617	0.600	0.000	0.599	0.617	30.59	0.695	0.464	0.017	0.543	0.571
	Mist	31.35	0.767	0.378	0.000	0.554	0.579	31.23	0.743	0.398	0.009	0.501	0.525
	IDProtector	29.16	0.757	0.413	0.000	0.444	0.474	29.43	0.764	0.418	0.013	0.232	0.250
	Ours	**12.91** ↑55.73%	**0.374** ↑39.38%	**0.645** ↑7.50%	**0.006** –	**0.113** ↑74.55%	**0.125** ↑73.63%	**8.84** ↑69.96%	**0.171** ↑75.40%	**0.708** ↑52.59%	**0.225** ↑1223.53%	**0.139** ↑40.09%	**0.140** ↑44.00%
TalkVid	Clean	36.94	0.960	0.018	0.000	0.623	0.636	39.19	0.971	0.013	0.000	0.628	0.650
	PhotoGuard	28.51	0.602	0.567	0.000	0.585	0.598	29.97	0.691	0.438	0.001	0.572	0.595
	Mist	30.44	0.761	0.349	0.000	0.551	0.568	30.24	0.734	0.380	0.001	0.527	0.545
	IDProtector	28.69	0.751	0.395	0.000	0.431	0.463	29.10	0.765	0.403	0.001	0.310	0.327
	Ours	**12.17** ↑57.31%	**0.381** ↑36.71%	**0.637** ↑12.35%	**0.013** –	**0.160** ↑62.88%	**0.164** ↑64.58%	**8.00** ↑72.51%	**0.187** ↑72.94%	**0.709** ↑61.87%	**0.146** –	**0.275** ↑11.29%	**0.286** ↑12.54%

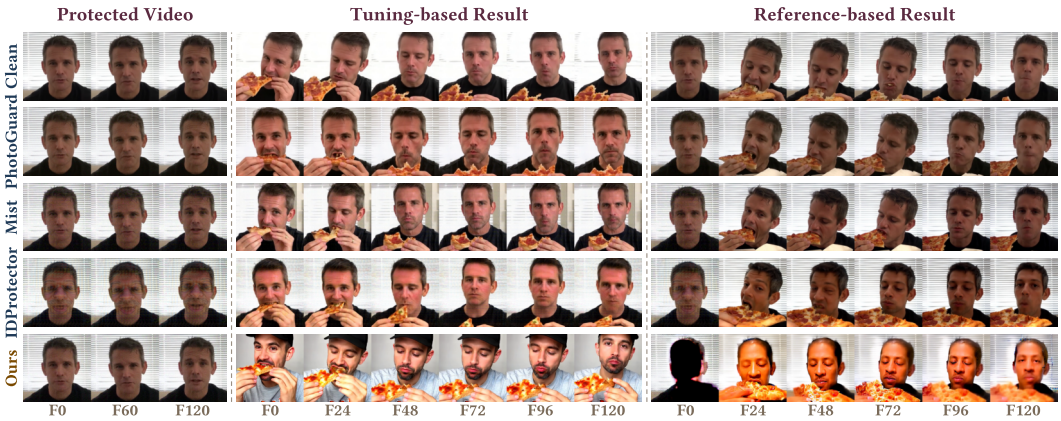


Figure 5: Qualitative comparison of reference- and tuning-based customization results.

Applying the learned UAP to unseen videos. After optimization, given a new video V with length T, the protected video V' can be obtained by repeating δ_y along the temporal dimension until it covers all frames, and cropping it to length T, i.e., $V' = V + \mathrm{Repeat}_T(\delta_y)$. Therefore, the identity-level multi-frame UAP provides generalizable and efficient protection, as it can be reused across unseen videos of the same identity.

5 Experiments

Datasets. To the best of our knowledge, there is no existing benchmark for protecting videos against diffusion-based customization. Therefore, we adapt three human-centered video datasets and evaluate on them: HDTF [84], CelebV-HQ [91] and TalkVid [7]. For each dataset, we select 20 identities and split their videos into 15 training clips for UAP optimization and 15 test clips for evaluating protection effectiveness. **Evaluation metrics.** For protection effectiveness, we report PSNR, SSIM [74] and LPIPS [83] for VAE reconstruction, and Face Detection Failure Rate (FDFR) [12] and identity similarity measured by ArcFace [11] (ISM1) and CurricularFace [25] (ISM2) for identity preservation. For invisibility, we report Video Multi-Method Assessment Fusion (VMAF) [51]. Higher VMAF indicates better perceptual quality and less visible perturbation. **Implementation.** For each identity, we set the UAP length to 9 frames and optimize it on 15 training clips with perturbation budget $\eta = 0.1$. Following Eq. (9), the learned UAP is repeated and added to each test clip to obtain the protected test set. For tuning-based evaluation, we fine-tune LoRA adapters for LTX-2.3 [18] and Wan2.2-5B [66] on the protected test set. For reference-based evaluation, the official LTX-2.3 and Wan2.2-5B TI2V pipelines are used, each conditioned on an extracted frame of a protected video. The surrogate temporal attack $\mathcal{T}$ is implemented as frame averaging, in which adjacent frames are averaged with weights $0.5/0.5$. **Baselines.** We consider three baseline image protection methods:

Table 3: Robustness results under different unseen temporal attacks on the TalkVid dataset.

Operation	Methods	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Original	PGD	**6.19**	**0.096**	**0.857**	**0.169**	**0.001**	**0.006**	24.43	0.638	0.446	0.001	0.534	0.551
	Ours	12.17	0.381	0.637	0.013	0.160	0.164	**8.00**	**0.187**	**0.709**	**0.146**	**0.275**	**0.286**
Box Filter	PGD	26.21	0.741	0.351	0.000	0.463	0.481	34.43	0.849	0.282	0.004	0.550	0.576
	Ours	**14.66**	**0.443**	**0.647**	**0.001**	**0.136**	**0.141**	**8.92**	**0.223**	**0.708**	**0.168**	**0.151**	**0.156**
Gaussian Filter	PGD	32.82	0.803	0.356	0.000	0.469	0.491	32.37	0.806	0.342	0.002	0.540	0.561
	Ours	**14.49**	**0.432**	**0.681**	**0.013**	**0.111**	**0.117**	**8.91**	**0.217**	**0.715**	**0.165**	**0.164**	**0.171**
Low-pass Filter	PGD	34.66	0.861	0.280	0.000	0.503	0.523	36.64	0.905	0.216	0.002	0.361	0.389
	Ours	**14.23**	**0.426**	**0.685**	**0.017**	**0.097**	**0.107**	**8.57**	**0.197**	**0.754**	**0.220**	**0.077**	**0.071**

PhotoGuard [57], Mist [37], IDProtector [62]. We apply each baseline to every frame of the test videos and adopt the same perturbation budget $\eta = 0.1$. *Further details are provided in Appendix C.*

5.1 Main Results

Table 2: Perturbation Invisibility (VMAF↑).

Method	HDTF	CelebV-HQ	TalkVid
Clean	99.17	100.64	100.14
Random ($\eta = 0.1$)	92.65	93.87	94.01
PhotoGuard	80.86	82.76	81.93
Mist	74.63	76.76	75.49
IDProtector	52.53	54.94	56.50
Ours	**82.76** (+1.90)	**84.60** (+1.84)	**84.81** (+2.88)

Comparisons on LTX-2.3 [18] and Wan2.2-5B [66] are shown in Tables 1 and 7. On LTX-2.3, our method achieves the lowest ISM and highest FDFR across all datasets, indicating that the generated identity differs most from the target identity and is even undetectable as a face in many frames. We also analyze VAE reconstruction to verify whether the input latent is effectively disrupted. Compared with the strongest baseline, our method reduces reconstruction PSNR by 15.96 dB in the tuning-based customization setting and 20.31 dB in the reference-based customization setting. These results show that our method strongly disrupts the VAE latent, which is consistent with its strong ability to reduce identity preservation in customization. In addition, our method achieves the highest VMAF (Table 2), indicating that our protection is the most imperceptible.

We also present qualitative results in Fig. 5. Under our method, the generated faces no longer look like the target person, while most baselines still produce faces that clearly look like the target. Among them, IDProtector shows some degree of protection in reference-based customization results. However, we notice that IDProtector adds clearly visible patterns at fixed facial regions, with further analysis in Appendix G.2. More qualitative comparisons are shown in Appendix D.5.

Robustness evaluation against temporal attacks. We further evaluate the robustness of our method against temporal attacks. We test three unseen attacks: box filter, Gaussian filter, and low-pass filter. The details of the three temporal attacks are shown in Appendix C.2. We compare our method with a PGD baseline that directly optimizes a per-video perturbation for each video against the VAE encoder, without any temporal-consistency design, as in Eq. (8). This gives strong protection on each video. Results are shown in Table 3. On the original protected videos, the PGD baseline achieves the strongest protection. However, under the temporal attacks, the PGD baseline loses most of its protective effect. In contrast, our method changes slightly after each filter. These results demonstrate that our method is robust to unseen temporal attacks. This confirms that our temporal-consistency design provides robustness, rather than resistance to only the specific attack seen during training.

More results. We further provide **ablation study** in Appendix E, conduct **hyperparameter analyses** of the perturbation budget, UAP length, temporal-attack loss weight, and sliding-window length and assess **computational efficiency** in Appendix F, and evaluate **black-box transferability** and **robustness against diverse transformations** in Appendix G.

6 Conclusion

In this work, we study video protection against unauthorized reference- and tuning-based customization, and identify three key temporal challenges: temporal compression, temporal overfitting, and temporal inconsistency. We propose `TC-UAP`, the first video-level protection method against both customization threats. By learning an identity-level multi-frame UAP from multiple videos of the same identity, `TC-UAP` generalizes to unseen videos and clips with different lengths, while intrinsic temporal modeling and an extrinsic surrogate temporal-attack loss improve robustness against temporal attacks. Extensive experiments show that `TC-UAP` achieves imperceptible, generalizable, and robust protection, taking a step toward securing video content against unauthorized customization.

Acknowledgements

Mingming Gong is supported by ARC DP240102088 and WIS–MBZUAI 142571. Tongliang Liu is partially supported by the following Australian Research Council projects: DP260102466, FT220100318, DP220102121, LP220100527, LP220200949.

References

[1] Rameen Abdal, Or Patashnik, Ivan Skorokhodov, Willi Menapace, Aliaksandr Siarohin, Sergey Tulyakov, Daniel Cohen-Or, and Kfir Aberman. Dynamic concepts personalization from single videos. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–9, 2025.

[2] Aniruddha Bala, Rohit Chowdhury, Rohan Jaiswal, and Siddharth Roheda. Dct-shield: A robust frequency-domain defense against malicious image editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

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

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[5] Jiale Cai, Gezheng Xu, Zhihao Li, Ruiyi Fang, Ruizhi Pu, Di Wu, Qicheng Lao, Charles Ling, and Boyu Wang. FUSE: Full-spectrum unlearnable examples via spectral equalization. In *Forty-third International Conference on Machine Learning*, 2026.

[6] Junjie Cao, Kaizhou Li, Xinchun Yu, Hongxiang Li, and Xiaoping Zhang. Videoguard: Protecting video content from unauthorized editing. *arXiv preprint arXiv:2508.03480*, 2025.

[7] Shunian Chen, Hejin Huang, Yexin Liu, Zihan Ye, Pengcheng Chen, Chenghao Zhu, Michael Guan, Rongsheng Wang, Junying Chen, Guanbin Li, et al. Talkvid: A large-scale diversified dataset for audio-driven talking head synthesis. *arXiv preprint arXiv:2508.13618*, 2025.

[8] Shuo Chen, Fengming Huang, Yu Yao, Mingming Gong, and Tongliang Liu. Scalable minimal-change learning for controllable image editing. In *Advances in Neural Information Processing Systems*, 2026.

[9] Yingjie Chen, Shilun Lin, Cai Xing, Qixin Yan, Wenjing Wang, Dingming Liu, Hao Liu, Chen Li, and Jing Lyu. Identity as presence: Towards appearance and voice personalized joint audio-video generation. *arXiv preprint arXiv:2603.17889*, 2026.

[10] Rohit Chowdhury, Aniruddha Bala, Rohan Jaiswal, and Siddharth Roheda. Vid-freeze: Protecting images from malicious image-to-video generation via temporal freezing. *arXiv preprint arXiv:2509.23279*, 2025.

[11] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2019.

[12] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020.

[13] Weitao Feng, Lixu Wang, Peizhuo Lv, Tianyi Wei, Jie Zhang, Chongyang Gao, Sinong Zhan, and Wei Dong. Token buncher: Shielding llms from harmful reinforcement learning fine-tuning. *arXiv preprint arXiv:2508.20697*, 2025.

[14] Sensen Gao, Zhaoqing Wang, Qihang Cao, Dongdong Yu, Changhu Wang, Tongliang Liu, Mingming Gong, and Jiawang Bian. Oneworld: Taming scene generation with 3d unified representation autoencoder. In *European Conference on Computer Vision*, pages 95–113. Springer, 2026.

[15] Dongnan Gui, Xun Guo, Wengang Zhou, and Yan Lu. I2vguard: Safeguarding images against misuse in diffusion-based image-to-video models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12595–12604, 2025.

[16] Xu Guo, Fulong Ye, Qichao Sun, Liyang Chen, Bingchuan Li, Pengze Zhang, Jiawei Liu, Songtao Zhao, Qian He, and Xiangwang Hou. Dreamid-omni: Unified framework for controllable human-centric audio-video generation. *arXiv preprint arXiv:2602.12160*, 2026.

[17] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

[18] Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

[19] Xuanhua He, Quande Liu, Shengju Qian, Xin Wang, Tao Hu, Ke Cao, Keyu Yan, and Jie Zhang. Id-animator: Zero-shot identity-preserving human video generation. *arXiv preprint arXiv:2404.15275*, 2024.

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

[21] Ziming Hong, Tianyu Huang, Runnan Chen, Shanshan Ye, Mingming Gong, Bo Han, and Tongliang Liu. Adlift: Lifting adversarial perturbations to safeguard 3d gaussian splatting assets against instruction-driven editing. *arXiv preprint arXiv:2512.07247*, 2025.

[22] Ziming Hong, Yongli Xiang, and Tongliang Liu. Toward robust non-transferable learning: A survey and benchmark. *arXiv preprint arXiv:2502.13593*, 2025.

[23] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

[24] Mingwang Hu, Yi Tu, Dandan Tu, and Lanjun Wang. Targeted ensemble defense against unauthorized text-to-image identity customization. *Information Fusion*, page 103696, 2025.

[25] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: Adaptive curriculum learning loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5901–5910, June 2020. doi: 10.1109/CVPR42600.2020.00594.

[26] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[27] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[28] Felix Juefei-Xu, Run Wang, Yihao Huang, Qing Guo, Lei Ma, and Yang Liu. Countering malicious deepfakes: Survey, battleground, and horizon. *International journal of computer vision*, 130(7):1678–1734, 2022.

[29] Rawal Khirodkar, Jyun-Ting Song, Jinkun Cao, Zhengyi Luo, and Kris Kitani. Harmony4d: A video dataset for in-the-wild close human interactions. *Advances in Neural Information Processing Systems*, 37:107270–107285, 2024.

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

[31] Chanhui Lee, Donggyu Choi, Seunghyun Shin, Hae-gon Jeon, and Jeany Son. Universal image immunization against diffusion-based image editing via semantic injection. In *European Conference on Computer Vision (ECCV)*, 2026.

[32] Guanlin Li, Shuai Yang, Jie Zhang, and Tianwei Zhang. Prime: Protect your videos from malicious editing. *arXiv preprint arXiv:2402.01239*, 2024.

[33] Xinfeng Li, Tianze Qiu, Yingbin Jin, Lixu Wang, Hanqing Guo, Xiaojun Jia, Xiaofeng Wang, and Wei Dong. Webcloak: Characterizing and mitigating threats from llm-driven web agents as intelligent scrapers. In *Proceedings of the 2026 IEEE Symposium on Security and Privacy (SP)*, 2026.

[34] Yiming Li, Shuo Shao, Yu He, Junfeng Guo, Tianwei Zhang, Zhan Qin, Pin-Yu Chen, Michael Backes, Philip Torr, Dacheng Tao, et al. Rethinking data protection in the (generative) artificial intelligence era. *arXiv preprint arXiv:2507.03034*, 2025.

[35] Zhihao Li, Jiale Cai, Gezheng Xu, Hao Zheng, Qiuyue Li, Fan Zhou, Shichun Yang, Charles Ling, and Boyu Wang. Versatile transferable unlearnable example generator. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[36] Zhihao Li, Gezheng Xu, Jiale Cai, Ruiyi Fang, Di Wu, Qicheng Lao, Charles Ling, and Boyu Wang. When priors backfire: On the vulnerability of unlearnable examples to pretraining. In *The Fourteenth International Conference on Learning Representations*, 2026.

[37] Chumeng Liang and Xiaoyu Wu. Mist: Towards improved adversarial examples for diffusion models. *arXiv preprint arXiv:2305.12683*, 2023.

[38] Chumeng Liang, Xiaoyu Wu, Yang Hua, Jiaru Zhang, Yiming Xue, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. *arXiv preprint arXiv:2302.04578*, 2023.

[39] Runqi Lin, Bo Han, Fengwang Li, and Tongliang Liu. Understanding and enhancing the transferability of jailbreaking attacks. In *The Thirteenth International Conference on Learning Representations*.

[40] Runqi Lin, Alasdair Paren, Suqin Yuan, Muyang Li, Philip Torr, Adel Bibi, and Tongliang Liu. Force: Transferable visual jailbreaking attacks via feature over-reliance correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8610–8620, 2026.

[41] Yexiong Lin, Jia Shi, Shanshan Ye, Wanyu Wang, Yu Yao, and Tongliang Liu. Subflow: Sub-mode conditioned flow matching for diverse one-step generation. *arXiv preprint arXiv:2604.12273*, 2026.

[42] Yexiong Lin, Yu Yao, Yang Zhou, and Tongliang Liu. Beyond optimal transport: Model-aligned coupling for flow matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pages 3955–3964, June 2026.

[43] Hanqing Liu, Shouwei Ruan, Jiahuan Long, Junqi Wu, Jiacheng Hou, Huili Tang, Tingsong Jiang, Weien Zhou, and Wen Yao. Eva-vla: Evaluating vision-language-action models' robustness under real-world physical variations. *arXiv preprint arXiv:2509.18953*, 2025.

[44] Yixin Liu, Chenrui Fan, Yutong Dai, Xun Chen, Pan Zhou, and Lichao Sun. Metacloak: Preventing unauthorized subject-driven text-to-image diffusion-based synthesis via meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24219–24228, 2024.

[45] Zeqian Long, Ozgur Kara, Haotian Xue, Yongxin Chen, and James M Rehg. Immune2v: Image immunization against dual-stream image-to-video generation. *arXiv preprint arXiv:2604.10837*, 2026.

[46] Giulio Lovisotto, H.C.M. Turner, Ivo Sluganovic, Martin Strohmeier, and Ivan Martinovic. Slap: Improving physical adversarial examples with short-lived adversarial perturbations. In *USENIX Security Symposium*, 2020. URL https://api.semanticscholar.org/CorpusID:220403405.

[47] Yingzi Ma, Xiaogeng Liu, Yawen Zheng, and Chaowei Xiao. Safegen-bench: Benchmarking safety in image-conditioned text-to-video generation. *arXiv preprint arXiv:2606.01481*, 2026.

[48] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[49] Yibo Miao, Yifan Zhu, Lijia Yu, Jun Zhu, Xiao-Shan Gao, and Yinpeng Dong. T2vsafetybench: Evaluating the safety of text-to-video generative models. *Advances in Neural Information Processing Systems*, 37:63858–63872, 2024.

[50] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017.

[51] Netflix. VMAF: Video multi-method assessment fusion. https://github.com/Netflix/vmaf.

[52] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning (ICML)*, 2022.

[53] Yan Pang, Baicheng Chen, Yang Zhang, and Tianhao Wang. Vgmshield: Mitigating misuse of video generative models. *arXiv preprint arXiv:2402.13126*, 2024.

[54] Josephine Passananti, Stanley Wu, Shawn Shan, Haitao Zheng, and Ben Y Zhao. Disrupting style mimicry attacks on video imagery. *arXiv preprint arXiv:2405.06865*, 2024.

[55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[56] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir 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.

[57] Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious AI-powered image editing. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 29894–29918. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/salman23a.html.

[58] Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

[59] Fahad Shamshad, Muzammal Naseer, and Karthik Nandakumar. Clip2protect: Protecting facial privacy using text-guided makeup via adversarial latent search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20595–20605, 2023.

[60] Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y Zhao. Glaze: Protecting artists from style mimicry by Text-to-Image models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2187–2204, 2023.

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

[62] Yiren Song, Pei Yang, Hai Ci, and Mike Zheng Shou. Idprotector: An adversarial noise encoder to protect against id-preserving image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3019–3028, 2025.

[63] Yuhao Sun, Lingyun Yu, Hongtao Xie, Jiaming Li, and Yongdong Zhang. Diffam: Diffusion-based adversarial makeup transfer for facial privacy protection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24584–24594, June 2024.

[64] Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc N Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2116–2127, 2023.

[65] Duc Vu, Anh Nguyen, Chi Tran, and Anh Tran. Anti-i2v: Safeguarding your photos from malicious image-to-video generation. *arXiv preprint arXiv:2603.24570*, 2026.

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

[67] Zhenchen Wan, Yanwu Xu, Dongting Hu, Weilun Cheng, Tianxi Chen, Zhaoqing Wang, Feng Liu, Tongliang Liu, and Mingming Gong. Mft-viton: High-fidelity virtual try-on with minimal input via a mask-free transformer-diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1985–1994, 2025.

[68] Zhenchen Wan, Ce Chen, Runqi Lin, Jiaxin Huang, Tianxi Chen, Yanwu Xu, Tongliang Liu, and Mingming Gong. Mobile-vton: High-fidelity on-device virtual try-on. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 38081–38090, 2026.

[69] Feifei Wang, Zhentao Tan, Tianyi Wei, Yue Wu, and Qidong Huang. Simac: A simple anti-customization method for protecting face privacy against text-to-image synthesis of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12047–12056, 2024.

[70] Kaisiyuan Wang, Qianyi Wu, Linsen Song, Zhuoqian Yang, Wayne Wu, Chen Qian, Ran He, Yu Qiao, and Chen Change Loy. Mead: A large-scale audio-visual dataset for emotional talking-face generation. In *ECCV*, 2020.

[71] Lixu Wang, Shichao Xu, Ruiqi Xu, Xiao Wang, and Qi Zhu. Non-transferable learning: A new approach for model ownership verification and applicability authorization. *arXiv preprint arXiv:2106.06916*, 2021.

[72] Zhaoqing Wang, Xiaobo Xia, Zhuolin Bie, Jinlin Liu, Dongdong Yu, Jia-Wang Bian, and Changhu Wang. Taming camera-controlled video generation with verifiable geometry reward. *arXiv preprint arXiv:2512.02870*, 2025.

[73] Zhaoqing Wang, Xiaobo Xia, Runnan Chen, Dongdong Yu, Changhu Wang, Mingming Gong, and Tongliang Liu. Lavin-dit: Large vision diffusion transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20060–20070, 2025.

[74] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612, 2004.

[75] Zihao Wang, Yuxiang Wei, Fan Li, Renjing Pei, Hang Xu, and Wangmeng Zuo. Ace: Anti-editing concept erasure in text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23505–23515, 2025.

[76] Qiangqiang Wu, Yi Yu, Chenqi Kong, Ziquan Liu, Jia Wan, Haoliang Li, Alex C Kot, and Antoni B Chan. Temporal unlearnable examples: Preventing personal video data from unauthorized exploitation by object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11110–11121, 2025.

[77] Zhihao Wu, Yushi Cheng, Tianyang Sun, Xiaoyu Ji, and Wenyuan Xu. Myopia: Protecting face privacy from malicious personalized text-to-image synthesis via unlearnable examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 905–913, 2025.

[78] Yongli Xiang, Ziming Hong, Zhaoqing Wang, Xiangyu Zhao, Bo Han, and Tongliang Liu. When safety collides: Resolving multi-category harmful conflicts in text-to-image diffusion via adaptive safety guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14746–14755, 2026.

[79] Yongli Xiang, Zhifang Zhang, Bojun Yang, Ziming Hong, Lei Feng, Miao Xu, and Tongliang Liu. When agents learn to be you: Benchmarking privacy leakage, impersonation risk, and defenses in persona skills. *arXiv preprint arXiv:2608.03700*, 2026.

[80] Xiao Yang, Dingcheng Yang, Yinpeng Dong, Wenjian Yu, Hang Su, and Jun Zhu. Delving into the adversarial robustness on face recognition. *arXiv preprint arXiv:2007.04118*, 2020.

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

[82] Shenghai Yuan, Jinfa Huang, Xianyi He, Yunyang Ge, Yujun Shi, Liuhan Chen, Jiebo Luo, and Li Yuan. Identity-preserving text-to-video generation by frequency decomposition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12978–12988, 2025.

[83] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[84] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3661–3670, 2021.

[85] Longjie Zhao, Ziming Hong, Jiaxin Huang, Runnan Chen, Mingming Gong, and Tongliang Liu. Intellectual property protection for 3d gaussian splatting assets: A survey. *arXiv preprint arXiv:2602.03878*, 2026.

[86] Zhengyue Zhao, Jinhao Duan, Xing Hu, Kaidi Xu, Chenan Wang, Rui Zhang, Zidong Du, Qi Guo, and Yunji Chen. Unlearnable examples for diffusion models: Protect data from unauthorized exploitation. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*, 2024. URL https://openreview.net/forum?id=CHTRtLPYNF.

[87] Bowen Zheng, Yongli Xiang, Ziming Hong, Zerong Lin, Chaojian Yu, Tongliang Liu, and Xinge You. Vii: Visual instruction injection for jailbreaking image-to-video generation models. *arXiv preprint arXiv:2602.20999*, 2026.

[88] Jiyang Zheng, Siqi Pan, Yu Yao, Zhaoqing Wang, Dadong Wang, and Tongliang Liu. Aligning what matters: Masked latent adaptation for text-to-audio-video generation. *Advances in Neural Information Processing Systems*, 38:173244–173272, 2026.

[89] Zangwei Zheng, Xiangyu Peng, Yuxuan Lou, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, et al. Open-sora 2.0: Training a commercial-level video generation model in 200k. *arXiv preprint arXiv:2503.09642*, 2025.

[90] Ce Zhou, Qiben Yan, Yan Shi, and Lichao Sun. Doublestar: Long-range attack towards depth estimation based obstacle avoidance in autonomous systems. In *USENIX Security Symposium*, August 2022.

[91] Hao Zhu, Wayne Wu, Wentao Zhu, Liming Jiang, Siwei Tang, Li Zhang, Ziwei Liu, and Chen Change Loy. CelebV-HQ: A large-scale video facial attributes dataset. In *European Conference on Computer Vision*, 2022.

Appendices

A Related Work **18**

B Extended Methodology **18**

B.1 Analysis of Video Protection 18

B.2 Analysis of UAP 20

B.3 `TC-UAP` Optimization Procedure 21

C Extended Experimental Setup **21**

C.1 Data Preparation 21

C.2 Evaluation Metrics and Protocols 24

C.3 Baselines 25

C.4 Implementation Details 25

D Additional Experimental Results **26**

D.1 Full Results on Wan2.2 26

D.2 Additional Evaluation Metrics 26

D.3 Generated Outcomes Analysis across Prompts and Seeds 28

D.4 Protection under Challenging Video Contents 28

D.5 More Qualitative Results 28

E Ablation Study **33**

F Analysis of `TC-UAP` Hyperparameters **34**

F.1 Perturbation Budget η 34

F.2 UAP Length N 34

F.3 Temporal-Attack Loss Weight λ 34

F.4 Sliding-Window Length L 35

G Transferability and Robustness Evaluation **36**

G.1 Evaluation of `TC-UAP` 36

G.2 Baseline Comparison and Analysis 36

H Limitations and Broader Impacts **38**

A Related Work

Image protection against generative models. Existing image protection methods mainly aim to protect images from two types of unauthorized use. For *tuning-based customization*, where malicious attackers collect multiple images to fine-tune a text-to-image model via DreamBooth [56] or LoRA [23], existing defenses typically add imperceptible adversarial perturbations to the images, making the fine-tuned model fail to capture the identity or artistic style [38, 60, 37, 64, 44, 69, 77, 86]. For *reference-based customization*, existing defenses mainly protect images which can be used as reference images for image-to-image (I2I) editing [4, 68, 67] and image-to-video (I2V) generation. Specifically, imperceptible adversarial perturbations are added to the reference image, leading to distorted editing results in I2I editing [57, 75, 2, 31, 62, 21] and low-quality videos or unstable motion in I2V generation [53, 15, 65, 10, 45]. However, existing works focus on protecting images, leaving video-level protection against both reference- and tuning-based customization unexplored.

Video protection against unauthorized usage. Recent works have explored video protection under different unauthorized scenarios, including object tracking [76], malicious video editing [32, 6], and style mimicry on video imagery [54]. However, these video protection methods do not address unauthorized personalized customization using video diffusion models. Notably, although Gimbal [54] targets video content, it still performs image-level protection by protecting extracted video frames from being used to fine-tune text-to-image (T2I) diffusion models.

Customizable video generation models. Recent video generation models [30, 81, 66, 17, 73, 72, 88, 14] enable the same two types of customization discussed in image-protection works above. For *tuning-based customization*, open-source models such as LTX-Video [17], Wan [66], CogVideoX [81], and HunyuanVideo [30] allow users to train LoRA adapters [23] on a few videos and then generate new videos from text prompts. For *reference-based customization*, a pretrained image-to-video (I2V) model is conditioned on a single reference image. Open I2V pipelines include LTX-2.3 I2V [18], Wan2.2-5B TI2V [66], CogVideoX-5b-I2V [81], Stable Video Diffusion [3]. Together, these tools make video customization broadly accessible and raise concerns about identity misuse, highlighting the need for video protection against diffusion-based customization.

B Extended Methodology

B.1 Analysis of Video Protection

B.1.1 Failure of Image-Level Protection on Videos

Static-content video and universal perturbations. To verify that the effect of *temporal compression caused by video VAEs* differs from that of *directly removing perturbations via pixel-space or spatial denoising*, we use two variants: (i) the static-video setting in Fig. 6; and (ii) image-level Universal Adversarial Perturbations (UAPs) in Fig. 7. For (i), all frames are constructed by repeating the first protected frame (*the same content and perturbations across frames*). For (ii), the same image-level UAP is applied to all frames (*different content but the same perturbation across frames*). As shown in Figs. 6 and 7, the reconstructed video for *static video* remains near-black but the reconstructed video for *image-level UAP* still recovers in later frames just like the naive image-level perturbation. This suggests that the recovery observed in Fig. 2(d) *does not come from the video VAE simply removing image-level perturbations from a single protected frame*. Instead, the recovery relies on temporal compression over different frames in the video, where information from previous frames can help restore the visual content. These additional results further support that frame-wise image perturbations cannot protect videos because they do not consider the temporal compression in 3D video VAEs.

Temporal receptive field analysis. We analyze the *temporal receptive field* of the 3D causal video VAE. As illustrated in Fig. 8(a), we measure the influence of each frame index t by comparing the middle-position latent of an all-black video and a single-white-frame video. Fig. 8(b) (left column) shows that earlier frames can still affect the middle temporal position, indicating that each latent is obtained through temporal compression and information aggregation over previous frames. The influence decays with temporal distance from the middle latent position, showing that temporally distant previous frames contribute less.

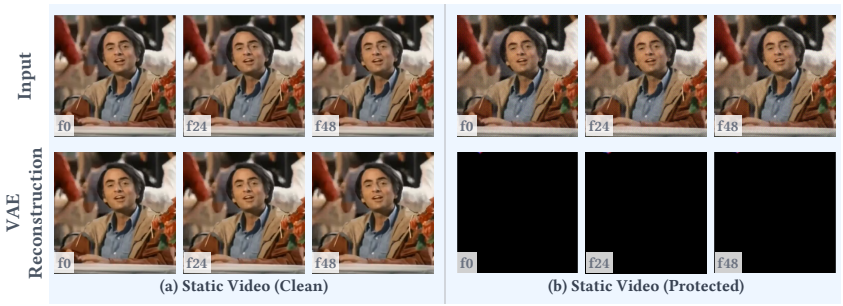


Figure 6: VAE reconstruction under the static video setting. (a) Clean static video. (b) Protected static video obtained by applying the same image protection to all frames.

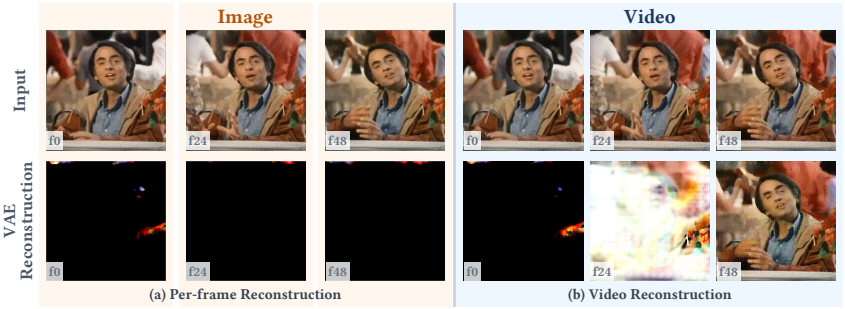


Figure 7: VAE reconstruction under the image-level UAP setting. (a) Protected frame reconstruction, where frames are extracted and reconstructed independently. (b) Video reconstruction.

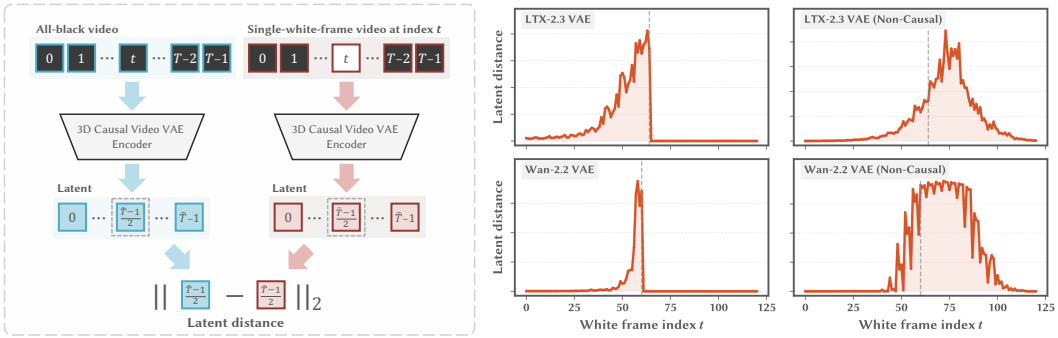


Figure 8: Temporal receptive-field analysis of 3D causal video VAEs. (a) Latent distance computation. We encode an all-black video and a single-white-frame video, in which only the t-th frame is changed to white, and compute their *latent distance at the middle temporal position*. The middle latent position is defined as the rounded-down midpoint of the latent temporal positions. (b) Temporal receptive fields of LTX-2.3 [18] and Wan2.2 [66] VAEs. The x-axis denotes the white-frame index t, and the y-axis shows the corresponding latent distance computed in (a). The left and right columns show the original causal and modified non-causal versions. Larger latent distance indicates stronger influence.

Non-causal video VAE. We provide additional results to further analyze the observation shown in Fig. 2(d). In the original LTX-2.3 VAE [18], the encoder is causal along the temporal dimension, meaning that each latent position only compresses information from the current and *previous frames* without accessing future frames. We modify the encoder into a **non-causal version** by allowing temporal convolution to also use *future frames*. Since the VAE was originally trained with a causal encoder structure, this modification introduces some reconstruction mismatch. Therefore, we first evaluate clean video reconstruction under the non-causal encoder as a reference. As shown in Fig. 9(a), the reconstructed clean video brings some distortion, but the visual content remains recognizable.

We then use the same non-causal encoder to encode the per-frame protected video. As shown in Fig. 9(b), the reconstruction recovers all content in the video, instead of only recovering in later frames as observed in Fig. 2(d). This indicates that the gradual recovery is related to the causal temporal structure of the video VAE. With non-causal encoding, earlier latent positions can also use information from later frames. Therefore, the temporal information used for reconstruction is no longer limited to previous frames, and the recovery is observed across all frames. This is because the non-causal temporal receptive field also aggregates information from later frames, as shown in Fig. 8(b) (right column).



Figure 9: VAE reconstruction with a modified non-causal VAE encoder. (a) Clean video. (b) Per-frame protected video.

B.1.2 Vulnerability to Temporal Attacks

We define the retention rate ρ as the ratio of latent disruption after and before applying a temporal attack:

$$\rho(\mathcal{T}, \delta_y) \;=\; \frac{\left\| \mathcal{E}(\mathcal{T}(V')) - \mathcal{E}(\mathcal{T}(V)) \right\|_2^2}{\left\| \mathcal{E}(V') - \mathcal{E}(V) \right\|_2^2}. \tag{16}$$

A value of $\rho \to 0$ indicates the perturbation is erased by $\mathcal{T}$, while $\rho \to 1$ indicates the perturbation remains effective.

Temporal attacks. We evaluate seven temporal attacks used in Fig. 4. Each attack is applied independently to every spatial-channel location (c, h, w) along the temporal axis of $V \in \mathbb{R}^{T \times C \times H \times W}$. For clarity, the operations in Table 4 omit the (c, h, w) subscripts, and indices outside $[0, T-1]$ are clamped to the nearest valid frame.

Table 4: Temporal attacks used in the retention analysis.

Attack	Operation
AVG-2	$\mathcal{T}(V)_t = 0.5\, V_t + 0.5\, V_{t+1}$
SKIP	$\mathcal{T}(V)_t = 0.5\, V_{t-1} + 0.5\, V_{t+1}$
AVG-5	$\mathcal{T}(V)_t = \frac{1}{5} \sum_{i=-2}^{2} V_{t+i}$
GAUSS	$\mathcal{T}(V)_t = \sum_{i=-r}^{r} g_i V_{t+i}$, where $g_i \propto \exp(-i^2/(2\sigma^2))$, $r = \lceil 3\sigma \rceil$, and $\sigma = 2$
LPF	$\mathcal{T}(V) = \mathcal{F}_t^{-1}(M_k \odot \mathcal{F}_t(V))$, where $M_k(\omega) = 1$ if $\min(\omega, T-\omega) < k$ and 0 otherwise, with $k = 8$
DOWN-2×	$\mathcal{T}(V) = \mathrm{Upsampling}_{2\times}(\mathrm{Downsampling}_{2\times}(V))$
DROP-30%	Randomly drop frames with $m_t \sim \mathrm{Bern}(1-p)$, $p = 0.3$, and fill each dropped position with its nearest kept frame

B.2 Analysis of UAP

One-frame UAP does not provide sufficient temporal expressiveness for video protection. One-frame UAP optimizes a single-frame perturbation and repeats it across all video frames. We analyze this design to test whether a one-frame UAP is sufficient for video protection.

We optimize the one-frame UAP with the same loss $\mathcal{L}_{\text{lat}}$ in Eq. (11), which maximizes the VAE latent distance between clean and protected videos. As shown in Fig. 10(a) and (c), earlier temporal latent positions reach much lower losses, and the protection mainly appears in earlier frames, while later frames remain recognizable.

To examine whether this failure is caused by optimization imbalance, we *reweight* the per-position losses by assigning each position a normalized weight inversely proportional to its loss magnitude. As shown in Fig. 10(b) and (d), reweighting makes the loss more balanced across positions and extends the disruption to later frames. However, the reconstructed video still contains recognizable content.

This indicates that the limitation of one-frame UAP is not only an optimization imbalance across temporal latent positions. It has limited *temporal expressiveness* to provide full-video protection.

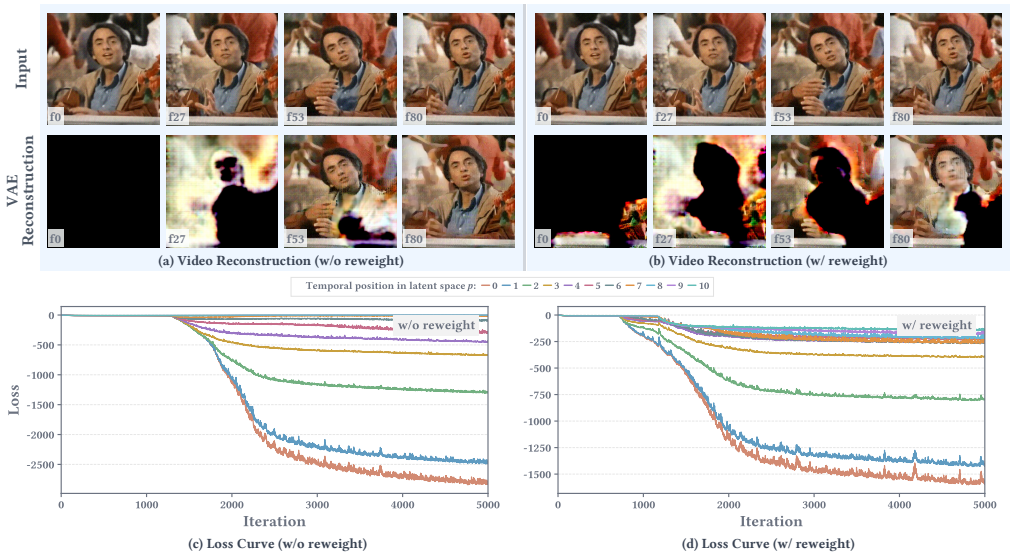


Figure 10: Analysis of one-frame UAP, where a single-frame UAP is shared across all frames and optimized against the video VAE. (a) Video reconstruction without reweight. (b) Video reconstruction with reweight. (c) Loss curves for different temporal latent positions without reweight. (d) Loss curves for different temporal latent positions with reweight.

B.3 `TC-UAP` Optimization Procedure

We summarize the optimization process in Algorithm 1.

C Extended Experimental Setup

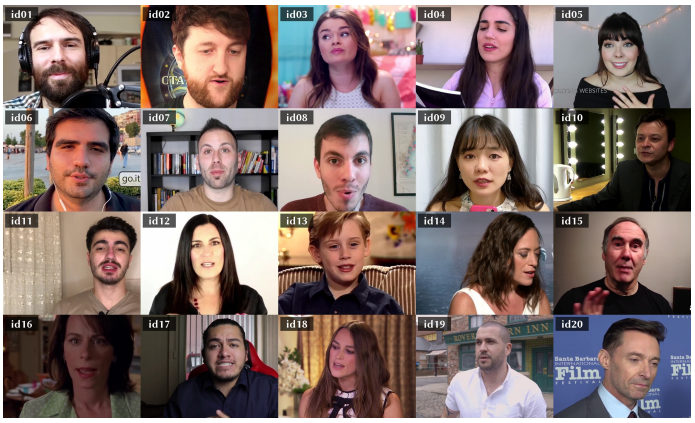
C.1 Data Preparation

To evaluate video protection against diffusion-based customization, we construct an evaluation set from three human-centered video datasets: HDTF [84], CelebV-HQ [91], and TalkVid [7]. These datasets are selected because they contain identity-centric videos, provide sufficient video samples per identity, and cover medium- to high-resolution human facial videos. All datasets are used for non-commercial research purposes only and in accordance with their respective terms of use, with their sources and licenses summarized in Table 5.

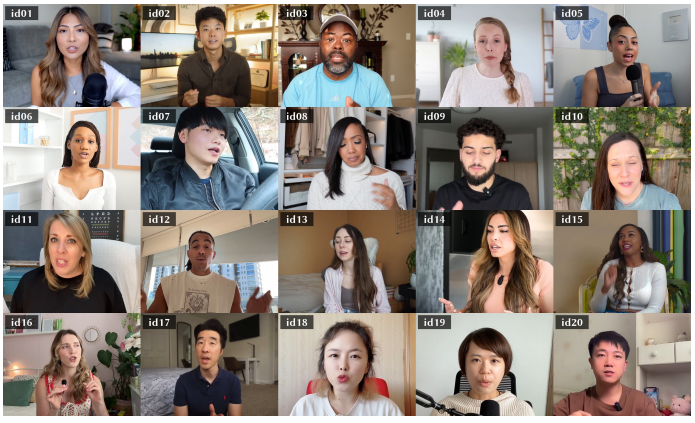
- **HDTF.** HDTF [84] is a large-scale, in-the-wild talking-face dataset collected from YouTube. It contains videos of 362 identities with a total duration of approximately 16 hours. The videos are provided at 720p or 1080p resolution and mainly contain frontal or near-frontal talking-head footage. Due to its relatively clean identity structure and high visual quality, HDTF is suitable for evaluating identity-level protection in talking-face videos.
- **CelebV-HQ.** CelebV-HQ [91] is a large-scale high-quality celebrity video dataset designed for face video generation and facial attribute analysis. It contains 35,666 video clips from 15,653



(a) HDTF [84]



(b) CelebV-HQ [91]



(c) TalkVid [7]

Figure 11: Visualization of selected identities from the three datasets. One representative frame is shown for each identity.

Algorithm 1 Training the identity-level multi-frame UAP δ_y

Require: training videos for an identity y: $\mathcal{V}_y^{\mathrm{Tr}} = \{V_i\}_{i=1}^{M_{\mathrm{Tr}}}$, multi-frame UAP length N, window length L, budget η, surrogate temporal attack $\mathcal{T}$, surrogate-loss weight λ, learning rate γ, epochs E, and R from Eq. (13).

Initialize $\theta_y \sim \mathrm{Uniform}(-0.01, 0.01)$
for $e = 1$ **to** E **do**
  for each $V_i \in \mathcal{V}_y^{\mathrm{Tr}}$ **do**
    $\delta_y \leftarrow R(\theta_y)$ ▷ Eq. (13)
    $\delta_{T_i} \leftarrow \mathrm{Repeat}_{T_i}(\delta_y)$ ▷ Eq. (9)
    $V_i' \leftarrow V_i + \delta_{T_i}$
    for each window position k sampled from V_i **do**
      $X \leftarrow V_i[k : k+L-1], \quad X' \leftarrow V_i'[k : k+L-1]$ ▷ clean / protected window
      $z \leftarrow \mathcal{E}(X), \quad z' \leftarrow \mathcal{E}(X')$
      $\mathcal{L}_{\mathrm{lat}} \leftarrow -\|z' - z\|_2^2$
      $\tilde{z} \leftarrow \mathcal{E}(\mathcal{T}(X)), \quad \tilde{z}' \leftarrow \mathcal{E}(\mathcal{T}(X'))$
      $\mathcal{L}_{\mathrm{temporal}} \leftarrow -\|\tilde{z}' - \tilde{z}\|_2^2$
      $\mathcal{L}_{\mathrm{id}} \leftarrow \mathcal{L}_{\mathrm{lat}} + \lambda \cdot \mathcal{L}_{\mathrm{temporal}}$ ▷ overall objective
      $\theta_y \leftarrow \mathrm{Adam}\big(\theta_y, \nabla_{\theta_y}\mathcal{L}_{\mathrm{id}}, \gamma\big)$ ▷ update θ_y
      $\delta_y \leftarrow R(\theta_y)$
      $\delta_y \leftarrow \mathrm{clip}(\delta_y, [-\eta, \eta])\,; \quad \theta_y \leftarrow R^{-1}(\delta_y)$ ▷ ℓ_∞ projection in δ_y-space
    end for
  end for
end for
return $\delta_y \leftarrow R(\theta_y)$

identities. Compared with HDTF, CelebV-HQ contains more diverse appearance attributes, head poses, expressions, and visual conditions, making it useful for evaluating whether a protection method can generalize across more diverse facial videos.

- **TalkVid.** TalkVid [7] is a large-scale talking-head video dataset containing 7,729 unique speakers and more than 1,244 hours of HD and 4K footage. It covers 15 languages and a broad age range. Compared with HDTF and CelebV-HQ, TalkVid provides more diverse talking-head videos in terms of speaker demographics, language, and recording scenarios.

Identity selection and preprocessing. For each dataset, we select 20 identities and split their videos into 15 training clips for UAP optimization and 15 test clips for evaluating protection effectiveness.

- **Identity selection.** For each dataset, we select 20 identities. When metadata is available, such as in CelebV-HQ and TalkVid, we prioritize diversity in gender, race, and age group. We also prefer identities whose video clips cover multiple scenes, lighting conditions, and head poses. For datasets with limited metadata, we manually inspect candidate identities and select identities with clear facial visibility and sufficient video quality. Some visualization examples are shown in Fig. 11.
- **Video preprocessing.** For each selected identity, we uniformly sample 30 video clips. Each clip is center-cropped around the face region and resized to 640×480. Each video clip contains 121 frames at 24 fps. We split the 30 clips of each identity into 15 training clips and 15 test clips. The training clips are used only for optimizing the identity-level universal adversarial perturbation (UAP), while the test clips are used for all downstream evaluations. After the UAP is trained, it is applied to the test clips to obtain the protected test set.

Table 5: Datasets used in this work and their public sources.

Dataset	Source URL	License / Terms of Use
HDTF [84]	https://github.com/MRzzm/HDTF	CC BY 4.0
CelebV-HQ [91]	https://github.com/CelebV-HQ/CelebV-HQ	–
TalkVid [7]	https://github.com/FreedomIntelligence/TalkVid	CC BY-NC 4.0

C.2 Evaluation Metrics and Protocols

We evaluate our method from two perspectives: *protection effectiveness* and *invisibility*. For *protection effectiveness*, we report VAE reconstruction metrics and identity preservation metrics. For *invisibility*, we report invisibility metrics.

VAE reconstruction metrics. Since our method targets the video VAE encoder, we need to evaluate how much the protected videos disrupt VAE reconstruction. Given a clean video V and its protected version V', we encode and decode the protected video using the VAE:

$$\hat{V}' = \mathcal{D}(\mathcal{E}(V')). \tag{17}$$

We then compare the reconstructed result $\hat{V}'$ against V using PSNR, SSIM [74], and LPIPS [83]:

- PSNR measures pixel-level reconstruction fidelity $\mathrm{PSNR}(V, \hat{V}') = 10 \log_{10} \left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}(V, \hat{V}')} \right)$, where MAX is the maximum possible pixel value and MSE is the mean squared error between the clean and reconstructed videos. Lower PSNR indicates stronger VAE disruption.
- SSIM measures structural similarity between the clean and reconstructed contents. Lower SSIM indicates that the reconstructed structure deviates more from the clean input.
- LPIPS measures perceptual distance using deep visual features; higher LPIPS indicates stronger perceptual distortion.

We report both *video-level* and *image-level* VAE reconstruction metrics, as the two customization settings take different forms of input:

- For the *tuning-based customization setting*, where the input is a set of videos, video-level metrics are computed by reconstructing each protected video through the video VAE.
- For the *reference-based customization setting*, where the input is a reference image, image-level metrics are computed by reconstructing the protected conditioning frame through the video VAE, treating it as a single-frame video.

Identity preservation metrics. For generated videos, we also need to measure whether the identity in the generated video still matches the target identity. Hence, we report Face Detection Failure Rate (FDFR) and Identity Score Matching (ISM).

- For each generated frame, we first use RetinaFace [12] to detect the face. If no face is detected, the frame is counted as a face detection failure. Given a generated video with M frames, the Face Detection Failure Rate is defined as:

$$\mathrm{FDFR} = \frac{M_{\text{fail}}}{M}, \tag{18}$$

  where M_{fail} is the number of frames where no face is detected. A higher FDFR indicates stronger protection, since the generated face becomes harder to detect.
- For frames where a face is successfully detected, we compute face embeddings using two face recognition models: ArcFace [11] and CurricularFace [25]. We report the corresponding identity similarity scores as ISM-ArcFace (ISM1) and ISM-Cur (ISM2). Given a generated frame embedding $f(\hat{I}_m)$ and a reference identity embedding e_y, the frame-level identity similarity is computed by cosine similarity:

$$s_m = \frac{f(\hat{I}_m)^\top e_y}{\|f(\hat{I}_m)\|_2 \|e_y\|_2}. \tag{19}$$

  The video-level ISM is then averaged over all frames with successfully detected faces:

$$\mathrm{ISM} = \frac{1}{|\Omega|} \sum_{m \in \Omega} s_m, \tag{20}$$

  where Ω denotes the set of generated frames with detected faces. Lower ISM indicates stronger identity protection.

For the *tuning-based customization setting*, the reference identity embedding e_y is built from the clean video set of the same identity. Specifically, we uniformly sample 16 frames from the clean videos, detect the face in each frame, extract the face embeddings, and average them to obtain the

identity reference. For the *reference-based customization setting*, the reference identity embedding is computed from the conditioning frame used by the image-to-video pipeline.

Invisibility metrics. To evaluate whether the perturbation remains visually imperceptible, we measure the perceptual quality of the protected video V' with respect to the clean video V using Video Multi-Method Assessment Fusion (VMAF) [51]. VMAF is a full-reference video quality assessment metric that predicts human-perceived video quality by fusing multiple objective quality measurements, including detail loss, structural similarity, and motion-related features.

Given a clean video V and its protected counterpart V', we compute $\mathrm{VMAF}(V, V')$. A higher VMAF score indicates that the protected video is perceptually closer to the clean video, and therefore that the perturbation is less visible. In our evaluation, VMAF is used as the primary invisibility metric because it is designed for video-level perceptual quality assessment and accounts for temporal video characteristics beyond frame-wise image similarity.

Evaluation prompts. For both tuning- and reference-based evaluations, we use the same set of five prompts to generate videos, as listed in Table 6. The token "*p3r5on*" is used as the trigger token for the target identity in tuning-based customization.

Table 6: Text prompts used in the experiments.

ID	Prompt
P_1	The *p3r5on* is licking an ice-cream cone, smiling between licks while facing the camera.
P_2	The *p3r5on* takes a bite from a slice of pizza they are holding, chews slowly, and looks back at the camera.
P_3	The *p3r5on* raises a coffee mug to their lips, takes a sip, sets it down, and continues facing the camera.
P_4	The *p3r5on* opens a small notebook, glances at a page briefly, then closes it and looks at the camera.
P_5	The *p3r5on* lifts a water bottle, takes a few sips, lowers it, and continues facing the camera.

Temporal attacks. For robustness evaluation, we use three unseen temporal attacks selected from the retention analysis in Table 4:

- AVG-5, which denotes *box filtering* with a temporal window size of 5.
- GAUSS, which denotes *Gaussian filtering* with $\sigma = 2$.
- LPF, which denotes *low-pass filtering* with $k = 8$.

These attacks are not used in the surrogate temporal-attack loss during optimization.

C.3 Baselines

Baseline selection. Official implementations of I2VGuard [15] and Anti-I2V [65] were not publicly available at the time of our experiments, which prevented us from adding them to our baselines. Moreover, both methods are designed to protect images against I2V generation, whereas our setting protects video data against both I2V and fine-tuning-based customization.

Baseline implementation. We reproduce all baselines using their official implementations, released checkpoints, and original surrogate models. All baselines and our method use the same perturbation budget $\eta = 0.1$.

- For a *baseline with a pretrained perturbation generator*, such as IDProtector [62], we independently process every frame of each of the 15 test videos. Since the generator is pretrained for image protection, adapting it to video models would require retraining it with a video-specific objective.
- For *optimization-based baselines*, including PhotoGuard [57] and Mist [37], we optimize each frame of the 15 test videos using their original objectives and surrogate models, and then concatenate the protected frames in temporal order to form the protected videos. Following the baseline implementations in I2VGuard [15] for PhotoGuard and Anti-I2V [65] for Mist, we use the original image-domain surrogate models, i.e., the Stable Diffusion VAE.

C.4 Implementation Details

UAP optimization. For each identity, we optimize an identity-level universal adversarial perturbation (UAP) on the 15 training clips. The UAP has a temporal length of 9 frames and is repeated along the

temporal dimension to protect videos of arbitrary length. The perturbation is constrained by an ℓ_∞ budget of $\eta = 0.1$.

We optimize the UAP against the video VAE encoder. The λ in Eq. (14) is set to 1.0. The learning rate is set to 4×10^{-3}, and the UAP is optimized for one epoch over the training clips. After each optimization step, the perturbation is projected back to the ℓ_∞ ball.

Tuning-based customization. For tuning-based evaluation, we fine-tune LoRA adapters on the protected test videos of each identity. The training caption uses the trigger token "*p3r5on*". We use AdamW as the optimizer with learning rate 1×10^{-4} and batch size 1. For LTX-2.3 [18], LoRA fine-tuning is performed for 1,500 steps. For Wan2.2-5B [66], LoRA fine-tuning is performed for 2,000 steps. After fine-tuning, we generate videos using the evaluation prompts described in Appendix C.2. The generated videos are evaluated using identity preservation metrics, including FDFR, ISM-ArcFace, and ISM-Cur.

Reference-based customization. For reference-based evaluation, we use the official image-to-video pipelines of LTX-2.3 and Wan2.2-5B. Each generated video is conditioned on one frame extracted from a protected test video. The same evaluation prompts are used as in the tuning-based setting. Since the reference-based pipeline directly consumes a protected frame, we additionally report image-level VAE reconstruction metrics on the conditioning frame.

Hardware. UAP optimization is conducted on a single NVIDIA RTX 4090 GPU (24G). LoRA fine-tuning and video generation are performed on a single NVIDIA RTX Pro 6000 GPU (96G).

D Additional Experimental Results

D.1 Full Results on Wan2.2

We provide the complete Wan2.2 results on all three datasets in Table 7, which show that our method provides strong protection across all datasets.

Table 7: Results of Wan2.2. **Bold** indicates the best result and underline the second-best.

Dataset	Method	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
HDTF	Clean	41.58	0.979	0.007	0.024	0.645	0.660	42.99	0.982	0.006	0.003	0.628	0.640
	PhotoGuard	30.00	0.678	0.483	0.095	0.627	0.641	30.95	0.733	**0.476**	**0.073**	0.364	0.383
	Mist	31.93	0.776	0.353	0.102	0.600	0.617	32.68	0.796	0.352	0.013	0.494	0.510
	IDProtector	29.41	0.771	0.386	0.037	0.400	0.411	**29.65**	0.788	0.386	0.010	0.491	0.515
	Ours	**13.61**	**0.552**	**0.725**	**0.945**	**0.167**	**0.171**	29.87	**0.706**	0.452	0.023	**0.354**	**0.375**
CelebV-HQ	Clean	40.72	0.977	0.009	0.138	0.661	0.680	42.14	0.981	0.007	0.028	0.581	0.605
	PhotoGuard	29.32	0.660	0.468	0.186	0.599	0.619	30.03	0.698	**0.468**	0.065	0.369	0.387
	Mist	31.57	0.761	0.342	0.133	0.589	0.606	32.28	0.778	0.341	0.027	0.451	0.472
	IDProtector	29.46	0.762	0.371	0.126	0.426	0.441	29.68	0.777	0.377	0.025	0.460	0.484
	Ours	**12.79**	**0.532**	**0.708**	**0.838**	**0.110**	**0.118**	**28.59**	**0.697**	0.453	**0.066**	**0.349**	**0.367**
TalkVid	Clean	39.32	0.973	0.008	0.057	0.675	0.694	40.47	0.976	0.007	0.022	0.600	0.622
	PhotoGuard	28.92	0.664	0.468	0.041	0.636	0.652	29.79	0.713	**0.472**	**0.046**	0.410	0.433
	Mist	30.39	0.749	0.362	0.044	0.618	0.633	31.07	0.772	0.371	0.007	0.517	0.538
	IDProtector	29.24	0.769	0.377	0.098	0.416	0.409	29.54	0.788	0.388	0.013	0.446	0.479
	Ours	**13.75**	**0.564**	**0.646**	**0.646**	**0.195**	**0.197**	**27.96**	**0.664**	0.448	0.012	**0.343**	**0.361**

D.2 Additional Evaluation Metrics

We further extend the evaluation to include *FVD* and *perceptual video-quality metrics* [27] for the videos generated through customization.

- **FVD results.** As shown in Table 8, our method achieves the highest FVD across all datasets.
- **Perceptual video-quality results.** As shown in Table 9, the results differ between the two customization settings:
  - **Reference-based customization.** Our method produces the lowest overall VBench Quality and degrades most quality dimensions.

Table 8: FVD of generated videos.

Dataset	Method	Tuning-based	Reference-based
CelebV-HQ	Clean	0.00	0.00
	IDProtector	9.16	6.40
	Mist	7.06	5.41
	PhotoGuard	6.02	5.00
	Ours	**15.15**	**15.39**
HDTF	Clean	0.00	0.00
	IDProtector	10.24	6.00
	Mist	7.50	5.12
	PhotoGuard	6.01	5.31
	Ours	**18.22**	**18.84**
TalkVid	Clean	0.00	0.00
	IDProtector	9.52	5.69
	Mist	7.28	5.15
	PhotoGuard	5.98	5.37
	Ours	**14.08**	**14.60**

- **Tuning-based customization.** Our method effectively changes the generated identity but does not consistently produce the lowest perceptual quality, which indicates that the protection is primarily reflected in identity non-preservation.

Table 9: Perceptual video-quality metrics of generated videos.

(a) Tuning-based customization.

Dataset	Method	Quality	Imaging	Aesthetic	Motion	Subject	Background
CelebV-HQ	Clean	0.825	0.660	0.530	0.993	0.911	0.936
	IDProtector	**0.795**	**0.606**	**0.479**	0.992	0.916	0.942
	Mist	0.814	0.659	0.518	0.987	0.905	0.923
	PhotoGuard	0.811	0.642	0.526	**0.981**	**0.900**	**0.921**
	Ours	0.824	0.694	0.559	0.989	0.920	0.930
HDTF	Clean	0.837	0.701	0.598	0.994	0.920	0.947
	IDProtector	**0.795**	**0.653**	**0.555**	0.994	0.936	0.948
	Mist	0.823	0.692	0.597	0.987	0.916	**0.928**
	PhotoGuard	0.814	0.678	0.609	**0.980**	**0.912**	0.929
	Ours	0.814	0.691	0.578	0.991	0.923	0.934
TalkVid	Clean	0.832	0.729	0.509	0.993	0.923	0.937
	IDProtector	**0.818**	0.720	**0.462**	0.991	0.926	0.947
	Mist	0.827	0.738	0.505	0.986	0.916	**0.926**
	PhotoGuard	0.819	0.728	0.499	**0.981**	**0.908**	0.931
	Ours	0.825	**0.709**	0.540	0.991	0.917	0.934

(b) Reference-based customization.

Dataset	Method	Quality	Imaging	Aesthetic	Motion	Subject	Background
CelebV-HQ	Clean	0.810	0.630	0.520	0.992	0.839	0.900
	IDProtector	0.789	0.646	0.530	0.991	0.743	**0.822**
	Mist	0.795	0.645	0.536	0.991	0.774	0.855
	PhotoGuard	0.797	0.635	0.514	0.991	0.793	0.861
	Ours	**0.717**	**0.553**	**0.475**	**0.982**	**0.673**	0.843
HDTF	Clean	0.821	0.678	0.574	0.994	0.848	0.895
	IDProtector	0.809	0.689	0.592	0.993	0.761	**0.833**
	Mist	0.816	0.691	0.572	0.993	0.799	0.867
	PhotoGuard	0.815	0.675	0.564	0.993	0.812	0.863
	Ours	**0.694**	**0.544**	**0.465**	**0.980**	**0.666**	0.836
TalkVid	Clean	0.821	0.716	0.496	0.993	0.852	0.902
	IDProtector	0.804	0.714	0.520	0.992	0.748	**0.824**
	Mist	0.809	0.716	0.509	0.992	0.785	0.847
	PhotoGuard	0.811	0.705	0.487	0.992	0.812	0.874
	Ours	**0.741**	**0.629**	**0.484**	**0.985**	**0.705**	0.852

Following prior work, both *visual-quality degradation* and *identity non-preservation* are effective protection outcomes. In particular, IDProtector [62] makes generated results dissimilar to the protected identity, while CLIP2Protect [59] and DiffAM [63] conceal machine-recognizable identity while maintaining natural visual quality.

Under reference-based customization, our method achieves both *identity non-preservation* and *visual-quality degradation*. Under tuning-based customization, the encoding of multiple video frames by the 3D causal video VAE attenuates the perturbation strength. Although our method substantially addresses this challenge, protection in the tuning-based setting is still reflected primarily in *identity non-preservation* rather than visual-quality degradation.

D.3 Generated Outcomes Analysis across Prompts and Seeds

While prior T2I protection methods [64] often produce *visually degraded generated outputs*, our method induces *identity swaps* in tuning-based customization settings. To better understand this phenomenon, we further evaluate identity similarity across three **prompts** (P_1, P_2, P_3) in Table 6 and three **random seeds** (S_{42}, S_{123}, S_{456}) on TalkVid dataset.

As reported in Table 10, the low cross-prompt and cross-seed similarities indicate that the generated identity is not consistently shifted toward a fixed alternative identity. Instead, it varies with the prompt and random seed, which may be consistent with the hypothesis that the model falls back on its pretrained prior when the protected videos no longer provide reliable identity information.

D.4 Protection under Challenging Video Contents

We further evaluate `TC-UAP` on clips covering more diverse video contents:

- **Multiple identities:** two-person conversational clips selected from CelebV-HQ dataset [91].
- **Non-frontal faces:** clips captured by a camera positioned 30° to the subject's left, providing a non-frontal facial view from MEAD dataset [70].
- **Occlusion, strong camera motion, and motion blur:** EgoView clips from Harmony4D dataset [29].

As shown in Table 11, our method remains highly effective under these challenging video contents.

D.5 More Qualitative Results

Figures 12 to 15 provide additional qualitative comparisons on LTX-2.3 [18] and Wan2.2-5B [66]. These results further demonstrate that `TC-UAP` provides effective protection under both tuning- and reference-based customization pipelines.



Figure 12: Qualitative comparison of customization results on the Wan2.2-5B model [66].

Table 10: Cross-prompt and cross-seed identity similarity.

(a) Tuning-based, ISM1 (ArcFace).

		Target Video								
		P_1-S_{42}	P_1-S_{123}	P_1-S_{456}	P_2-S_{42}	P_2-S_{123}	P_2-S_{456}	P_3-S_{42}	P_3-S_{123}	P_3-S_{456}
Reference Video	P_1-S_{42}	0.663	0.218	0.257	0.194	0.176	0.229	0.184	0.182	0.185
	P_1-S_{123}	0.216	0.580	0.233	0.182	0.194	0.205	0.198	0.182	0.154
	P_1-S_{456}	0.260	0.215	0.635	0.179	0.175	0.210	0.166	0.159	0.178
	P_2-S_{42}	0.192	0.175	0.206	0.665	0.235	0.218	0.173	0.162	0.146
	P_2-S_{123}	0.168	0.197	0.188	0.240	0.543	0.209	0.159	0.167	0.175
	P_2-S_{456}	0.219	0.205	0.227	0.231	0.202	0.635	0.202	0.192	0.169
	P_3-S_{42}	0.181	0.179	0.198	0.163	0.163	0.182	0.608	0.193	0.188
	P_3-S_{123}	0.198	0.196	0.183	0.170	0.178	0.194	0.201	0.631	0.219
	P_3-S_{456}	0.185	0.166	0.196	0.136	0.155	0.169	0.202	0.212	0.638

(b) Tuning-based, ISM2 (CurricularFace).

		Target Video								
		P_1-S_{42}	P_1-S_{123}	P_1-S_{456}	P_2-S_{42}	P_2-S_{123}	P_2-S_{456}	P_3-S_{42}	P_3-S_{123}	P_3-S_{456}
Reference Video	P_1-S_{42}	0.690	0.245	0.270	0.214	0.193	0.255	0.202	0.209	0.198
	P_1-S_{123}	0.234	0.607	0.244	0.193	0.211	0.226	0.204	0.204	0.163
	P_1-S_{456}	0.276	0.233	0.652	0.195	0.185	0.234	0.181	0.175	0.201
	P_2-S_{42}	0.206	0.185	0.215	0.685	0.245	0.226	0.181	0.180	0.153
	P_2-S_{123}	0.192	0.214	0.190	0.263	0.562	0.223	0.170	0.174	0.197
	P_2-S_{456}	0.230	0.220	0.238	0.226	0.213	0.652	0.221	0.210	0.179
	P_3-S_{42}	0.195	0.201	0.212	0.179	0.164	0.207	0.629	0.218	0.193
	P_3-S_{123}	0.214	0.217	0.197	0.185	0.187	0.215	0.212	0.655	0.222
	P_3-S_{456}	0.208	0.177	0.220	0.141	0.175	0.182	0.199	0.222	0.663

(c) Reference-based, ISM1 (ArcFace).

		Target Video								
		P_1-S_{42}	P_1-S_{123}	P_1-S_{456}	P_2-S_{42}	P_2-S_{123}	P_2-S_{456}	P_3-S_{42}	P_3-S_{123}	P_3-S_{456}
Reference Video	P_1-S_{42}	0.763	0.305	0.295	0.223	0.211	0.185	0.265	0.221	0.212
	P_1-S_{123}	0.288	0.698	0.254	0.190	0.210	0.192	0.219	0.242	0.224
	P_1-S_{456}	0.294	0.267	0.755	0.161	0.157	0.174	0.219	0.196	0.267
	P_2-S_{42}	0.202	0.165	0.150	0.569	0.192	0.175	0.214	0.180	0.153
	P_2-S_{123}	0.209	0.213	0.169	0.226	0.518	0.220	0.222	0.232	0.214
	P_2-S_{456}	0.192	0.195	0.181	0.208	0.209	0.542	0.209	0.190	0.198
	P_3-S_{42}	0.271	0.240	0.250	0.230	0.210	0.188	0.599	0.253	0.262
	P_3-S_{123}	0.229	0.252	0.198	0.217	0.234	0.228	0.287	0.562	0.273
	P_3-S_{456}	0.235	0.243	0.256	0.209	0.206	0.217	0.268	0.274	0.517

(d) Reference-based, ISM2 (CurricularFace).

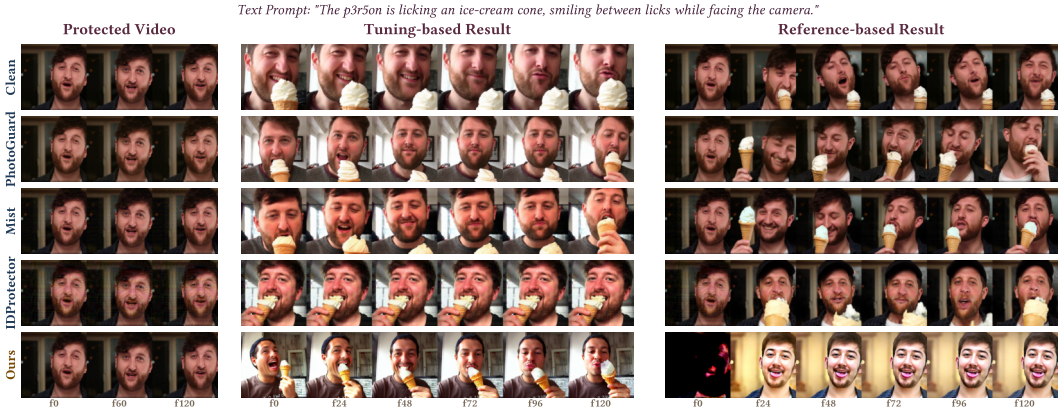
		Target Video								
		P_1-S_{42}	P_1-S_{123}	P_1-S_{456}	P_2-S_{42}	P_2-S_{123}	P_2-S_{456}	P_3-S_{42}	P_3-S_{123}	P_3-S_{456}
Reference Video	P_1-S_{42}	0.777	0.319	0.292	0.227	0.223	0.190	0.262	0.235	0.213
	P_1-S_{123}	0.295	0.723	0.263	0.194	0.226	0.200	0.239	0.259	0.242
	P_1-S_{456}	0.276	0.272	0.775	0.183	0.169	0.187	0.228	0.205	0.281
	P_2-S_{42}	0.205	0.171	0.172	0.592	0.193	0.192	0.230	0.196	0.168
	P_2-S_{123}	0.222	0.226	0.172	0.241	0.540	0.241	0.239	0.245	0.227
	P_2-S_{456}	0.196	0.195	0.187	0.215	0.206	0.563	0.219	0.198	0.203
	P_3-S_{42}	0.276	0.249	0.258	0.251	0.225	0.207	0.619	0.252	0.285
	P_3-S_{123}	0.253	0.267	0.212	0.245	0.252	0.231	0.306	0.597	0.286
	P_3-S_{456}	0.241	0.242	0.270	0.217	0.213	0.236	0.282	0.279	0.546

Table 11: Protection performance under challenging video contents.

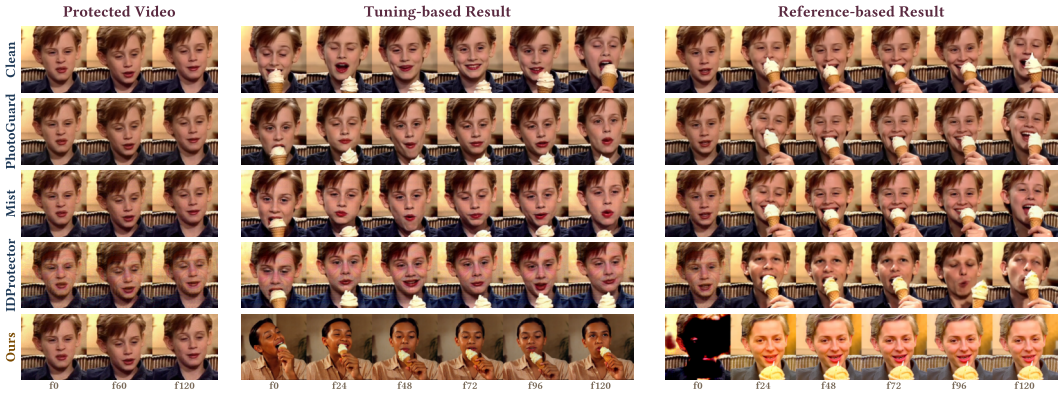
Dataset	Method	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
CelebV-HQ	Clean	30.96	0.917	0.022	0.000	0.374	0.405	31.89	0.928	0.018	0.000	0.756	0.762
	Ours	**11.11**	**0.212**	**0.712**	**0.000**	**0.015**	**0.022**	**13.58**	**0.274**	**0.625**	**0.000**	**0.356**	**0.369**
MEAD	Clean	42.52	0.988	0.008	0.000	0.697	0.681	44.08	0.991	0.006	0.000	0.635	0.625
	Ours	**14.69**	**0.243**	**0.627**	**0.055**	**0.021**	**0.020**	**13.47**	**0.096**	**0.573**	**0.721**	**0.179**	**0.143**
Harmony4D	Clean	36.89	0.963	0.053	0.164	0.584	0.626	41.10	0.985	0.015	0.000	0.342	0.399
	Ours	**12.37**	**0.203**	**0.738**	**0.000**	**0.068**	**0.087**	**11.07**	**0.054**	**0.792**	**0.985**	**0.134**	**0.149**



(a)

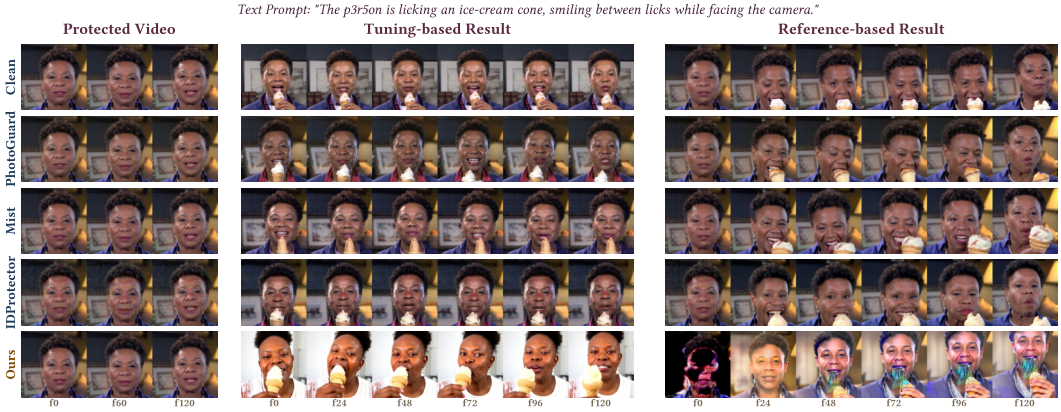


(b)

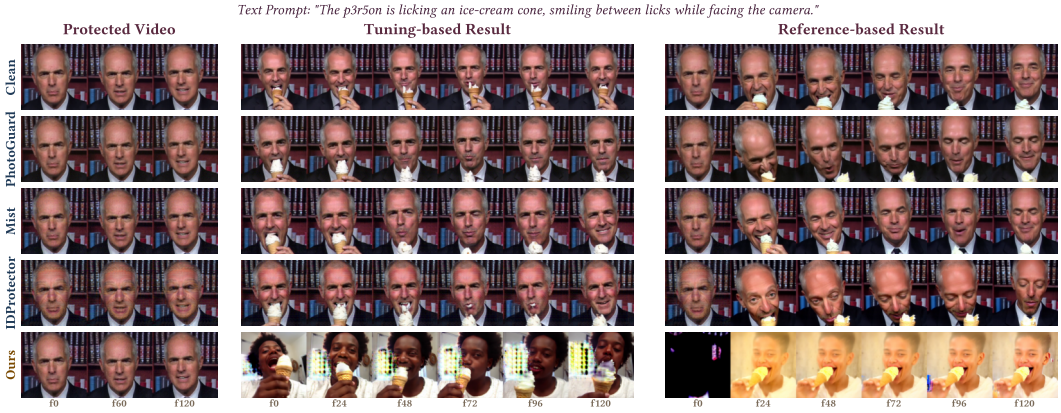


(c)

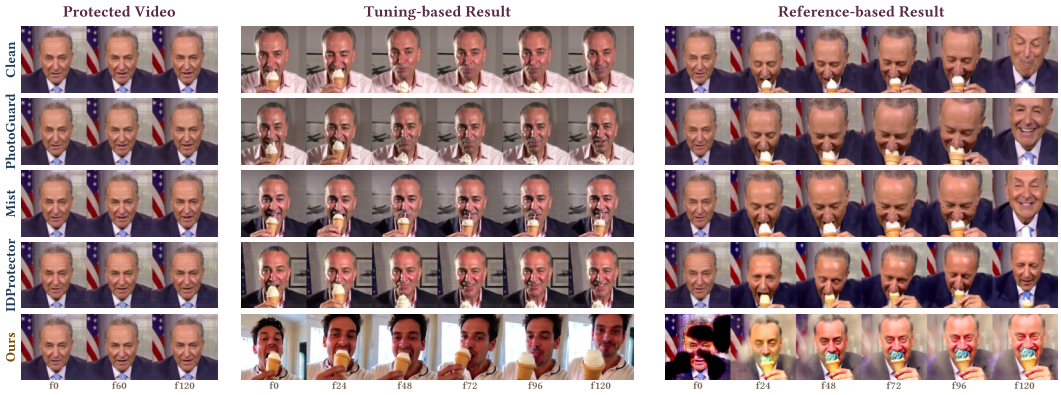
Figure 13: Qualitative comparison on CelebV-HQ Dataset.



(a)



(b)



(c)

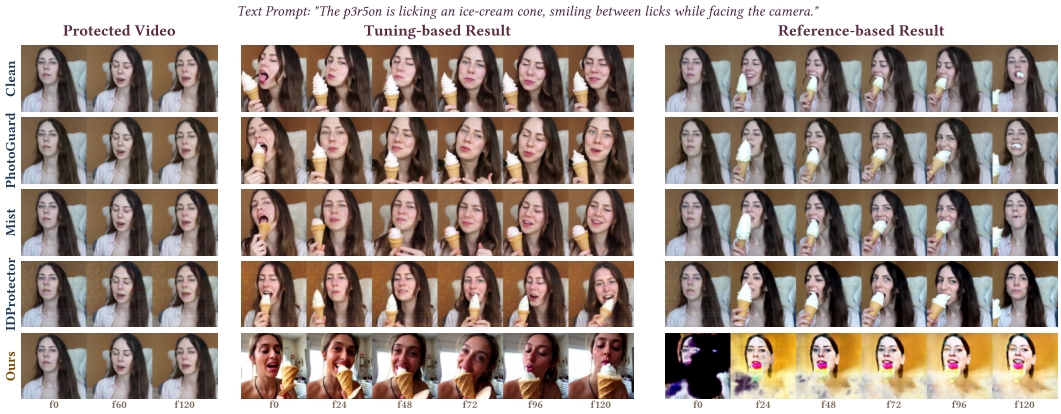
Figure 14: Qualitative comparison on HDTF Dataset.



(a)



(b)



(c)

Figure 15: Qualitative comparison on TalkVid Dataset.

E Ablation Study

We first conduct ablation studies on the two temporal designs: the intrinsic temporal modeling and the surrogate temporal-attack loss. Results are shown in Table 12.

Tuning-based pipeline. Removing the intrinsic consistency design strengthens protection on the original videos (PSNR 6.47, ISM1 0.010). However, once any temporal filter is applied, protection collapses with ISM1 rising to 0.224, 0.313, and 0.300 under the three filters, far higher than ours (0.136, 0.111, 0.097). Removing the surrogate loss also has an effect: protection on the original videos is stronger than ours, but under filters our full method consistently wins. Both ablations show that each design trades a small amount of protection strength for a substantial gain in robustness.

Reference-based pipeline. The reference-based pipeline shows a different pattern: our full method performs best overall on both the original protected videos and under filters. The reason is that the reference-based pipeline only consumes a single frame extracted from the protected video, so what matters is per-frame attack strength. Our temporal-consistency designs spread the perturbation evenly along time, so any extracted frame carries a strong attack. Without intrinsic consistency, the perturbation concentrates on a few frames and the extracted one may not be among them; this hurts even the original protection (PSNR 14.08 vs. ours 8.00, ISM1 0.415 vs. ours 0.275) and degrades further under filters. Without the surrogate loss, performance is close to ours under most settings, with the surrogate giving a clear advantage on the low-pass filter (ISM1 0.077 vs. 0.093). Consequently, our temporal-consistency designs add robustness without sacrificing per-frame attack strength.

Table 12: Ablation study on TalkVid dataset. **ITM** denotes the Intrinsic Temporal Modeling reparameterization (Eq. 13), and **STA** denotes the Surrogate Temporal-Attack loss (Eq. 12). ✓/✗ indicates whether the corresponding module is enabled.

Operation	ITM	STA	Tuning-based						Reference-based					
			VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
			PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Original	✓	✗	11.46	0.318	0.682	0.003	0.061	0.071	8.08	0.191	0.708	0.084	**0.232**	**0.240**
	✗	✓	**6.47**	**0.082**	**0.826**	0.000	**0.010**	**0.017**	14.08	0.358	0.609	0.008	0.415	0.425
	✓	✓	12.17	0.381	0.637	**0.013**	0.160	0.164	**8.00**	**0.187**	**0.709**	**0.146**	0.275	0.286
Box Filter	✓	✗	15.30	0.444	0.639	**0.013**	0.173	0.182	9.18	0.224	0.688	0.111	0.162	0.173
	✗	✓	18.50	0.501	0.574	0.000	0.224	0.239	14.93	0.415	0.607	0.056	0.320	0.333
	✓	✓	**14.66**	**0.443**	**0.647**	0.001	**0.136**	**0.141**	**8.92**	**0.223**	**0.708**	**0.168**	**0.151**	**0.156**
Gaussian Filter	✓	✗	15.36	0.437	0.656	0.013	0.167	0.181	9.17	0.221	0.687	0.105	**0.152**	**0.167**
	✗	✓	20.96	0.599	0.524	0.000	0.313	0.327	15.59	0.434	0.599	0.050	0.358	0.372
	✓	✓	**14.49**	**0.432**	**0.681**	**0.013**	**0.111**	**0.117**	**8.91**	**0.217**	**0.715**	**0.165**	0.164	0.171
Low-pass Filter	✓	✗	15.05	0.436	0.657	0.006	0.165	0.177	8.95	0.201	0.731	0.201	0.093	0.099
	✗	✓	21.35	0.600	0.553	0.000	0.300	0.314	15.29	0.426	0.628	0.060	0.217	0.237
	✓	✓	**14.23**	**0.426**	**0.685**	**0.017**	**0.097**	**0.107**	**8.57**	**0.197**	**0.754**	**0.220**	**0.077**	**0.071**

To isolate the contribution of the temporal design, we further add two one-frame UAP baselines, reported in Table 13, with the following settings:

- **Identity-level one-frame UAP:** One-frame perturbation is optimized for each identity over the same 15 training videos used by our method, and is then directly applied to all frames of the corresponding test videos.
- **Per-video one-frame UAP:** A separate one-frame perturbation is optimized directly on each of the 15 test videos and then broadcast to all frames of the corresponding video.

Both baselines are optimized over the full video without sliding-window optimization and use the same video VAE and perturbation budget as our method.

As shown in Table 13, both one-frame baselines are substantially less effective than ours in the original setting, indicating that a broadcast one-frame UAP does not provide sufficient temporal expressiveness to effectively disrupt video-VAE latent representations. Since the same one-frame perturbation is repeated across all frames, temporal filtering has limited effect on these baselines. Their robustness to purification and black-box transfer is also comparable to ours.

In contrast, our multi-frame design provides substantially greater temporal expressiveness while retaining robustness and transferability comparable to the one-frame UAPs. This demonstrates that the proposed temporal design improves protection effectiveness while preserving robustness.

Table 13: Ablation Study on the Temporal Design.

Method	Setting	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Clean	Original	36.94	0.960	0.018	0.000	0.623	0.636	39.19	0.971	0.013	0.000	0.628	0.650
Ours	Original	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
	Box Filter	14.66	0.443	0.647	0.001	0.136	0.141	8.92	0.223	0.708	0.168	0.151	0.156
	Gaussian Filter	14.49	0.432	0.681	0.013	0.111	0.117	8.91	0.217	0.715	0.165	0.164	0.171
	Low-pass Filter	14.23	0.426	0.685	0.017	0.097	0.107	8.57	0.197	0.754	0.220	0.077	0.071
	Purification	30.90	0.874	0.273	0.000	0.545	0.544	30.29	0.846	0.368	0.012	0.569	0.590
	Transferability	28.00	0.558	0.561	0.070	0.583	0.596	29.95	0.669	0.496	0.028	0.465	0.491
Identity-level one-frame UAP	Original	25.40	0.544	0.600	0.002	0.393	0.407	25.91	0.543	0.625	0.090	0.375	0.395
	Box Filter	25.79	0.544	0.649	0.026	0.318	0.332	26.04	0.540	0.661	0.115	0.305	0.318
	Gaussian Filter	25.90	0.545	0.658	0.054	0.312	0.326	26.01	0.539	0.659	0.125	0.311	0.325
	Low-pass Filter	25.76	0.541	0.660	0.056	0.322	0.340	26.01	0.529	0.703	0.098	0.186	0.199
	Purification	30.58	0.858	0.345	0.000	0.604	0.616	29.75	0.821	0.451	0.006	0.563	0.594
	Transferability	28.16	0.616	0.536	0.053	0.557	0.567	29.36	0.687	0.497	0.009	0.426	0.452
Per-video one-frame UAP	Original	25.53	0.558	0.594	0.000	0.341	0.357	26.10	0.559	0.617	0.131	0.352	0.365
	Box Filter	26.05	0.560	0.644	0.005	0.287	0.304	26.29	0.556	0.655	0.153	0.313	0.326
	Gaussian Filter	26.19	0.562	0.654	0.000	0.274	0.288	26.25	0.554	0.652	0.174	0.313	0.327
	Low-pass Filter	26.07	0.558	0.655	0.003	0.295	0.309	26.35	0.545	0.700	0.107	0.198	0.215
	Purification	30.61	0.860	0.328	0.000	0.622	0.635	29.84	0.825	0.432	0.001	0.560	0.590
	Transferability	28.41	0.629	0.529	0.048	0.532	0.547	29.67	0.703	0.489	0.016	0.458	0.480

F Analysis of `TC-UAP` Hyperparameters

We analyze the influence of different hyperparameters for `TC-UAP`, including the *perturbation budget* η, *UAP length* N, the *temporal-attack loss weight* λ, and the *sliding-window length* L.

All experiments in this section are conducted on 20 identities from TalkVid [7] using LTX-2.3 [18], and all metrics are reported as means across these identities. The default configuration is $N = 9$, $L = 25$, $\lambda = 1$, and $\eta = 0.1$.

F.1 Perturbation Budget η

As shown in Table 14, the smaller budget $\eta = 8/255$ still provides strong protection under both customization settings. Our default perturbation budget is $\eta = 0.1$, equivalent to $25.5/255$ for videos normalized to $[0, 1]$.

Table 14: Sensitivity analysis of the perturbation budget η.

η	Tuning-based						Reference-based					
	VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Clean	36.94	0.960	0.018	0.000	0.623	0.636	39.19	0.971	0.013	0.000	0.628	0.650
$25.5/255$	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
$8/255$	14.31	0.497	0.540	0.029	0.139	0.154	11.30	0.384	0.576	0.120	0.303	0.320

F.2 UAP Length N

As shown in Table 15, shorter UAP periods provide insufficient temporal expressiveness, whereas longer periods improve temporal expressiveness but increase the per-identity storage overhead.

F.3 Temporal-Attack Loss Weight λ

As shown in Table 16, the effectiveness of our method remains stable across the evaluated range of λ. As shown by the Original rows, different values of λ produce relatively similar protection performance. Besides, the three weight settings achieve comparable robustness under the box, Gaussian, and low-pass filters.

Table 15: Sensitivity analysis of the UAP length N.

N	Tuning-based						Reference-based					
	VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
1	23.06	0.556	0.613	0.006	0.433	0.444	18.40	0.462	0.618	0.010	0.484	0.490
9	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
33	11.89	0.374	0.632	0.093	0.044	0.049	7.07	0.140	0.732	0.322	0.098	0.102

Table 16: Sensitivity analysis of the temporal-attack loss weight λ.

λ	Operation	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
0.5	Original	12.05	0.376	0.641	0.019	0.167	0.172	7.91	0.181	0.714	0.158	0.284	0.294
	Box Filter	14.91	0.447	0.653	0.005	0.140	0.147	8.97	0.222	0.713	0.183	0.144	0.149
	Gaussian Filter	14.72	0.435	0.692	0.019	0.112	0.119	8.96	0.216	0.721	0.180	0.159	0.166
	Low-pass Filter	14.42	0.428	0.696	0.024	0.096	0.107	8.57	0.193	0.766	0.243	0.059	0.051
1	Original	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
	Box Filter	14.66	0.443	0.647	0.001	0.136	0.141	8.92	0.223	0.708	0.168	0.151	0.156
	Gaussian Filter	14.49	0.432	0.681	0.013	0.111	0.117	8.91	0.217	0.715	0.165	0.164	0.171
	Low-pass Filter	14.23	0.426	0.685	0.017	0.097	0.107	8.57	0.197	0.754	0.220	0.077	0.071
2	Original	12.26	0.386	0.634	0.018	0.151	0.157	8.08	0.191	0.706	0.139	0.257	0.269
	Box Filter	14.38	0.439	0.643	0.008	0.131	0.137	8.86	0.222	0.705	0.158	0.161	0.168
	Gaussian Filter	14.23	0.429	0.671	0.018	0.109	0.117	8.85	0.217	0.711	0.155	0.172	0.180
	Low-pass Filter	14.01	0.424	0.675	0.021	0.097	0.109	8.56	0.200	0.744	0.202	0.098	0.095

F.4 Sliding-Window Length L

As shown in Table 17, a longer sliding window generally provides stronger protection. For all evaluated window lengths, the protection remains effective after temporal filtering compared with the clean baseline.

Table 17: Sensitivity analysis of the sliding-window length L.

L	Operation	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
9	Original	21.19	0.572	0.593	0.083	0.387	0.393	17.49	0.461	0.628	0.079	0.406	0.425
	Box Filter	23.87	0.616	0.560	0.070	0.474	0.483	18.31	0.552	0.554	0.074	0.485	0.487
	Gaussian Filter	23.77	0.613	0.572	0.014	0.454	0.461	18.29	0.550	0.555	0.017	0.471	0.469
	Low-pass Filter	23.88	0.614	0.575	0.020	0.459	0.472	18.24	0.540	0.582	0.037	0.440	0.530
25	Original	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
	Box Filter	14.66	0.443	0.647	0.001	0.136	0.141	8.92	0.223	0.708	0.168	0.151	0.156
	Gaussian Filter	14.49	0.432	0.681	0.013	0.111	0.117	8.91	0.217	0.715	0.165	0.164	0.171
	Low-pass Filter	14.23	0.426	0.685	0.017	0.097	0.107	8.57	0.197	0.754	0.220	0.077	0.071
41	Original	8.10	0.184	0.745	0.058	0.007	0.015	7.31	0.143	0.722	0.280	0.183	0.194
	Box Filter	11.90	0.387	0.650	0.112	0.038	0.043	7.00	0.129	0.752	0.468	0.100	0.118
	Gaussian Filter	12.04	0.382	0.673	0.170	0.049	0.054	7.02	0.130	0.751	0.475	0.097	0.105
	Low-pass Filter	11.70	0.371	0.681	0.192	0.033	0.040	6.91	0.117	0.784	0.498	0.055	0.050

We further report the normalized protection retention in Table 18, defined as $\text{Retention} = \frac{m_{\text{post}} - m_{\text{clean}}}{m_{\text{pre}} - m_{\text{clean}}}$, where m_{post} denotes the post-attack metric value averaged over the three filters, m_{pre} denotes the corresponding pre-attack metric value, and m_{clean} denotes the metric value obtained by performing the customization on the clean video set. Besides, $L = 9$ generally exhibits lower normalized protection retention, while $L = 25$ and $L = 41$ show broadly comparable and stronger retention.

Computational efficiency. As shown in Table 19, the temporal-attack loss weight λ does not affect optimization efficiency, whereas a larger sliding-window length L increases the *per-step optimization time* and *GPU memory usage*.

Table 18: Normalized protection retention under different sliding-window lengths.

Setting	L	VAE Reconstruction			Identity Preservation		
		PSNR Retention↑	SSIM Retention↑	LPIPS Retention↑	FDFR Retention↑	ISM1 Retention↑	ISM2 Retention↑
Tuning-based	9	0.832	0.891	0.958	0.418	0.681	0.675
	25	0.908	0.909	1.055	0.795	1.098	1.090
	41	0.869	0.747	0.894	2.724	0.946	0.951
Reference-based	9	0.964	0.831	0.895	0.540	0.733	0.687
	25	0.974	0.968	1.024	1.263	1.409	1.421
	41	1.010	1.021	1.057	1.715	1.222	1.226

Table 19: Computational efficiency of TC-UAP.

Hyperparameter	Value	Per-step Time (s)	GPU Memory (GiB)
λ	0.5	1.477	16.70
	1	1.477	16.70
	2	1.477	16.70
L	9	0.589	7.49
	25	1.477	16.70
	41	3.141	25.53

G Transferability and Robustness Evaluation

G.1 Evaluation of TC-UAP

Black-box transferability. We conduct additional experiments to further validate the transferability of our method (LTX → Wan). As shown in Table 20, our method exhibits a certain degree of black-box transferability across different video VAEs and customization models.

Table 20: Black-box transferability across different customization models.

Method	Setting	Tuning-based						Reference-based					
		VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
		PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Clean	Wan	39.32	0.973	0.008	0.057	0.675	0.694	40.47	0.976	0.007	0.022	0.600	0.622
Random Noise	Wan	32.24	0.763	0.368	0.053	0.676	0.688	33.45	0.811	0.336	0.006	0.565	0.587
Ours	LTX → Wan	**28.00**	**0.558**	**0.561**	**0.070**	**0.583**	**0.596**	**29.95**	**0.669**	**0.496**	**0.028**	**0.465**	**0.491**

Robustness. We further evaluate robustness under (i) *additional temporal transformations*, including RIFE-based frame interpolation [26], temporal shuffling, frame-rate resampling, frame skipping, temporal downsampling, and random frame dropping, (ii) *different customization settings*, including shorter and longer video lengths, and (iii) *purification and compression*, including DiffPure [52], JPEG compression, and H.264 video recompression.

As shown in Table 21, TC-UAP remains effective across these operations. Under temporal transformations and video-length variations, the protected videos continue to exhibit substantial VAE reconstruction degradation and low identity preservation under both tuning- and reference-based customization. Purification and compression partially weaken the protection, but the identity-preservation scores remain lower than those of the clean setting, indicating that the protection remains effective.

G.2 Baseline Comparison and Analysis

We add experiments on IDProtector [62] for (a) *the LTX → Wan transfer setting* and (b) *the DiffPure and JPEG settings*, as reported in Tables 22 and 23.

For **black-box transferability**, our method generally causes the strongest disruption to the unseen Wan VAE representations, while our identity-protection performance is weaker than the IDProtector.

For **robustness**, our method retains stronger reconstruction-level disruption under JPEG compression. IDProtector achieves lower ISM values in the DiffPure and JPEG settings.

Table 21: Robustness under additional temporal transformations, different customization settings, purification and compression.

Operation	Tuning-based						Reference-based					
	VAE Reconstruction			Identity Preservation			VAE Reconstruction			Identity Preservation		
	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓	PSNR↓	SSIM↓	LPIPS↑	FDFR↑	ISM1↓	ISM2↓
Clean	36.94	0.960	0.018	0.000	0.623	0.636	39.19	0.971	0.013	0.000	0.628	0.650
Original	12.17	0.381	0.637	0.013	0.160	0.164	8.00	0.187	0.709	0.146	0.275	0.286
RIFE [26]	15.99	0.489	0.567	0.010	0.195	0.206	9.44	0.241	0.650	0.129	0.289	0.298
Temporal shuffling	16.00	0.471	0.585	0.020	0.182	0.191	9.42	0.239	0.651	0.124	0.250	0.259
FPS resampling	14.15	0.440	0.596	0.001	0.132	0.147	9.41	0.239	0.651	0.145	0.262	0.274
Frame skipping	15.54	0.458	0.614	0.004	0.164	0.170	9.46	0.241	0.653	0.127	0.252	0.259
Down-2×	15.54	0.457	0.598	0.001	0.160	0.171	9.45	0.241	0.652	0.127	0.252	0.259
Drop-30%	15.39	0.460	0.589	0.010	0.161	0.168	9.44	0.240	0.651	0.145	0.266	0.275
Video length 97f	14.11	0.437	0.598	0.018	0.143	0.152	9.41	0.239	0.651	0.144	0.243	0.253
Video length 145f	15.18	0.450	0.601	0.021	0.199	0.210	9.42	0.240	0.650	0.140	0.257	0.268
DiffPure [52]	30.90	0.874	0.273	0.000	0.545	0.544	30.29	0.846	0.368	0.012	0.569	0.590
JPEG compression	28.08	0.593	0.527	0.000	0.528	0.539	28.46	0.588	0.537	0.002	0.556	0.570
H.264 recompression	23.66	0.526	0.573	0.010	0.470	0.485	18.69	0.442	0.604	0.006	0.466	0.483

Table 22: Black-Box Transferability.

(a) Tuning-based.

Method	VAE Reconstruction			Identity Preservation		
	PSNR ↓	SSIM ↓	LPIPS ↑	FDFR ↑	ISM1 ↓	ISM2 ↓
Clean	39.32	0.973	0.008	0.057	0.675	0.694
Random Noise	32.24	0.763	0.368	0.053	0.676	0.688
IDProtector	29.24	0.769	0.377	**0.098**	**0.416**	**0.409**
Ours	**28.00**	**0.558**	**0.561**	0.070	0.583	0.596

(b) Reference-based.

Method	VAE Reconstruction			Identity Preservation		
	PSNR ↓	SSIM ↓	LPIPS ↑	FDFR ↑	ISM1 ↓	ISM2 ↓
Clean	40.47	0.976	0.007	0.022	0.600	0.622
Random Noise	33.45	0.811	0.336	0.006	0.565	0.587
IDProtector	**29.54**	0.788	0.388	0.013	**0.446**	**0.479**
Ours	29.95	**0.669**	**0.496**	**0.028**	0.465	0.491

Table 23: Robustness.

(a) Tuning-based.

Attack	Method	VAE Reconstruction			Identity Preservation		
		PSNR ↓	SSIM ↓	LPIPS ↑	FDFR ↑	ISM1 ↓	ISM2 ↓
–	Clean	36.94	0.960	0.018	0.000	0.623	0.636
Original	IDProtector	28.69	0.751	0.395	0.000	0.431	0.463
	Ours	**12.17**	**0.381**	**0.637**	**0.013**	**0.160**	**0.164**
DiffPure	IDProtector	**28.87**	**0.829**	**0.389**	0.000	**0.424**	**0.422**
	Ours	30.90	0.874	0.273	0.000	0.545	0.544
JPEG	IDProtector	28.67	0.755	0.391	0.000	**0.489**	**0.477**
	Ours	**28.08**	**0.593**	**0.527**	0.000	0.528	0.539

(b) Reference-based.

Attack	Method	VAE Reconstruction			Identity Preservation		
		PSNR ↓	SSIM ↓	LPIPS ↑	FDFR ↑	ISM1 ↓	ISM2 ↓
–	Clean	39.19	0.971	0.013	0.000	0.628	0.650
Original	IDProtector	29.10	0.765	0.403	0.001	0.310	0.327
	Ours	**8.00**	**0.187**	**0.709**	**0.146**	**0.275**	**0.286**
DiffPure	IDProtector	**28.32**	**0.793**	**0.483**	0.008	**0.353**	**0.380**
	Ours	30.29	0.846	0.368	**0.012**	0.569	0.590
JPEG	IDProtector	29.08	0.774	0.396	0.001	**0.354**	**0.362**
	Ours	**28.46**	**0.588**	**0.537**	**0.002**	0.556	0.570

However, we find that the performance of IDProtector may not arise solely from its ability to induce genuine identity changes in generated videos. Since its optimization objective explicitly involves face encoders, the learned perturbations may also directly interfere with the face representations extracted by these encoders, potentially inflating its measured protection performance.

To better understand this result, we separately report results under the original **non-adversarially robust face encoders** ArcFace [11] (ISM1) and CurricularFace [25] (ISM2) in Table 24 and **six adversarially robust face encoders** [80] in Table 25.

Under *ArcFace* and *CurricularFace*, IDProtector substantially reduces the measured face similarity even for the protected input videos. This is unexpected, because the perturbations should ideally affect identity information through the downstream generation process, rather than directly causing a marked change in the encoder-based similarity scores of the protected inputs, particularly under a limited perturbation budget. More importantly, the encoder-targeted interference induced by IDProtector may propagate through the generation model and persist in the generated videos, thereby contributing to an overestimation of its actual protection performance under the ISM1 and ISM2 metrics.

Table 24: Face Similarity with Non-Adversarially Robust Face Encoders.

(a) Tuning-based.

Method	Protected Video		Videos Generated by LTX-2.3		Videos Generated by Wan2.2	
	ArcFace	CurricularFace	ArcFace	CurricularFace	ArcFace	CurricularFace
Clean	0.900	0.912	0.623	0.636	0.675	0.694
IDProtector	**0.058**	**0.063**	0.431	0.463	0.416	0.409
Ours	0.855	0.865	**0.160**	**0.164**	**0.195**	**0.197**

(b) Reference-based.

Method	Protected Video		Videos Generated by LTX-2.3		Videos Generated by Wan2.2	
	ArcFace	CurricularFace	ArcFace	CurricularFace	ArcFace	CurricularFace
Clean	1.000	1.000	0.628	0.650	0.600	0.622
IDProtector	**0.068**	**0.076**	0.310	0.327	0.446	0.479
Ours	0.922	0.924	**0.275**	**0.286**	**0.343**	**0.361**

When evaluated using the *six adversarially robust face encoders*, the measured similarity of IDProtector increases substantially for both the protected input videos and the generated output videos. This result indicates that its previously observed advantage is partly attributable to direct interference with the face-encoder-based evaluation pipeline, rather than solely to genuine identity disruption induced through the generation process.

In contrast, our method yields comparable ISM values for the protected input videos across both non-robust and adversarially robust face encoders, while consistently achieving lower identity similarity than IDProtector for the generated videos under the six robust encoders.

Table 25: Face Similarity with Adversarially Robust Face Encoders from RobFR [80].

(a) **Tuning-based: Protected Video.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	0.910	0.895	0.908	0.962	0.862	0.902
IDProtector	**0.869**	**0.851**	**0.868**	**0.929**	**0.812**	**0.852**
Ours	0.908	0.893	0.906	0.962	0.859	0.899

(b) **Reference-based: Protected Video.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	1.000	1.000	1.000	1.000	1.000	1.000
IDProtector	**0.946**	**0.935**	**0.939**	**0.966**	**0.916**	**0.930**
Ours	0.995	0.995	0.995	0.998	0.993	0.994

(c) **Tuning-based: Videos Generated by LTX-2.3.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	0.811	0.784	0.813	0.907	0.727	0.800
IDProtector	0.800	0.776	0.804	0.893	0.716	0.779
Ours	**0.427**	**0.399**	**0.506**	**0.578**	**0.334**	**0.421**

(d) **Reference-based: Videos Generated by LTX-2.3.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	0.712	0.654	0.679	0.860	0.563	0.665
IDProtector	0.671	0.627	0.671	0.821	0.560	0.643
Ours	**0.508**	**0.451**	**0.537**	**0.650**	**0.394**	**0.491**

(e) **Tuning-based: Videos Generated by Wan2.2.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	0.689	0.642	0.679	0.843	0.555	0.663
IDProtector	0.732	0.702	0.729	0.861	0.627	0.713
Ours	**0.381**	**0.326**	**0.440**	**0.536**	**0.293**	**0.377**

(f) **Reference-based: Videos Generated by Wan2.2.**

Method	TRADES	PGD-Soft	PGD-AM	PGD-Sphere	PGD-Cos	PGD-Arc
Clean	0.740	0.696	0.733	0.860	0.637	0.717
IDProtector	0.745	0.714	0.760	0.842	0.681	0.742
Ours	**0.559**	**0.509**	**0.573**	**0.500**	**0.556**	**0.545**

These results suggest that the strong performance of IDProtector under certain non-robust face encoders is partly affected by *encoder-targeted interference*, whereas our method induces more *genuine identity disruption* through the generation process, with consistent effects across a diverse set of face encoders.

H Limitations and Broader Impacts

Limitations. As the first study on video protection against diffusion-based customization, we adopt a white-box setting to enable a detailed analysis of video-specific mechanisms. However, our method exhibits limited cross-model transferability. Moreover, a person's identity in video content includes acoustic characteristics (e.g., voice and speech patterns). Our method does not prevent attackers

from cloning the audio track of a video. Improving cross-model transferability and developing joint audio-visual protection are promising directions for future work.

Broader impacts. Personal videos posted online can be collected without the subject's authorization and used to fine-tune diffusion models or to condition image-to-video pipelines, producing customized content the subject never agreed to. `TC-UAP` gives individual users a practical tool to limit this: applying an imperceptible perturbation to a video before release leaves it visually unchanged for human viewers but renders it ineffective as training material for tuning-based customization or as a reference for I2V generation. The default thus shifts from "any uploaded video can be reused for identity-preserving generation" to "the uploader decides whether their identity can be replicated downstream." Content creators, journalists, and ordinary users can then share footage on public platforms while retaining control over how their face and likeness are reused.

At the policy level, opt-out clauses in current data-use regulations depend on data collectors voluntarily honoring them and are easily bypassed by automated scraping. `TC-UAP` complements such policies with a technical enforcement layer that does not require collector cooperation: even when a protected video is scraped, it provides no usable signal for downstream customization. This raises the cost of unauthorized identity replication and reduces the effective supply of clean training material for non-consensual deepfakes. We note two limitations of this perspective. First, protection applies only to videos released after the tool is adopted; previously released material cannot be retroactively shielded. Second, as with any defensive technique, `TC-UAP` may be evaded by future attackers with stronger temporal models or by collectors who obtain data from offline sources, so it should be viewed as one component of a broader set of privacy measures rather than a complete solution.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Sections 1, 4 and 5.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Appendix H.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

Justification: The paper does not present any formal theoretical results that require assumptions and proofs. The equations are used to define the threat model and protection objective. The main claims are validated through experiments.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Section 4 and Appendix C.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: Code, data access information, and instructions for reproducing the main results are available on our project page.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 5 and Appendix C.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: We follow the common evaluation practice in this field.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Appendix C.4 and Table 19.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that our research complies with it.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Appendix H.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper does not release high-risk assets.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Dataset sources and licenses are summarized in Table 5. Data usage and preparation are described in Appendix C.1.

    Guidelines:

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

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: Our code is available on the project page with documentation and a license.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

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

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.