

# VideoPure: Diffusion-based Adversarial Purification for Video Recognition

Kaixun Jiang, Zhaoyu Chen, Jiyuan Fu, Lingyi Hong, Jinglun Li, and Wenqiang Zhang

**Abstract**—Recent work indicates that video recognition models are vulnerable to adversarial examples, posing a serious security risk to downstream applications. However, current research has primarily focused on adversarial attacks, with limited work exploring defense mechanisms. Furthermore, due to the spatial-temporal complexity of videos, existing defense methods face issues of high cost, overfitting, and limited defense performance. Recently, diffusion-based adversarial purification leverages the randomness and deep gradient chains introduced by diffusion models during the purification process, achieving robust defense performance in the image domain. However, due to the additional temporal dimension in videos, directly applying these diffusion-based adversarial purification methods to the video domain suffers significant performance and efficiency degradation. To achieve an efficient and effective video adversarial defense method, we propose the first diffusion-based video purification framework to improve video recognition models’ adversarial robustness: VideoPure. Specifically, our framework is based on video diffusion models with DDIM sampling, employing proposed temporal DDIM inversion for the diffusing process. During DDIM denoising, we leverage intermediate results at each denoising step and conduct guided spatial-temporal optimization, removing adversarial noise while maintaining temporal consistency. Finally, we input the list of optimized intermediate results into the video recognition model for multi-step voting to obtain the predicted class. We investigate the defense performance of our method against SOTA black-box attacks, gray-box attacks, and adaptive attacks on benchmark datasets and models. Compared with other adversarial purification methods, our method demonstrates comprehensive better defense performance against various attacks. Moreover, our method can operate independently of video recognition models and enable flexible plug-in downstream tasks.

**Index Terms**—Video recognition model, adversarial defense, diffusion-based adversarial purification, VideoPure.

## I. INTRODUCTION

DEEP Neural Networks (DNNs) have exhibited remarkable performance in various computer vision tasks [1], [2], [3]. However, recent studies [4], [5], [6], [7], [8] have exposed a fundamental vulnerability inherent in DNNs’ capabilities, especially when facing adversarial examples. Adversarial examples introduce malicious and imperceptible perturbations to clean data, leading DNNs to make erroneous predictions with high confidence. Moreover, recent research has demonstrated the effectiveness of adversarial examples

Manuscript created October, 2020; This work was developed by the IEEE Publication Technology Department. This work is distributed under the L<sup>A</sup>T<sub>E</sub>X Project Public License (LPPL) (<http://www.latex-project.org/>) version 1.3. A copy of the LPPL, version 1.3, is included in the base L<sup>A</sup>T<sub>E</sub>X documentation of all distributions of L<sup>A</sup>T<sub>E</sub>X released 2003/12/01 or later. The opinions expressed here are entirely that of the author. No warranty is expressed or implied. User assumes all risk.

against video recognition models [9]. Actually, compared with images, videos as a higher-dimensional data modality, expose a larger attack surface, making video recognition models more susceptible to adversarial attacks. This vulnerability poses serious security risks to related downstream tasks such as autonomous driving [10], video surveillance [11], etc. Therefore, it becomes urgent to study video adversarial defense methods to enhance the security of video recognition models.

Unfortunately, current adversarial defense methods primarily focus on the image domain, while the video domain has received less attention. There are two main categories of adversarial defense methods. The first is adversarial training [4], [12], [13], [14], [15], where adversarial examples are used as a form of data augmentation to retrain the model so that it learns more robust distribution. However, adversarial training has several drawbacks: 1) Impact and inflexibility. It necessitates modifying the model’s intrinsic parameters, impacting the model’s accuracy with clean data and compromising its flexibility. 2) Overfitting. Adversarial training demonstrates poor defense performance against unseen adversarial examples (not including training data). 3) High cost. Adversarial training incurs higher training costs compared to standard training, as it requires continuous generation of adversarial examples. This issue is even more pronounced for video recognition models due to their more complex spatial-temporal architectures. Consequently, research on adversarial training for video recognition models remains limited [16], [17].

The second is adversarial purification [18], [19], [20], [21], [22], [23], [24], which treats adversarial perturbations as a form of noise and employs a pre-processing step to remove such noise before model inference. Unlike adversarial training, which requires modifying the model, adversarial purification only involves pre-processing the input during testing. Thus, minimal changes to the AI system can gain significant improvement in adversarial robustness. In the video domain, [22], [20], [23] have demonstrated the effectiveness of adversarial purification methods for defending video recognition models. However, when facing powerful adaptive attacks [25], [26], [27], these methods’ defense performance decreases dramatically. Recently, diffusion-based adversarial purification methods [24], [28], [29], [30], [31] have exhibited robust defense performance. These methods transform the adversarial image distribution into a simple Gaussian distribution through a diffusion process, and then remove the adversarial noise while retaining the clean image distribution through a denoising model reconstruction process. As a result, these methods showcase strong defense performance on image classifiers. Hence, a question arises: Is it possible to transpose the diffusion-based

purification defense onto video recognition models?

A straightforward approach is to treat the video as a sequence of images, applying an image diffusion model to denoise each frame individually. The resulting denoised video is then fed into video recognition models. However, this method disregards the temporal correlation between frames, leading to sub-optimal defense performance (as shown in Table V). Additionally, the iterative frame-by-frame processing, combined with DDPM sampling [32] which has been commonly utilized in previous adversarial purification methods, significantly influences the efficiency.

To tackle the challenges mentioned above and develop an effective and efficient method for defending against video adversarial attacks, we propose VideoPure, the first diffusion-based adversarial purification framework designed specifically for video recognition models. Our framework comprises three modules: temporal DDIM inversion, spatial-temporal optimization, and multi-step voting. Firstly, we improve efficiency by utilizing video diffusion models [33] with efficient DDIM sampling [34] as the baseline, replacing traditional image diffusion models with DDPM sampling. Next, since recovering video structures is more challenging than images, we introduce temporal DDIM inversion for the diffusion process. This method propagates noise predicted from the initial frame temporally, replacing stochastic diffusion [32], thereby transforming the initial distribution into a temporally consistent and trajectory-defined distribution. For the denoising process, we utilize intermediate results generated at each denoising timestep and perform spatial-temporal optimization on each intermediate result, preserving the video's original structure in the temporal domain while removing adversarial noise in the spatial domain. Finally, we input the list of optimized intermediate results into the video recognition model for multi-step voting to obtain the predicted class, thereby trapping the attack optimization into a local optimum for a certain purification results of the voting list. Our method not only inherits the advantages of adversarial purification methods, operating independently of the video recognition model and allowing flexible application in downstream tasks, but it also achieves robust defense performance against multiple attacks. Our contributions can be summarized as follows:

- To the best of our study, we are the first to introduce diffusion based adversarial purification in the video domain and propose an effective and efficient framework VideoPure.
- To enhance defense efficiency, we utilize the video diffusion model as the purification model and employ highly efficient DDIM sampling, replacing traditional image diffusion models and DDPM sampling.
- To improve defense performance, we propose temporal DDIM inversion for diffusion and spatial-temporal optimization for denoising, preserving the video's original structure while removing adversarial noise. Then, we predicted class through multi-step voting to trap the attack optimization in a local optimum.
- VideoPure achieves excellent defense performance under benchmark datasets and models against multiple attack methods (black-box, gray-box, adaptive) compared with

the state-of-the-art defense methods.

- Our method requires no training or prior information of protected models. Therefore, it can be applied as a flexible defense plugin for any video recognition models.

The remainder of the paper is organized as follows: Section II briefly reviews the related work to adversarial attacks and adversarial defenses. Section III first introduces the definition of diffusion-based video adversarial defense, and then details the proposed VideoPure framework. Section IV shows the experimental results to demonstrate the effectiveness of our method, including performance comparison, ablation studies, diagnostic experiments, effectiveness analysis, etc. Section V concludes our work.

## II. RELATED WORK

### A. Adversarial Attacks

Goodfellow et al. [35] first discover that introducing imperceptible perturbations to the original inputs can lead deep learning models to make incorrect predictions, such inputs are termed adversarial examples. Depending on the extent of information available to the attacker regarding the attacked model and defense method, adversarial attacks can be categorized as black-box attacks, gray-box attacks, and adaptive attacks.

1) **Black-box Attacks:** Attackers can not access information about the attacked model or defense mechanisms. They generally need to employ surrogate models for designing adversarial examples to attack the target model (attacked model). On video recognition, TT [36] utilizes video recognition models as surrogate models, aggregating gradients from different video input patterns to achieve effective attacks. GIE [37], I2V [38] employ image classification models as surrogate models, also achieving effective black-box attacks on video recognition models. Additionally, some video attack methods [39], [40], [41], [42] explore a soft black-box setting, wherein the attacker can access the model's output and optimize the attack based on it. However, as security awareness increases, it is becoming increasingly difficult for attackers to access the model's output multiple times [41]. Hence, in this study, we primarily focus on the primary black-box setting, i.e., no information can be accessed from attacked models, selecting TT as the black-box attack to evaluate the defense method's performance. Black-box attacks are the most difficult attack setting for attackers, leading to the generation of adversarial examples that are generally lower in performance than gray-box attacks and adaptive attacks. Therefore it is often not possible to fully assess the performance of defense methods.

2) **Gray-box Attacks:** Attackers can access all information about the attacked model but lack information about defense mechanisms. In the absence of adversarial defense considerations, gray-box attacks can be regarded as white-box attacks. Among them, gradient-based optimization methods, including FGSM [35], PGD [4], and I-FGSM [43], have achieved strong attack performance by computing the gradients of the model with respect to the input to design adversarial examples. Taking PGD as an example, which is often used to evaluate

the performance of adversarial defense, the generation process of adversarial examples can be represented as [4]:

$$x_{adv}^{(i+1)} = \Pi_{x+\epsilon_{adv}} \left( x_{adv}^{(i)} + a \operatorname{sgn} \left( \nabla_{x_{adv}^{(i)}} L(f_\theta(x_{adv}^{(i)}), y) \right) \right), \quad (1)$$

where  $x_{adv}^{(i)}$  denotes the adversarial example generated in the  $i^{th}$  iteration,  $\Pi_{x+\epsilon_{adv}}$  ensures that  $x_{adv}$  remains within the  $\epsilon$ -ball centered at the original input  $x$ ,  $a$  denotes the step size,  $f_\theta(\cdot)$  denotes the attacked model,  $\operatorname{sgn}(\cdot)$  denotes the sign function and  $\nabla_{x_{adv}} L(f_\theta(x_{adv}), y)$  denotes the gradient of loss with respect to  $x_{adv}$ .

On video recognition, [9] crafts adversarial examples for video recognition models through inter-frame perturbations and optimization with the Adam optimizer [44]. Leveraging the structural vulnerabilities inherent in video models, [45] effectively attacks by perturbing a single frame. Additionally, [46] introduces uniform RGB perturbations to each frame, mimicking variations in real-world lighting conditions. TT [36] also can be seen as a gray-box attack when the surrogate model becomes the attacked model. Consequently, gray-box attacks typically exhibit potent attack performance in the absence of defense methods, but their effectiveness significantly diminishes with the introduction of defense mechanisms. In this study, we employ PGD and TT as gray-box attacks to assess the performance of defense methods.

3) **Adaptive Attacks:** Attackers can access all information about the attacked model and defense methods. Therefore, attackers can treat the target model  $f_\theta(\cdot)$  and the defense method  $P(\cdot)$  as a new model  $F_\theta(\cdot) = f_\theta(P(\cdot))$ . Then, gray-box attacks such as PGD, can be utilized for adaptive attacks. For example, the adaptive attack form of PGD can be expressed as:

$$x_{adv}^{(i+1)} = \Pi_{x+\epsilon_{adv}} \left( x_{adv}^{(i)} + a \operatorname{sgn} \left( \nabla_{x_{adv}^{(i)}} L(F_\theta(x_{adv}^{(i)}), y) \right) \right), \quad (2)$$

i.e., the gradient with respect to the attacked model  $\nabla_{x_{adv}} L(f_\theta(x_{adv}), y)$  are transformed into the gradient with respect to the new model  $\nabla_{x_{adv}} L(F_\theta(x_{adv}), y)$ . However, some defense methods introduce non-differentiable operations (e.g. JPEG [21]) or cause memory overflow during gradient computation (e.g. diffusion-based adversarial purification [24]), preventing attackers from obtaining gradients. To address this problem, Backward Pass Differentiable Approximation (BPDA) [25] replaces  $P(\cdot)$  with an appropriate differentiable surrogate function  $g(\cdot)$ . For example, in adversarial purification, since the difference between the model's input and output mainly lies in the adversarial noise, the overall structure of the original input and the purified input remains largely consistent. Therefore,  $g(\cdot)$  can be represented as  $P(x) \approx g(x) = x$ . The gradients  $\nabla_{x_{adv}} L(F_\theta(x_{adv}), y)$  can then be transformed to  $\nabla_{P(x_{adv})} L(f_\theta(P(x_{adv})), y)$ .

Diffattack [47] is a recently proposed adaptive attack method specifically designed for diffusion-based adversarial purification. To address memory issues, it employs a segment-wise gradient estimation method, reducing memory consumption to  $O(1)$  cost. In order to mitigate the gradient vanishing or explosion problem caused by deep gradient chains, a residual-based reconstruction loss is proposed. Although Diffattack

estimates more accurate gradients, the segmented gradient computation incurs additional time costs.

Many defense methods (e.g. diffusion-based) improve the effectiveness of the defense by introducing randomness to bias the gradient estimated by the attacker when designing the attack. To deal with this situation, Expectation-Over-Transformation (EOT) [26] is proposed, which greatly improves the effectiveness of attacks against stochastic defenses by transforming the expected gradient, i.e.,  $\nabla_{x_{adv}} \mathbb{E}[L(F_\theta(x_{adv}), y)] = \mathbb{E}[\nabla_{x_{adv}} L(F_\theta(x_{adv}), y)]$ .

In summary, adaptive attacks represent the most challenging scenario for defense, thus, superior adaptive defense performance better demonstrates the efficacy of defense methods. In this study, we select PGD+BPDA, TT+BPDA, EOT+BPDA, and PGD+Diffattack to evaluate the performance of different defense methods against adaptive attacks from multiple perspectives.

## B. Adversarial Defense

1) **Adversarial Training:** Adversarial training involves using adversarial examples as a form of data augmentation to retrain the model. This process helps the model learn a more robust decision boundary. Goodfellow et al. [48] initially introduce adversarial examples during training by generating them using FGSM at each training step and injecting them into the training set. Subsequently, PGD gradually becomes the mainstream method for adversarial training [4]. On video recognition, MultiBN [16] explores the impact of different forms of adversarial training on defense performance by employing multiple batch normalization layers. [17] explores the impact of different forms of adversarial training on the defense performance of video models. Adversarial training offers the advantage of significantly improving the model's defense capability against known adversarial examples. However, it remains vulnerable to unseen types of adversarial examples. Additionally, adversarial training on video-based models poses more challenges due to their high computational complexity [20].

2) **Adversarial Purification:** Adversarial purification involves pre-processing input data before feeding it into a model to eliminate adversarial noise. This pre-processing may include specific denoising techniques like JPEG [21] or Super Resolution (SR) [49], as well as generative denoising models such as Defense-GAN [18] and autoregressive generative models [19]. In the domain of videos, [22] proposes an adaptive JPEG compression method based on optical flow. Additionally, [20] introduces designed defense patterns into input videos to enhance adversarial defense performance. Temporal Shuffle [23] defends against adversarial attacks by randomly shuffling the order of video frames, achieving notable defense performance. Recently, diffusion-based adversarial purification methods [24], [28], [29], [30], [31] have leveraged the potent generation and denoising capabilities of diffusion models, resulting in state-of-the-art robustness in image classification. Among them, Diffpure introduces random noise and deep gradient chains during the diffusion denoising process, rendering it resilient against advanced adversarial attacks. However, the

high-dimensional attributes of video, including the additional temporal dimension, significantly increase both the difficulty and cost of defense. This results in performance and efficiency degradation when existing diffusion-based adversarial purification methods are applied to the video domain. We are the first to explore diffusion-based adversarial purification for video models and propose an effective and efficient diffusion-based framework for video adversarial purification.

### III. METHOD

#### A. Preliminary

Given a clean video  $x \in \mathbb{R}^{N \times H \times W \times C}$  and video recognition model  $f_\theta(\cdot)$ , where  $N, H, W, C$  denote the number of frames, height, width, channel, respectively. We denote  $\arg \max f_\theta(x) = y$  as the process of correct classification, where  $y$  denotes the ground truth label of  $x$ .

We denote the adversarial video  $x_{adv}$ , on the one hand,  $x_{adv}$  needs to deceive the video recognition model, i.e.,  $\arg \max f_\theta(x_{adv}) \neq y$ , typically constrained by maximizing the loss. On the other hand,  $x_{adv}$  also needs to be similar to  $x$ , typically constrained by  $l_p$  distance, i.e.,  $\|x_{adv} - x\|_p$ .

**Diffusion-based Adversarial Purification.** We represent the process of diffusion-based adversarial purification as  $P(\cdot)$ . Therefore, combining  $P(\cdot)$ , the defended video recognition model can be written as  $f_\theta(P(\cdot))$ . The optimization objective for defense can be expressed as:

$$\max_{P(\cdot)} \sum_i \arg \max f_\theta(P(x^i)) = y^i, \quad s.t. \quad x^i \in \{x^i, x_{adv}^i\}, \quad (3)$$

where  $(x^i, y^i)$  denotes  $i^{th}$  test sample.  $x_{adv}^i$  denotes the adversarial video generated by  $x^i$ . Therefore, the designed defense method  $P(\cdot)$  should aim to minimally impact the recognition accuracy of clean data while enhancing the recognition accuracy of adversarial examples.

In this study, we employ a popular video-based latent diffusion model: ModelScope [33] as the denoising model. Initially, ModelScope encodes the input video into a latent space, subsequently performs diffusion and denoising in the latent space, and finally decodes latent from latent space to pixel space. As a result, the process  $P(\cdot)$  can be decomposed into four components: encoding  $\mathcal{E}(\cdot)$ , diffusion  $q(\cdot)$ , denoising  $\epsilon_\theta(\cdot)$ , and decoding  $\mathcal{D}(\cdot)$ .

**Encoding & Decoding.** The latent space in this paper is proposed by VQGAN [50], and the process of encoding and decoding can be represented as:

$$z_0 = \mathcal{E}(x), \quad x^* = \mathcal{D}(z_0), \quad (4)$$

where  $z_0 \in \mathbb{R}^{N \times \frac{H}{8} \times \frac{W}{8} \times 4}$  denotes the representation of the video in the latent space,  $x^*$  denotes the purified  $x$ . Then, we remove the structure of  $z_0$  by forward diffusion process  $q(\cdot)$ .

**Diffusion.** Forward diffusion process can be seen as a Markov chain, commencing from the initial state  $z_0$ , gradually introducing small noise onto the  $z_0$  to transform the complex input distribution into a simpler, well-defined distribution, such

as the Gaussian distribution. This process can be formally defined as [32]:

$$\begin{aligned} q(z_{1:t^*} | z_0) &:= \prod_{t=1}^{t^*} q(z_t | z_{t-1}), \\ q(z_t | z_{t-1}) &:= \mathcal{N}\left(z_t; \sqrt{1-\beta_t} z_{t-1}, \beta_t I\right), \end{aligned} \quad (5)$$

where  $t^*$  denotes the total timesteps of diffusion,  $\beta_t$  denotes forward process variance of  $t^{th}$  timestep, which is generally a small constant that increases linearly [32]. The above recursive equation can be simplified as follows [32]:

$$\begin{aligned} q(z_{t^*} | z_0) &= \mathcal{N}(z_{t^*}; \sqrt{\bar{\alpha}_{t^*}} z_0, (1 - \bar{\alpha}_{t^*}) I), \\ z_{t^*} &= \sqrt{\bar{\alpha}_{t^*}} \cdot z_0 + \sqrt{1 - \bar{\alpha}_{t^*}} \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \end{aligned} \quad (6)$$

where  $\alpha_{t^*} = 1 - \beta_{t^*}$  and  $\bar{\alpha}_{t^*} = \prod_{t=1}^{t^*} \alpha_t$ . Therefore, we only need  $z_0$  and the timestep  $t^*$  to obtain the corresponding  $z_{t^*}$ . From the diffusion process of Eq. 6, we can draw the following conclusions: 1) Since  $\epsilon$  is sampled from a Gaussian distribution, the generation of  $z_t$  exists randomness. 2) The larger  $t^*$  is, the smaller  $\sqrt{\bar{\alpha}_{t^*}}$  becomes, making  $z_{t^*}$  closer to a Gaussian distribution. This implies that as adversarial noise is increasingly smoothed out, the disruption to the original structural information also increases.

**Denoising.** Reverse denoising process aims to remove the noise starting from  $z_{t^*}$  to recover  $z_0^*$ . This process can be illustrated as follows:

$$p_\theta(z_{t-1}^* | z_t^*) = \mathcal{N}(z_{t-1}^* | \mu_\theta(z_t^*, t), \sigma_t^2 I), \quad (7)$$

where  $z_t^*$  denotes the denoised latent at timestep  $t$ , among them,  $z_{t^*}^* = z_{t^*}$ .  $\mu_\theta(z_t^*, t)$  denotes the predicted mean, generated by diffusion model with  $z_t^*$  and predicted noise  $\epsilon_\theta(z_t^*, t) \in \mathbb{R}^{N \times \frac{H}{8} \times \frac{W}{8} \times 4}$ .  $\sigma_t^2 I$  denotes the  $t^{th}$  timestep's variance. In general,  $\sigma_t^2$  is often defined as a hyper-parameter [32]. Taking DDPM sampling as an example,  $\mu_\theta(z_t^*, t) = \frac{1}{\sqrt{\alpha_t}} \left( z_t^* - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(z_t^*, t) \right)$ . Since DDPM is predicated on a Markov chain process, it necessitates a step-by-step denoising operation, progressing from  $z_{t^*}$  to reconstruct  $z_0^*$ . To enhance denoising efficiency, DDIM [34] is widely used as an efficient skip-step sampler. The formal representation of DDIM is illustrated as follows [34]:

$$\begin{aligned} z_{t-1}^* &= \sqrt{\bar{\alpha}_{t-1}} \left( \frac{z_t^* - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_t^*, t)}{\sqrt{\alpha_t}} \right) \\ &\quad + \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_\theta(z_t^*, t), \end{aligned} \quad (8)$$

DDIM is a non-Markov chain sampling technique that allows for skip-step sampling implementation. This means that it is not required to specify all  $z_{1:t^*}^*$  step by step; instead, we can perform skip-step sampling within subsets of the process.

Previous research [24] confirms that the distribution of adversarial examples and clean data will converge to a similar distribution after a certain timestep  $t^*$  of diffusion. As the diffusion model is inherently trained on clean data, the denoised  $z_0^*$  produced by DDPM or DDIM preserves the structure of  $z_0$  while effectively removing the adversarial noise. Consequently, this process achieves a desirable defense effect. Due to the complexity of reconstructing video structures compared to images, we propose temporal DDIM inversion to

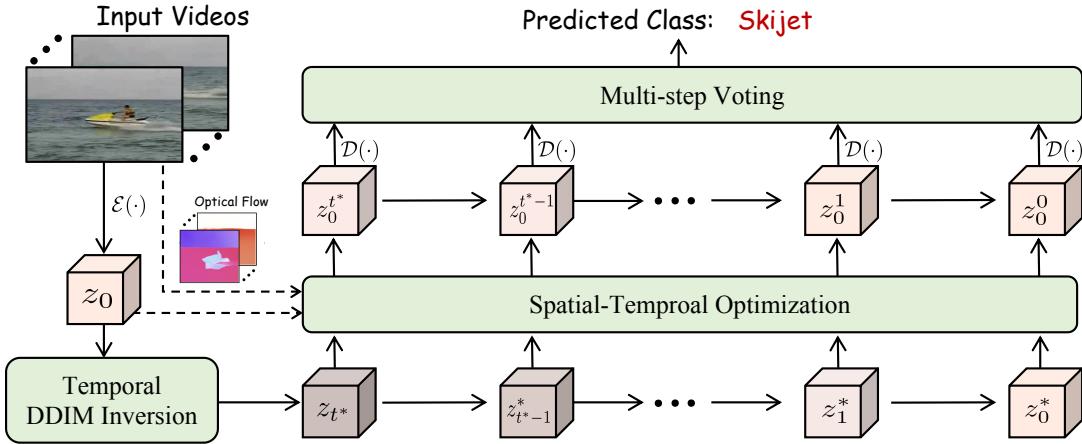


Fig. 1. Overview of our VideoPure framework. First, we encode video input to the latent space, getting  $z_0$ . And then we adopt temporal DDIM inversion to diffuse  $z_0$  to  $z_t$ . For the denoising process, we fully utilize the intermediate results obtained from DDIM sampling, optimizing each  $z_t^*$  in both spatial and temporal domain to derive the corresponding  $z_0^t$ . Finally, we input all optimized  $z_0^t$  into the video recognition model after decoding, and aggregate the predictions through voting to obtain the final prediction category.

transform the initial distribution of clean videos or adversarial videos into a temporally consistent and trajectory-defined simple distribution, helping reconstruct more original structural information during the denoising process. Additionally, we propose a strategy that combines spatial-temporal optimization and multi-step voting during denoising. By optimizing multiple reconstructed video distributions in the spatial-temporal domain, confusing the attacker's optimization direction while robustly removing adversarial noise.

### B. VideoPure Framework

**Overview.** Fig. 1 provides an overview of our VideoPure framework, comprising three modules: temporal DDIM inversion, spatial-temporal optimization, and multi-step voting. Initially, we transform the video  $x$  to  $z_0$  from the pixel space to the latent space, utilizing temporal DDIM inversion for diffusion to obtain  $z_{t^*}$ . Subsequently, we employ DDIM sampling for denoising. During the denoising process, we fully exploit the intermediate results obtained at each denoising timestep, and optimize them by spatial-temporal optimization. Specifically, we disrupt the structural adversarial noise in the spatial domain while preserving the consistency of the video structure in the temporal domain. Finally, we aggregate all optimized intermediate results to achieve multi-step voting. Specifically, after decoding them back to the pixel space, we input them into the video recognition model. The prediction category with the highest number of votes is selected as the final prediction result.

1) **Temporal DDIM Inversion:** In the domain of real image editing [51], [52], to preserve the structure of the original input, DDIM inversion [34] is widely adopted to transform the initial latent representation  $z_0$  into a corresponding  $z_{t^*}$  through a specific inversion scheme, rather than relying on the stochastic diffusion process described in Eq. 6. Since the trajectory of DDIM inversion is deterministic,  $z_{t^*}$  encoded via DDIM inversion can effectively reconstruct the details and structure of the initial latent representation  $z_0$  after DDIM sampling. Therefore, to ensure maximal retention of recognition accuracy

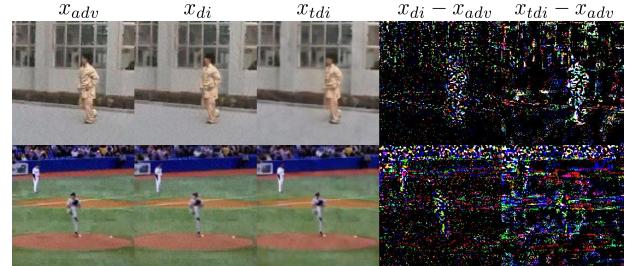


Fig. 2. Visualizations after different inversion methods.  $x_{adv}$ ,  $x_{di}$ ,  $x_{tdi}$  denote adversarial examples generated by PGD, DDIM inversion, temporal DDIM inversion. Comparing the second and third columns, our temporal DDIM inversion removes adversarial noise compared to DDIM inversion. The fourth and fifth columns represent our temporal DDIM inversion removes more adversarial noise, showing difference with  $x_{adv}$ .

for clean data, we opt for DDIM inversion for latent encoding. Specifically, DDIM inversion first transforms Eq. 8 as follows:

$$z_t = \sqrt{\bar{\alpha}_t} \left( \frac{z_{t-1} - \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_\theta(z_t, t)}{\sqrt{\bar{\alpha}_{t-1}}} \right) + \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_t, t). \quad (9)$$

Eq. 9 describes the process of generating  $z_t$  from the encoding of  $z_{t-1}$ , which is the reverse process of denoising. However, every stage of encoding relies on the predicted noise of the next step  $\epsilon_\theta(z_t, t)$ . As  $z_t$  is not available, the predicted noise cannot be obtained. Hence,  $\epsilon_\theta(z_{t-1}, t)$  is adopted as a substitute, under the assumption that  $z_{t-1}$  is approximately equal to  $z_t$  [53]. Through this alteration, the formal expression of the DDIM inversion is transformed as follows:

$$z_t = \sqrt{\bar{\alpha}_t} \left( \frac{z_{t-1} - \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_\theta(z_{t-1}, t)}{\sqrt{\bar{\alpha}_{t-1}}} \right) + \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_{t-1}, t). \quad (10)$$

However, while DDIM inversion guarantees the recognition accuracy of clean data, it restores too much structural information of  $z_0$ . As can be observed from Fig. 2, the sole use of DDIM inversion leads to the restoration of the structure of

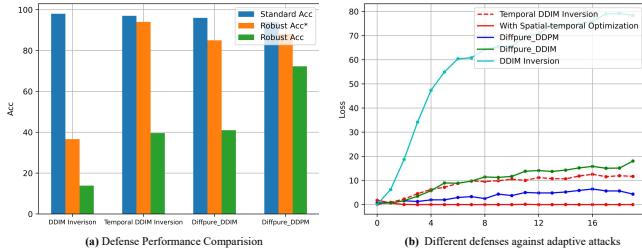


Fig. 3. (a) shows the performance comparison of different defense methods. (b) shows the loss of different defense methods in facing adaptive attacks, and the greater the rise of the loss, the worse the performance of the defense. PGD for evaluating Robust Acc\* and PGD+BPDA for evaluating Robust Acc.

the adversarial noise, thus failing to achieve effective defense against adversarial examples. Based on this, we optimize DDIM inversion to suit the video adversarial purification. Specifically, motivated by the contributions of temporal noise initialization to video generation [54], we replace the noise added to each frame at each step with the first frame of predicted noise. That is, for the diffusion at timestep  $t$ , Eq. 10 becomes:

$$z_t = \sqrt{\bar{\alpha}_t} \left( \frac{z_{t-1} - \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_\theta^*(z_{t-1}, t)}{\sqrt{\bar{\alpha}_{t-1}}} \right) + \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta^*(z_{t-1}, t), \quad (11)$$

where  $\epsilon_\theta^*(z_{t-1}, t)$  denotes the noise replicated from the predicted first frame and applied  $N$  times to each subsequent frame. After this transformation, we can transform the initial distribution of clean videos or adversarial videos into a temporally consistent and trajectory-defined simple distribution. The advantage of temporal DDIM inversion lies in two main aspects. Firstly, compared to the stochastic diffusion in Eq. 6, the diffusion trajectory of temporal DDIM inversion is more deterministic, which allows it to reconstruct more of the video's original structure after DDIM denoising. Secondly, unlike standard DDIM inversion, our method introduces a temporal difference between the diffusion trajectory and DDIM denoising trajectory. This temporal discrepancy enables the denoised video to retain substantial structural information while discarding finer details, such as adversarial noise. As illustrated in Fig. 2.

2) **Spatial-temporal Optimization:** As shown in Fig. 3(a), temporal DDIM inversion greatly enhances the model's defense performance against gray-box attacks, i.e., high Robust Acc\*. However, under the adaptive attack setting, where the attacker can access the details of defense methods, there is still poor defense performance compared with Diffpure. In the adaptive attack setting, attackers treat the defense model and video recognition model as a new model. For instance, in the case of PGD attacks, they can estimate the gradient of the combined model to implement an effective attack. Fig. 3(b) shows that temporal DDIM inversion under this attack setting indicates a linear ascent in loss with the number of iterations, failing to provide any defense effect. Therefore, considering the characteristics of videos and the optimization goal of the defense, we propose spatial-temporal optimization, aiming to optimize the denoised latent through the introduction of

complex spatial-temporal noise patterns, making it difficult for attackers to find adversarial examples that consistently exceed the probability decision boundary.

In adversarial defense, there are two optimization objectives: firstly, when confronted with adversarial examples, the aim is to eliminate adversarial noise; secondly, when handling clean examples, it is essential to maintain structural similarity. Hence, we address these two optimization objectives separately in the spatial and temporal domains.

In the spatial domain, for each frame, the adversarial perturbation is attached to each pixel, thereby influencing the judgment of the classifier. As such, we aim to purify adversarial perturbations by distancing the denoising-generated  $z_0^*$  from the original input-encoded  $z_0$ . Therefore, we employ the Mean Squared Error (MSE) loss as a constraint, which is explicitly represented as follows:

$$L_{spa} = -\|z_0 - z_0^*\|_2, \quad (12)$$

Through the optimization, the noise of each frame of the video is optimized to a variable degree, which increases the difficulty of the attack optimization. However, it's apparent that the spatial loss, not only disrupts adversarial noise, but also undermines the structure of original videos. Therefore, from the temporal domain, we ensure content consistency among adjacent frames, which results in maintaining the consistency of the temporal structure between the denoising video and the original video. Motivated by [55], we leverage temporal dynamics in videos to optimize the denoising latent. Specifically, we constrain the process by calculating the  $L_1$  reconstruction loss between the flow-based warping of  $z_0^*$  and  $z_0^*$  itself, which can be represented as follows:

$$L_{temp} = \sum_{i=1}^{N-1} \|Warp(z_0^*(i+1), O(i)) - z_0^*(i)\|_1, \quad (13)$$

where  $Warp(z_0^*(i+1), O(i))$  denotes the warping frame of  $i^{th}$  frame, i.e., backward warping, generated based on  $i+1^{th}$  frame and the optical flow between  $i^{th}$  and  $i+1^{th}$  frame.

Combining  $L_{temp}$ , our loss can be described as  $L = \lambda_1 L_{temp} + \lambda_2 L_{spa}$ . Finally, the process of spatial-temporal optimization can be represented as:

$$z_0^* = z_0^* - \alpha_s \sigma^2 \nabla_{z_0^*} L. \quad (14)$$

where  $\alpha_s$  is the optimization size,  $\sigma^2$  denotes the variance of the DDIM schedule [55]. Hence, when confronted with adversarial attacks, particularly adaptive attacks, each attack optimization iteration introduces an complex spatial-temporal noise. This makes it challenging for the attacker to accurately estimate the gradient and consequently difficult to generate adversarial examples that consistently surpass the probability decision boundary, as shown in Fig. 3 (b).

3) **Multi-step Voting:** Diffusion-based adversarial purification leverages the extremely deep gradient chain during inference to trap attackers in gradient explosion, vanishing, or memory explosion, thereby achieving effective defense [47]. Inspired by this, we propose to design more complex gradient chains to further increase the difficulty of attack optimization. Specifically, we propose to broaden the gradient chain

**Algorithm 1:** VideoPure

---

**Input:** input video  $x$ , encoder  $\mathcal{E}$ , decoder  $\mathcal{D}$ , video recognition model  $f_\theta$ , denoising model  $\epsilon_\theta$ , diffusion timestep  $t^*$

**Output:** predicted classification

- 1:  $vote\_list = \{\}$ .
- 2:  $z_0 = \mathcal{E}(x)$ .
- 3: Generate  $z_{t^*}$  using temporal DDIM inversion.
- 4: **for**  $t$  in  $[t^*, 1]$  **do**
- 5:    $z_0^t = \frac{z_t^* - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_t^*, t)}{\sqrt{\bar{\alpha}_t}}$
- 6:    $z_{t-1}^* = \sqrt{\bar{\alpha}_{t-1}} z_0^t + \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_\theta(z_t^*, t)$ .
- 7:   Calculate  $L_{temp}$  and  $L_{spa}$  with  $z_0^t, z_0$ .
- 8:    $z'_t = z_t^* - \alpha_s \sigma_t^2 \nabla_{z_t^*} L$ .
- 9:    $z_0^t = \frac{z'_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_t^*, t)}{\sqrt{\bar{\alpha}_t}}$ .
- 10:    $vote\_list = vote\_list + \{z_0^t\}$ .
- 11: **end for**
- 12: Calculate  $L_{temp}$  and  $L_{spa}$  with  $z_0^*, z_0$ .
- 13:  $z_0^0 = z_0^* - \alpha_s \sigma_t^2 \nabla_{z_0^*} L$ .
- 14:  $vote\_list = vote\_list + \{z_0^0\}$ .
- 15:  $y_{pred} = []$
- 16: **for**  $z_0^t$  in  $vote\_list$  **do**
- 17:    $y_{pred} = y_{pred} + \arg \max f_\theta(\mathcal{D}(z_0^t))$ .
- 18: **end for**
- 19: **return**  $\max y_{pred}$

---

by utilizing the intermediate results of each denoising step and combine it with spatial-temporal optimization to further increase the difficulty of attack optimization, trapping attacks in a local optimum of a voting sample.

DDIM sampling in Eq. 8 can be divided into two steps. The first step predicts the process of  $z_0^t$  based on the current time step  $z_t^*$  and  $\epsilon_\theta(z_t^*, t)$ , and the second step obtains  $z_{t-1}^*$  by using Eq. 6. Therefore, we can gain a series of  $z_0^t$  during denoising process. Specifically, as  $\epsilon_\theta(z_t^*, t)$  can be approximated as Gaussian noise,  $\epsilon$  in Eq. 6 can be replaced by  $\epsilon_\theta(z_t^*, t)$ . Upon reshaping Eq. 6, we can calculate the predicted  $z_0^t$  based on timestep  $t$ :

$$z_0^t = \frac{z_t^* - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(z_t^*, t)}{\sqrt{\bar{\alpha}_t}}. \quad (15)$$

Therefore, in the denoising process, each timestep can yield its corresponding predicted  $z_0^t$ . To fully utilize this information, we carry out the above spatial-temporal optimization for each step  $z_0^t$  of the denoising process. Specifically,  $z_t$  first generates a corresponding  $z_0^t$  based on Eq. 15. Next, we calculate the  $L_{temp}$  and  $L_{spa}$ , and then optimize  $z_t$  for the improvement of Eq. 14:

$$z'_t = z_t^* - \alpha_s \sigma_t^2 \nabla_{z_t^*} L, \quad (16)$$

where  $\sigma_t^2$  denotes the variance in each step of the DDIM schedule. Then, we use  $z'_t$  to generate the optimized  $z_0^t$ . Finally, we obtain  $t^*+1$  optimized  $z_0^t$ . After decoding all, they are inputted into the video recognition model to yield  $t^*+1$  classification results. The category with the highest vote count is selected as the final classification result. Multi-step voting makes it possible for an attack optimization to get trapped in

a local optimum by trapping a candidate solution ( $z_0^t$ ) in the voting list, so multi-step voting can also improve the overall defense performance. Algorithm 1 describes the whole process of our VideoPure.

## IV. EXPERIMENTS

## A. Experimental Settings

**Datasets.** To evaluate the performance of our proposed framework, VideoPure, we employ two widely used video recognition datasets: UCF-101 [56] and Kinetics-400 [57]. The UCF-101 dataset comprises 13,320 video clips spanning 101 action categories. Following [36], [41], we adopt a random sampling approach, selecting one video from each category for evaluation. The Kinetics-400 dataset encompasses 400 action categories, with approximately 240,000 video clips for training and 20,000 for validation. Considering the computational cost of comparison methods (e.g., DiffPure), we randomly select 100 categories and one video from each category for evaluation. We ensure that all the selected videos can be correctly classified by all video recognition models.

**Models.** To thoroughly assess the performance of our proposed framework, we conduct comprehensive evaluations on two widely used pretrained video recognition models: Non-local Network (NL) [58] and SlowFast [59]. For each video recognition model, we employ ResNet-50 [1] as the backbone architecture. we standardize the input size for all models to  $N \times C \times H \times W = 32 \times 3 \times 224 \times 224$ .

**Metrics.** We evaluate the defense performance from three aspects: 1). Standard Acc(%) [24]: the recognition accuracy against clean data. 2). Robust Acc(%) [24]: the recognition accuracy against adaptive attacks. 3). Robust Acc\*(%): the recognition accuracy against gray-box attacks.

**Attack Methods.** We select strong baseline attack methods to evaluate our defense method. For gray-box attacks, we choose PGD [4] and TT [36] which have strong attack performance for image and video models, respectively. For adaptive attacks, we select PGD, TT, Expectation Over Transformation (EOT) [26], AutoAttack [27] as attack methods, and we combine them with BPDA [25], Diffattack [47] to calculate defense methods' gradient, achieving effective adaptive attacks. We also select TT to verify the defense performance against black-box attacks.

**Defense Methods.** We primarily select state-of-the-art (SOTA) adversarial purification methods in both the image and video domains for performance comparison. We initially opt for the classic JPEG [21] and SR [49] methods. Next, we select the efficient video defense method Temporal Shuffle [23]. We then choose diffusion-based adversarial purification method Diffpure [24], comparing both its DDPM (Diffpure\_DDPM) and DDIM (Diffpure\_DDIM) versions. Additionally, we consider DDIM inversion [34] (without the inclusion of any modules we propose) as the baseline for our comparison. To ensure a fair comparison and demonstrate the effectiveness of our proposed method, we utilize the same video diffusion model ModelScope [33] that our method adopts, as the denoising model for all compared approaches.

**Implement Details.** For  $l_\infty$  attacks, we set  $\epsilon_{adv} = 4/255$ , for  $l_2$  attacks, we set  $\epsilon_{adv} = 2$ . For most experiments

TABLE I

COMPARISON OF DEFENSE PERFORMANCE OF OUR METHOD TO OTHER SIX DEFENSE METHODS UNDER SIX ATTACK METHODS. AVERAGE DENOTES THE AVERAGE VALUE OF THE DEFENSE PERFORMANCE AGAINST THE SIX ATTACK METHODS.

Dataset	Model	Defense Method	Attack Method							
			None	PGD	PGD+BPDA	TT	TT+BPDA	EOT+BPDA	PGD+Diffattack	Average
UCF-101	NL	None	100	8.9	-	11.8	-	-	-	-
		JPEG	99	8.9	9.9	13.9	16.8	9.9	-	11.8
		WN	100	7.9	9.9	12.9	12.9	9.9	-	10.7
		Temporal Shuffle	95	84.1	12.87	12.9	9.9	5.9	-	25.1
		Diffpure_DDPM	94	88.2	72.3	72.3	62.4	39.6	65.4	66.7
		Diffpure_DDIM	96	85.1	41.6	54.5	54.5	13.9	22.8	45.4
		DDIM Inversion	98	36.6	13.9	21.8	18.8	12.9	22.8	21.1
		Ours	96	94.1	100	55.4	80.2	97.0	89.1	85.9
		None	100	7.9	-	5.0	-	-	-	-
Kinetics-400	SlowFast	JPEG	99	8.9	7.9	5.9	8.9	11.9	-	8.7
		WN	99	8.9	7.9	5.0	5.9	8.9	-	7.3
		Temporal Shuffle	93	74.2	5.0	8.9	5.0	5.9	-	19.8
		Diffpure_DDPM	94	88.1	80.2	79.2	74.3	52.5	77.2	75.3
		Diffpure_DDIM	93	91.1	62.4	68.3	61.3	19.8	39.6	57.1
		DDIM Inversion	97	33.7	6.9	17.8	5.9	7.9	13.9	14.4
		Ours	98	75.2	90.1	68.3	85.1	91.0	63.4	78.9
		None	100	5.0	-	5.0	-	-	-	-
		JPEG	97	7.0	6.0	6.0	7.0	6.0	-	6.4
Kinetics-400	NL	WN	99	7.0	5.0	6.0	6.0	4.0	-	5.6
		Temporal Shuffle	96	86.0	6.0	4.0	2.0	3.0	-	20.2
		Diffpure_DDPM	98	93.0	79.0	75.0	65.0	54.0	66.0	72.0
		Diffpure_DDIM	98	87.0	54.0	57.0	54.0	27.0	27.0	51.0
		DDIM Inversion	98	42.0	9.0	17.0	30.0	9.0	14.0	20.2
		Ours	91	90.0	98.0	61.0	73.0	98.0	81.0	83.5
		None	100	4.0	-	5.0	-	-	-	-
		JPEG	98	3.0	3.0	6.0	7.0	3.0	-	4.4
		WN	98	3.0	3.0	5.0	5.0	3.0	-	3.8
Kinetics-400	SlowFast	Temporal Shuffle	92	80.0	4.0	5.0	3.0	3.0	-	19.0
		Diffpure_DDPM	96	87.0	71.0	72.0	64.0	44.0	71.0	68.2
		Diffpure_DDIM	95	82.0	47.0	61.0	40.0	18.0	26.0	45.7
		DDIM Inversion	96	26.0	3.0	16.0	12.0	3.0	14.0	12.3
		Ours	93	82.0	91.0	66.0	82.0	93.0	71.0	80.8

we chose the most popular  $l_\infty$  attacks to evaluate defense performance. We set attack iteration is 10 for PGD and TT. We set  $eot\_attack\_reps = 20$  for EOT. For AutoAttack, we follow the default *random* setting. For the extra hyper-parameters in our method, we set timestep  $t^* = 6$  for DDIM ( $T = 50$ ),  $t^* = 101$  for DDPM ( $T = 1000$ ), optimization step size  $\alpha_s = -4$ , and  $\epsilon_t^2$  is followed [55], i.e.,  $\epsilon_t^2 = \log \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \beta_t$ . To balance the range of  $l_{temp}, l_{spa}$ , we set  $\lambda_1 = 5, \lambda_2 = 800$ . Hyper-parameters tuning's details are in Section IV-F. We adopt the RAFT model [60] for optical flow estimation. Our method is implemented using Hugging Face<sup>1</sup>.

### B. Performance Comparison

In this section, we compare the defense performance of seven defense methods against eight adversarial attacks on two benchmark models and datasets. PGD and TT indicate gray-box attacks to evaluate Robust Acc\* of defense methods. PGD+BPDA, TT+BPDA, EOT+BPDA and PGD+Diffattack, AutoAttack+BPDA indicate the adaptive attacks to evaluate Robust Acc of defense methods. We also select TT+MI as the black-box attack. All attacks use the widely popular  $l_\infty$

norm. For gray-box and adaptive attacks,  $\epsilon_{adv} = 4/255$ , and for black-box attacks,  $\epsilon_{adv} = 16/255$ .

**1) PGD & PGD+BPDA:** PGD is the most powerful first-order white-box attack method and is an effective tool to evaluate the performance of defense methods. As shown in Table I, we find: 1) PGD shows strong attack performance on video models without defense methods. 2) JPEG and WN have almost no defense effect against PGD and PGD+BPDA. 3) Temporal Shuffle shows high Robust Acc\* under PGD attack, but when facing the adaptive attack of PGD+BPDA, it shows weak defense performance. 4) Of the two methods in the Diffpure, Diffpure\_DDPM shows better performance due to the deeper gradient chain and stochasticity. 5) Compared to the second-best Diffpure\_DDPM, our method holds the Robust Acc\* steady (a decrease of 3.8% on average), and significantly enhances the Robust Acc (an increase of 19.1% on average).

**2) TT & TT+BPDA:** TT is specifically crafted for video recognition models. It disrupts the temporal information of videos by combining gradients from videos subjected to various shuffles. When left undefended, TT performs similarly to PGD in attacks. However, its attack efficacy against defense methods is notably enhanced. JPEG and WN remain the weak defense performance. Temporal Shuffle enhances defense by introducing randomness to input, yet TT's adaptability to

<sup>1</sup>Hugging Face, <https://huggingface.co/ali-vilab/text-to-video-ms-1.7b>

TABLE II  
DEFENSE PERFORMANCE ON AUTOATTACK

Defense Method	Robust Acc
Temporal Shuffle	3.3
Diffpure_DDPM	26.7
Ours	<b>73.3</b>

different temporal patterns results in significant degradation compared to PGD. For instance, with NL/UCF101, Robust Acc\* of Temporal Shuffle decrease by 71.2% under TT. Our method outperforms Diffpure\_DDPM in both Robust Acc\* and Robust Acc. In comparison to Diffpure\_DDPM, our Robust Acc\* is lower but Robust Acc is higher by 13.6% on average. It's worth noting that our method's defense performance against TT attacks shows a reduction compared to PGD. This is likely due to the distortion introduced by TT in the process of optical flow estimation, which can cause deviations during spatial-temporal optimization. In the next section, we will further explore the impact of attacks against optical flow on our defense methods.

3) **EOT+BPDA**: EOT is explicitly tailored for defense mechanisms incorporating randomness. Consequently, there exists a notable decline in performance for methods such as Temporal Shuffle and the Diffpure series. Our method does not incorporate randomness throughout its entirety. VideoPure employs temporal DDIM inversion for the diffusion process and DDIM sampling for denoising. In essence, both the diffusing and denoising procedures rely on predicted noises by the denoising model. As a result, EOT's effectiveness is diminished against our method.

4) **PGD+Diffattack**: Diffattack is specifically devised to address challenges associated with inaccurate gradient estimation and memory overflow encountered in Diffpure's complex computational graphs, thereby achieving enhanced adaptive attack capabilities. Diffattack introduces a segment-wise forward-backward algorithm to enable memory-efficient gradient backpropagation. Consequently, when compared to PGD+BPDA, PGD+Diffattack demonstrates superior attack efficacy against most defense strategies owing to more precise gradient estimation. Furthermore, in contrast with various defense methods, our method maintains superior defense performance, surpassing Diffpure\_DDPM and Diffpure\_DDIM by margins of 6%, and 47% respectively.

5) **AutoAttack**: We also verify the performance of different defense methods against stronger adaptive attacks, AutoAttack. Considering the large time cost (e.g. 10.3 hours for APGD-CE on one video), we randomly select 30 videos on UCF101 for testing. As shown in Table II, our method still maintains a significant defense performance under stronger adaptive attacks.

6) **Black-box Attack**: To comprehensively evaluate the performance of our method, we also investigate its defense performance against black-box attacks, where the attacker can not get knowledge of the video recognition model and defense method. Specifically, we employ TT+MI [36], [61] as black-box attack and conduct two experiments in two datasets, respectively: adversarial examples generated by NL

TABLE III  
DEFENSE PERFORMANCE UNDER BLACK-BOX ATTACKS.

None	JPEG	Temporal Shuffle	Diffpure DDPM	DDIM Inversion	Ours
32.3	34.5	33.3	48.7	43.0	<b>50.8</b>

to attack SlowFast and adversarial examples generated by SlowFast to attack NL, thereby simulating a black-box attack scenario. We adhere to the default black-box attack parameters with  $\epsilon_{adv} = 16/255$  and  $step = 10$ . The average defense performance are presented in Table III, revealing that our method maintains superior performance under black-box attack settings. Compared to the baseline DDIM inversion, our method demonstrates an 8% improvement in black-box defense performance.

7) **Overall**: In summary, JPEG and WN exhibit minimal defense effectiveness across all models and datasets. Temporal Shuffle experiences notable performance decline when confronted with attacks targeting randomness or temporal aspects, and has weak defense performance against adaptive attacks. Diffpure\_DDPM outperforms Diffpure\_DDIM consistently across various attack methods, yet both exhibit substantial performance degradation against EOT attacks designed for randomness. Our method demonstrates robust defense performance against all attack types. Table I showcases our method's optimal average defense performance across each model and dataset.

### C. Ablation Studies

In this section, we undertake a series of experiments to validate the functionalities of various modules and designs within our method. All experiments are performed utilizing PGD and PGD+BPDA attacks ( $l_\infty$  norm), employing the NL model, and the UCF-101 dataset.

1) **Key Modules**: We validate the efficacy of temporal DDIM inversion, spatial-temporal optimization, and multi-step voting in enhancing defense performance, as shown in Table IV. Firstly, comparing the first and the second line, temporal DDIM inversion effectively suppresses the reconstruction ability of DDIM inversion towards the video, preserving the original video structure and cleaning adversarial noise, thereby achieving high Robust Acc\* and Standard Acc. Secondly, comparing the second and the third line, we observe multi-step voting improves Robust Acc because of the existence of gradient estimation bias in adaptive attacks, wherein the model-estimated gradient may become trapped in the attack's local optimum for one of the candidate solutions ( $z_0^t$ ) in the voting list. Regarding the roles of spatial-temporal optimization, when comparing the second and fourth lines, it primarily enhances resistance against adaptive attacks, resulting in a 59% increase. However, this improvement comes at the cost of decreased Clean Acc and Robust Acc\*. Finally, the combination of multi-step voting and spatial-temporal optimization mitigates the negative impact of spatial optimization on Clean Acc and Robust Acc\* performance, thereby achieving robust defense performance.

TABLE IV  
ABLATION STUDY ON KEY MODULES.

Temporal DDIM Inversion	Spatial-temporal Optimization	Multi-step Voting	Standard Acc	Robust Acc*	Robust Acc
✓			<b>98.0</b>	36.6	13.9
✓		✓	97.0	94.1	39.6
✓	✓	✓	97.0	<b>95.1</b>	69.3
✓	✓	✓	89.1	85.1	99.0
✓	✓	✓	96.0	94.1	<b>100</b>

TABLE V  
ABLATION STUDY ON DIFFUSION MODELS.

Diffusion Model	Defense	Standard Acc	Robust Acc*	Robust Acc
Stable Diffusion	Diffpure_DDPM	49.5	30.7	22.8
	Diffpure_DDIM	63.4	49.5	35.6
Guided Diffusion	Diffpure_DDPM	86.1	80.2	31.7
	Diffpure_DDIM	92.1	68.3	16.8
ModelScope	Diffpure_DDPM	94.1	88.2	72.3
	Diffpure_DDIM	<b>96.0</b>	85.1	41.6
	Ours	<b>96.0</b>	<b>94.1</b>	<b>100</b>

2) **Diffusion Model:** To examine the influence of image diffusion models and video diffusion models on defense performance, we assess the efficacy of the latent diffusion model (Stable Diffusion V1.5 [62]) and pixel-level diffusion model (Guided Diffusion [63]) in comparison to our adopted video diffusion model. As depicted in Table V, whether employing Stable Diffusion or Guided Diffusion, the performance falls short of our adopted video diffusion model, particularly concerning Robust Acc. The weak defense performance of stable diffusion primarily results from significant disruption to the original image structure (notably, Diffpure\_DDPM achieves only 49.5% Standard Acc). Although Guided Diffusion demonstrates respectable Standard Acc and Robust Acc\*, it still exhibits inadequate defense capabilities against adaptive attacks. Modelscope, a video diffusion model tailored specifically for video input, excels in learning rich temporal information from videos. Consequently, following diffusion and denoising, it better preserves the original video structure (leading to improved Standard Acc). In summary, compared to the two image diffusion models, the video diffusion model consistently showcases superior defense performance.

3) **Attacking Video Recognition and Optical Flow Models:** To further confirm the efficacy of our method, we conduct attacks on both video recognition models and optical flow models to generate adversarial examples. Table VI presents a performance comparison between our method and Diffpure. It is evident that the perturbation of optical flow does impact our method's performance. Nonetheless, when compared to Diffpure, our method consistently demonstrates superior defense performance.

4) **Single-Step Prediction or Multi-step Voting:** To further evaluate the efficacy of multi-step voting, we conduct a comparative experiment as shown in Table VII. In this experiment, we select one candidate solution  $z_0^t$  from the voting list to serve as the purified result and compare it with the multi-step voting approach. Our findings reveal that different candidate solutions have varying impacts on the final performance, with optimal results typically achieved from intermediate steps. We suggest that  $z_0^t$  at the midpoint strikes an effective balance

TABLE VI  
DEFENSE PERFORMANCE WHEN ATTACKING BOTH VIDEO RECOGNITION AND OPTICAL FLOW MODEL.

Defense Method	Robust Acc*	Robust Acc
Diffpure_DDIM	86.1	67.3
Diffpure_DDPM	88.1	79.2
DDIM Inversion	53.5	52.5
Ours	<b>89.1</b>	<b>97.0</b>

TABLE VII  
DEFENSE PERFORMANCE OF SINGLE STEP PREDICTION OR MULTI-STEP VOTING

selected $z_0^t$	Standard Acc	Robust Acc*	Robust Acc
6	95.0	91.1	100
5	96.0	91.1	100
4	96.0	91.1	100
3	96.0	92.1	100
2	92.1	89.1	100
1	90.1	87.1	100
0	89.1	85.1	99.0
multi-step voting	96.0	94.1	100

between preserving the original input's inherent structure and effectively eliminating adversarial noise. Ultimately, multi-step voting consistently demonstrates superior performance.

5)  $z'_t$  replaces  $z_t^*$ : In Algorithm 1 (line 6), the optimized  $z'_t$  is not directly substituted for  $z_t^*$  during denoising. In other words, our optimization process does not directly influence the denoising procedure. Therefore, Table VIII examines the performance impact when  $z_t^*$  is replaced with  $z'_t$  for denoising. It is evident that such a substitution significantly disrupts the normal denoising process, resulting in greater distortion to the original structure (Standard Acc decrease 17%). Consequently, we opt to preserve the regular denoising trajectory and solely optimize the latent requiring voting.

6) **Loss of spatial-temporal optimization:** We verify the effects of  $L_{temp}$  and  $L_{spa}$  on defense performance, as shown in Table IX. We find that  $L_{spa}$  can greatly improve Robust Acc, but it affects Standard Acc and Robust Acc\*. Thus,  $L_{temp}$  exactly balances this effect.

#### D. Effectiveness Analysis

In this section, we analyze the effectiveness of our method in terms of defense performance when facing adaptive attacks. We choose PGD+BPDA as the adaptive attack method and NL as the video recognition model. During the adaptive attack process, the attacker estimates the overall gradient of the defense method and video recognition model at each iteration, thereby increasing the loss. On video recognition, this loss is

TABLE VIII  
ABLATION STUDY ON  $z'_t$  WHEN  $z'_t$  REPLACES  $z_t^*$ . OURS\* INDICATES  $z'_t$  REPLACES  $z_t^*$  WHEN DENOISING.

Defense Method	Standard Acc	Robust Acc*	Robust Acc
Ours*	79.2	66.3	76.2
Ours	<b>96.0</b>	<b>94.1</b>	<b>100</b>

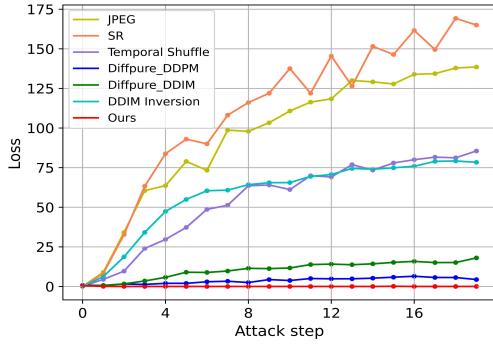


Fig. 4. Loss changes of different defense methods with attack steps under PGD+BPDA attack on NL model and UCF101 dataset.

TABLE IX

ABALATION STUDY ON LOSS OF SPATIAL-TEMPORAL OPTIMIZATION

$L_{temp}$	$L_{spa}$	Standard ACC	Robust Acc*	Robust Acc
	✓	<b>97.0</b>	<b>95.1</b>	71.3
✓	✓	95.0	90.1	100
	✓	96.0	94.1	<b>100</b>

cross-entropy loss. When the loss rises to a certain extent, the video recognition model will make incorrect judgments about the input. Therefore, a good adversarial defense method should be able to suppress this gradient rise. As shown in the Fig. 4, we randomly select 20 videos from UCF101 for testing and calculate the mean change of the loss in each video during 20 iterations. Combining Table I, we find the performance of the JPEG, SR, DDIM inversion, and Temporal Shuffle, which perform poorly, presents an apparent increase in loss, with just a few iterative steps making the loss skyrocket, thus rendering it incapable of effective defense. The relatively effective defense methods such as Diffpure\_DDPM slow down the rise of loss, but none of them successfully suppresses it. Our method successfully arrest the rise in loss, making the loss fluctuate within a small range, thus achieving a better adaptive defense performance.

#### E. Time Cost

We verify the time cost comparison between our method and Diffpure\_DDPM on one NVIDIA GeForce 3090 GPU, as shown in Table X. Our method not only excels in defense performance but also enjoys a speed advantage of nearly four times.

#### F. Diagnostic Experiments

In this section, we discuss the impact of different attack  $\epsilon_{adv}$ , attack steps and norm constraints on the defense performance of our method. In addition, we also conduct hyper-parameters tuning on optimization step size  $\alpha_s$  and diffusion timestep  $t^*$  of our method.

##### 1) Attack $\epsilon_{adv}$ & Norm Constraint $l_2$ & Attack Step:

Table XI illustrates the influence of attack  $\epsilon_{adv}$  on defense performance. A larger perturbation range increases the denoising difficulty, resulting in a slight decline in our method's Robust Acc\*, although without impacting our Robust Acc.

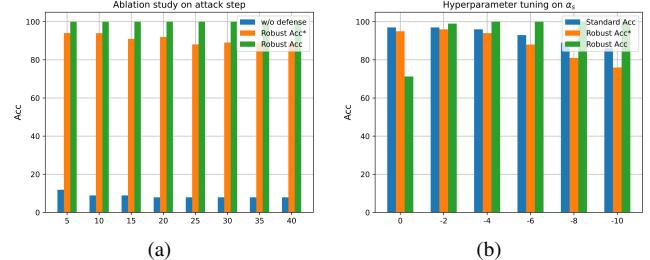


Fig. 5. (a) shows the effects of attack steps under PGD & PGD+BPDA on NL model and UCF101 dataset. (b) shows hyper-parameters tuning on optimization step size.

TABLE X  
COMPARISON OF TIME COST

Defense Method	Time(s)
Diffpure_DDPM	37.8
Ours	<b>8.5</b>

Table XII presents the defense performance of our method compared to other methods under the  $l_2$  norm constraint. It is evident that varying norm constraints do not undermine the superiority of our method. Fig. 5 (a) depicts the defense performance of our method under different attack steps. We observe slight fluctuations in Robust Acc\* while Robust Acc remains unaffected. In summary, we evaluate our defense method from three attack perspectives, further substantiating its generalization capability.

2) **Diffusion Timestep:** Fig. 6 shows the impact of diffusion timestep  $t^*$  on the defense performance of our method. We find that the Standard Acc overall shows a downward trend, and the Robust Acc\* also demonstrates a fluctuating downward trend with the timestep. This is because larger timestep, while providing greater coverage for adversarial noise, will also cause more substantial destruction to the original structure of the video, leading to a performance decrease. Robust Acc with respect to timestep shows an increasing trend, as larger timestep bring more candidate solutions for voting, further increasing the difficulty for the attacker to estimate the gradient. Considering all three metrics, we choose  $t^* = 6$ .

3) **Optimization Step Size  $\alpha_s$ :** Fig. 5 (b) shows the impact of different optimization step size on performance. The

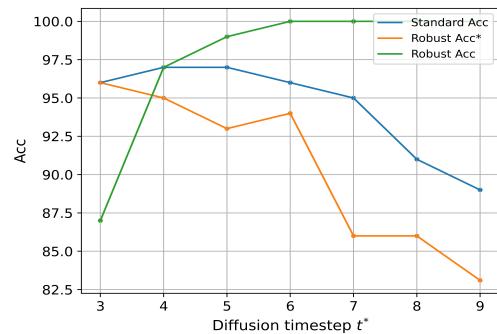


Fig. 6. Hyper-parameters tuning on diffusion timestep.

TABLE XI  
DEFENSE PERFORMANCE UNDER DIFFERENT ATTACK  $\epsilon_{adv}$

Defense Method	$\epsilon_{adv}$	w/o Defense	Robust Acc*	Robust Acc
Diffpure_DDPM	4	8.9	88.2	72.3
	8	5.9	82.2	40.6
	16	2.0	67.3	32.7
Ours	4	8.9	94.1	100
	8	5.9	85.1	100
	16	2.0	73.3	100

TABLE XII  
DEFENSE PERFORMANCE ON  $l_2$  ATTACK

Defense Method	w/o Defense	Robust Acc*	Robust Acc
Diffpure_DDIM	21.8	93.1	89.1
Diffpure_DDPM	21.8	93.1	93.1
DDIM Inversion	21.8	73.3	53.5
Ours	21.8	<b>96.0</b>	<b>97.0</b>

greater the degree of spatial-temporal optimization, the better performance on Robust Acc, but it affects the performance of Standard Acc and Robust Acc\*. Considering the overall defense performance, we select  $\alpha_s = -4$ .

## V. CONCLUSION

In this paper, we introduce VideoPure, a novel diffusion-based adversarial purification framework for video recognition models. Our framework enhances adversarial defense through temporal DDIM inversion, spatial-temporal optimization, and multi-step voting strategies. Temporal DDIM inversion ensures the recognition accuracy of clean data, while the spatial-temporal optimization and multi-step voting strategies disrupt attack optimization, significantly enhancing our method's defense capabilities. We validate the defense performance of our approach against state-of-the-art (SOTA) attack methods under various attack settings. Our method demonstrates superior defense performance against image attack methods, video attack methods, attacks designed for randomness, and those targeting diffusion models. Additionally, we conduct extensive experiments to verify the effectiveness of each module and the overall efficiency of our method. In conclusion, our method is an effective and flexible adversarial defense tool that easily integrates into various video recognition models.

## REFERENCES

- [1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.
- [2] Z. Zhou, X. Zhou, Z. Chen, P. Guo, Q. Liu, and W. Zhang, "Memory network with pixel-level spatio-temporal learning for visual object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 11, pp. 6897–6911, 2023.
- [3] Z. Chen, B. Li, J. Xu, S. Wu, S. Ding, and W. Zhang, "Towards practical certifiable patch defense with vision transformer," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 15 148–15 158.
- [4] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Int. Conf. Learn. Represent.*, 2018.
- [5] Y. Ran, W. Wang, M. Li, L. Li, Y. Wang, and J. Li, "Cross-shaped adversarial patch attack," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2289–2303, 2024.
- [6] A. V. Subramanyam, "Meta generative attack on person reidentification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 4429–4434, 2023.
- [7] Z. Chen, B. Li, S. Wu, J. Xu, S. Ding, and W. Zhang, "Shape matters: Deformable patch attack," in *Eur. Conf. Comput. Vis.*, 2022, p. 529–548.
- [8] Z. Chen, B. Li, S. Wu, S. Ding, and W. Zhang, "Query-efficient decision-based black-box patch attack," *Trans. Info. For. Sec.*, vol. 18, p. 5522–5536, jan 2023.
- [9] X. Wei, J. Zhu, S. Yuan, and H. Su, "Sparse adversarial perturbations for videos," in *AAAI*, vol. 33, no. 01, 2019, pp. 8973–8980.
- [10] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The apolloscope dataset for autonomous driving," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2018, pp. 954–960.
- [11] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt *et al.*, "A system for video surveillance and monitoring," *VSAM final report*, vol. 2000, no. 1–68, p. 1, 2000.
- [12] S. Gowal, C. Qin, J. Uesato, T. Mann, and P. Kohli, "Uncovering the limits of adversarial training against norm-bounded adversarial examples," *arXiv preprint arXiv:2010.03593*, 2020.
- [13] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *Int. Conf. Mach. Learn.* PMLR, 2019, pp. 7472–7482.
- [14] S. Gowal, S.-A. Rebuffi, O. Wiles, F. Stimberg, D. A. Calian, and T. A. Mann, "Improving robustness using generated data," *Adv. Neural Inform. Process. Syst.*, vol. 34, pp. 4218–4233, 2021.
- [15] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," in *Int. Conf. Learn. Represent.*, 2016.
- [16] S.-Y. Lo and V. M. Patel, "Defending against multiple and unforeseen adversarial videos," *IEEE Trans. Image Process.*, vol. 31, pp. 962–973, 2021.
- [17] K. A. Kinfu and R. Vidal, "Analysis and extensions of adversarial training for video classification," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, June 2022, pp. 3416–3425.
- [18] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," in *Int. Conf. Learn. Represent.*, 2018.
- [19] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman, "Pixeldefend: Leveraging generative models to understand and defend against adversarial examples," in *Int. Conf. Learn. Represent.*, 2018.
- [20] H. J. Lee and Y. M. Ro, "Defending video recognition model against adversarial perturbations via defense patterns," *IEEE Trans. Dependable Secure Comput.*, 2023.
- [21] N. Das, M. Shanbhogue, S.-T. Chen, F. Hohman, S. Li, L. Chen, M. E. Kounavis, and D. H. Chau, "Shield: Fast, practical defense and vaccination for deep learning using jpeg compression," in *ACM SIGKDD*, 2018, pp. 196–204.
- [22] Y. Cheng, X. Wei, H. Fu, S.-W. Lin, and W. Lin, "Defense for adversarial videos by self-adaptive jpeg compression and optical texture," in *ACM Int. Conf. Multimedia Asia*, 2021, pp. 1–7.
- [23] J. Hwang, H. Zhang, J.-H. Choi, C.-J. Hsieh, and J.-S. Lee, "Temporal shuffling for defending deep action recognition models against adversarial attacks," *Neural Networks*, vol. 169, pp. 388–397, 2024.
- [24] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, "Diffusion models for adversarial purification," in *Int. Conf. Mach. Learn.* PMLR, 2022, pp. 16 805–16 827.
- [25] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *Int. Conf. Mach. Learn.* PMLR, 2018, pp. 274–283.
- [26] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *Int. Conf. Mach. Learn.* PMLR, 2018, pp. 284–293.
- [27] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *Int. Conf. Mach. Learn.* PMLR, 2020, pp. 2206–2216.
- [28] T. Blau, R. Ganz, B. Kawar, A. Bronstein, and M. Elad, "Threat model-agnostic adversarial defense using diffusion models," *arXiv preprint arXiv:2207.08089*, 2022.
- [29] Q. Wu, H. Ye, and Y. Gu, "Guided diffusion model for adversarial purification from random noise," *arXiv preprint arXiv:2206.10875*, 2022.
- [30] J. Wang, Z. Lyu, D. Lin, B. Dai, and H. Fu, "Guided diffusion model for adversarial purification," *arXiv preprint arXiv:2205.14969*, 2022.
- [31] B. Zhang, W. Luo, and Z. Zhang, "Enhancing adversarial robustness via score-based optimization," *Adv. Neural Inform. Process. Syst.*, vol. 36, 2024.
- [32] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Adv. Neural Inform. Process. Syst.*, vol. 33, pp. 6840–6851, 2020.

- [33] J. Wang, H. Yuan, D. Chen, Y. Zhang, X. Wang, and S. Zhang, "Mod-elscope text-to-video technical report," *arXiv preprint arXiv:2308.06571*, 2023.
- [34] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *Int. Conf. Learn. Represent.*, 2020.
- [35] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Int. Conf. Learn. Represent.*, 2014.
- [36] Z. Wei, J. Chen, Z. Wu, and Y.-G. Jiang, "Boosting the transferability of video adversarial examples via temporal translation," in *AAAI*, vol. 36, no. 3, 2022, pp. 2659–2667.
- [37] R. Wang, Y. Guo, and Y. Wang, "Global-local characteristic excited cross-modal attacks from images to videos," in *AAAI*, vol. 37, no. 2, 2023, pp. 2635–2643.
- [38] Z. Wei, J. Chen, Z. Wu, and Y.-G. Jiang, "Cross-modal transferable adversarial attacks from images to videos," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 15 064–15 073.
- [39] L. Jiang, X. Ma, S. Chen, J. Bailey, and Y.-G. Jiang, "Black-box adversarial attacks on video recognition models," in *ACM Int. Conf. Multimedia*, 2019, pp. 864–872.
- [40] H. Zhang, L. Zhu, Y. Zhu, and Y. Yang, "Motion-excited sampler: Video adversarial attack with sparked prior," in *Eur. Conf. Comput. Vis.* Springer, 2020, pp. 240–256.
- [41] K. Jiang, Z. Chen, H. Huang, J. Wang, D. Yang, B. Li, Y. Wang, and W. Zhang, "Efficient decision-based black-box patch attacks on video recognition," in *Int. Conf. Comput. Vis.*, 2023, pp. 4379–4389.
- [42] K. Jiang, Z. Chen, X. Zhou, J. Zhang, L. Hong, J. Wang, B. Li, Y. Wang, and W. Zhang, "Towards decision-based sparse attacks on video recognition," in *ACM Int. Conf. Multimedia*, 2023, pp. 1443–1454.
- [43] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*. Chapman and Hall/CRC, 2018, pp. 99–112.
- [44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [45] J. Hwang, J.-H. Kim, J.-H. Choi, and J.-S. Lee, "Just one moment: Structural vulnerability of deep action recognition against one frame attack," in *Int. Conf. Comput. Vis.*, 2021, pp. 7668–7676.
- [46] R. Pony, I. Naeh, and S. Mannor, "Over-the-air adversarial flickering attacks against video recognition networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 515–524.
- [47] M. Kang, D. Song, and B. Li, "Diffattack: Evasion attacks against diffusion-based adversarial purification," *Adv. Neural Inform. Process. Syst.*, vol. 36, 2024.
- [48] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.
- [49] A. Mustafa, S. H. Khan, M. Hayat, J. Shen, and L. Shao, "Image super-resolution as a defense against adversarial attacks," *IEEE Trans. Image Process.*, vol. 29, pp. 1711–1724, 2019.
- [50] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 12 873–12 883.
- [51] R. Mokady, A. Hertz, K. Aberman, Y. Pritch, and D. Cohen-Or, "Null-text inversion for editing real images using guided diffusion models," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 6038–6047.
- [52] Z. Chen, B. Li, S. Wu, K. Jiang, S. Ding, and W. Zhang, "Content-based unrestricted adversarial attack," in *Adv. Neural Inform. Process. Syst.*, vol. 36, 2023, pp. 51 719–51 733.
- [53] Y. Huang, J. Huang, Y. Liu, M. Yan, J. Lv, J. Liu, W. Xiong, H. Zhang, S. Chen, and L. Cao, "Diffusion model-based image editing: A survey," *arXiv preprint arXiv:2402.17525*, 2024.
- [54] L. Khachatryan, A. Movsisyan, V. Tadevosyan, R. Henschel, Z. Wang, S. Navasardyan, and H. Shi, "Text2video-zero: Text-to-image diffusion models are zero-shot video generators," in *Int. Conf. Comput. Vis.*, 2023, pp. 15 954–15 964.
- [55] X. Yang, C. He, J. Ma, and L. Zhang, "Motion-guided latent diffusion for temporally consistent real-world video super-resolution," *arXiv preprint arXiv:2312.00853*, 2023.
- [56] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.
- [57] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 6299–6308.
- [58] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7794–7803.
- [59] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Int. Conf. Comput. Vis.*, 2019, pp. 6202–6211.
- [60] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *Eur. Conf. Comput. Vis.* Springer, 2020, pp. 402–419.
- [61] Y. Dong, F. Liao, T. Pang, X. Hu, and J. Zhu, "Discovering adversarial examples with momentum," *arXiv preprint arXiv:1710.06081*, vol. 5, 2017.
- [62] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 10 684–10 695.
- [63] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Adv. Neural Inform. Process. Syst.*, vol. 34, pp. 8780–8794, 2021.