Correction-based Defense Against Adversarial Video Attacks via Discretization-Enhanced Video Compressive Sensing

Wei Song, Cong Cong, Haonan Zhong, and Jingling Xue, *UNSW Sydney*

https://www.usenix.org/conference/usenixsecurity24/presentation/song-wei

This paper is included in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Correction-based Defense Against Adversarial Video Attacks via Discretization-Enhanced Video Compressive Sensing

Wei Song
UNSW Sydney

Cong Cong
UNSW Sydney

Haonan Zhong
UNSW Sydney

Jingling Xue
UNSW Sydney

Abstract

We introduce SECVID, a correction-based framework that defends video recognition systems against adversarial attacks without prior adversarial knowledge. It uses discretization-enhanced video compressive sensing in a black-box pre-processing module, transforming videos into a sparse domain to disperse and neutralize perturbations. While SECVID's discretized compression disrupts perturbation continuity, its reconstruction process minimizes adversarial elements, causing only minor distortions to the original videos. Though not completely restoring adversarial videos, SECVID significantly enhances their quality, enabling accurate classification by SECVID-enhanced video classifiers and preventing adversarial attacks. Tested on C3D and I3D with the UCF-101 and HMDB-51 datasets against five types of advanced video attacks, SECVID outperforms existing defenses, improving detection accuracy by 38.5% to 866.2%. Specifically designed for high-risk environments, SECVID addresses trade-offs like minor accuracy reduction, additional pre-processing training, and longer inference times, with potential optimization through selective security impacting strategies.

1 Introduction

Deep Neural Networks (DNNs) are pivotal in video recognition, applied in areas like face recognition [93], action distinction [24], autonomous driving [61, 68], and visual anomaly detection [9, 77]. Additionally, they aid urban safety in video surveillance by detecting accidents or illegal activities [77].

Despite significant advancements, DNNs remain vulnerable to adversarial attacks [5, 6, 55, 56, 90]. For instance, subtly altered stop signs can mislead DNNs in autonomous vehicles, risking catastrophic outcomes. In surveillance, clever adversarial perturbations might hide criminal activities [9]. These are not hypothetical risks; real-world examples have highlighted such vulnerabilities [7, 90].

Given the susceptibility of DNNs to adversarial attacks, the development of robust defenses is increasingly vital, as DNNs gain prevalence in video applications [10, 56, 61, 68, 90]. Most current solutions [16, 27, 37, 60, 71] are designed for static images and may struggle in video attack scenarios [42, 78]. Addressing this gap requires overcoming several key limitations to create effective video defense systems, including: (1) **Temporal Inadequacy:** Existing image-focused strategies [16, 27, 37, 60, 71] overlook video's dynamic aspects, reducing their effectiveness in video contexts [5, 42, 90]; (2) **Model Intrusiveness:** Some image-based methods [34, 45] necessitate extensive model alterations to the original model architecture, compromising model integrity; and (3) **Prior Knowledge of Adversarial Perturbations:** Some image-based methods [27, 71], which resort to adversarial training, rely on numerous adversarial samples and intensive training to effectively counter specific perturbations.

AdvIT [89], the first video-specific defense, detects adversarial videos by checking frame-to-frame temporal consistency. However, it is less effective against sophisticated attacks such as U3D [90] and StyleFool [5] that also maintain temporal coherence. While adversarial training, originally designed for images [27, 71], has expanded to videos [42, 78], it depends on known adversarial samples and faces the significant costs of model retraining. Thus, there is a critical need for new, more effective video defense strategies.

In this paper, we introduce SECVID, an advanced correction-based pre-processing defense framework utilizing discretization-enhanced video compressive sensing (VCS) [14, 85, 92, 94] to safeguard video recognition systems against adversarial attacks. Operating as a seamless, non-intrusive black-box module, SECVID effectively neutralizes adversarial threats without requiring specific perturbation knowledge. It integrates smoothly with existing classifiers by leveraging their original training datasets, avoiding the need for direct access, and is adept at countering sophisticated attacks like StyleFool [5], U3D [90], and Geo-Trap [48]. Unlike solutions such as AdvIT [89] and OUDefend [51], SECVID not only restores adversarial videos but also maintains the quality and fidelity of the original content, allowing enhanced video classifiers to accurately perform video classification and effectively counter adversarial perturbations.

SecVID's key insight lies in its innovative application of VCS as a defensive mechanism against adversarial perturbations, a technique originally for video compression, this technique has been adeptly repurposed for enhancing security. VCS operates through sparse transformation and compression, efficiently diffusing adversarial noise throughout the video's spatio-temporal spectrum. This process creates a dispersal effect, characterized by Sparsity Change (SC) [98], Intensity Redistribution (IR) [66], and Positional Redistribution (PR) [11], which collectively diminishes adversarial impacts through differential emphasis (SC), energy modulation (IR), and data repositioning post-transformation (PR). The unique dispersal ability of VCS [14, 85, 92, 94], underpinned by compressive sensing theory [20, 84, 98], is pivotal in neutralizing adversarial video attacks. Additionally, VCS's compression phase plays a crucial role in further filtering out perturbations, thereby fortifying the defense mechanism.

Building on the foundational role of VCS, our research enhances its effectiveness with discretized compression, a technique more effective than traditional compression in disrupting adversarial perturbations. This dual strategy of discretization and compression disrupts the continuity of adversarial perturbations typical in continuous data like images, videos, and audios [18, 29, 47, 54]. By shifting the video's sparse representation from a continuous to a segmented discrete space, our approach not only filters but also structurally compromises adversarial smoothness, significantly enhancing VCS's defensive capabilities.

Disrupting and filtering out perturbations are key, but SecVID's capability to restore high-quality video from discretized forms is equally crucial for accurate video classification and defeating attacks. After sparse transformation and discretized compression, our focus shifts to robust video reconstruction to maintain quality and fidelity. This process, vital for mitigating fidelity-degrading adversarial perturbations [5], employs a multi-faceted loss function to effectively maximize the restoration of the video.

Tested on C3D and I3D, two premier video classifiers, using the UCF-101 and HMDB-51 datasets, SecVID significantly surpasses two video-specific defenses—AdvIT [89] and OUDefend [51]—and five image-based defenses adapted for videos: Adversarial Training [27, 42, 71, 78], Input Transformations [28], Random Smoothing [34], ComDefend [37], and DiffPure [60], boosting accuracy by 38.5% – 866.2%. SecVID also robustly defends against sparse adversarial attacks [4]. Tailored for high-risk security environments, SecVID provides robust defense but with some drawbacks like reduced recognition accuracy, the need for additional training of its pre-processing module, and longer inference times. These can potentially be mitigated by future optimizations that may impact security (as discussed in Section 7).

In summary, this paper makes the following contributions:

- **Video-Centric Defense.** SecVID is the first defense leveraging VCS to protect video classifiers from adversarial attacks, serving as a pre-processing solution that corrects videos without requiring knowledge of specific perturbations or necessitating retraining of downstream classifiers. It outperforms leading defenses such as AT (Adversarial Training) [27, 71, 78] and DiffPure [60] with average improvements of 61.8% and 46.6%, respectively, effectively neutralizing major attacks like StyleFool [5], U3D [90], and Geo-Trap [48]. Additionally, SecVID excels against sparse adversarial attacks [4] designed to target it.

- **Discretization-Enhanced VCS.** We establish the first link between VCS theory [14, 85, 92, 94] and defense strategies against adversarial video attacks with SecVID. Additionally, we integrate a novel discretized compression strategy into VCS, boosting SecVID's defense capabilities by an average of 29.5% while maintaining the quality and fidelity of reconstructed videos.

- **Comprehensive Testing.** SecVID counters adversarial video attacks with minimal impact on recognition accuracy and requires moderate additional training. Inference times range from 1.71× to 4.40×, significantly lower than DiffPure's [60] 43.06× to 73.82×. These times could be further reduced through targeted security strategies.

2 Background

2.1 DNN-based Video Classification Systems

DNNs have transformed video recognition, influencing sectors like anomaly detection [77], autonomous vehicles [61], and smart security cameras [40]. Architectures such as LRCN [21] and C3D [79] combine convolutional layers with temporal elements, while advanced models like TSN [73] utilize two-stream inflated filters. Spatial-temporal networks like CNN+LSTM [95], C3D [79], and I3D [8] have achieved high benchmarks. Addressing the vulnerabilities of these models, now vital to safety systems, is essential.

2.2 Video Compressive Sensing

Video Compressive Sensing (VCS) [14, 85, 92, 94] is a technique for video acquisition and processing, aiming to reconstruct a signal $x \in \mathbb{R}^N$ from randomized measurements $y = \Phi x$, where $\Phi \in \mathbb{R}^{M \times N}$ is a matrix with $M \ll N$. Reconstruction of x from y using Φ requires x to be sparse in an invertible basis Ψ, leading to $x = \Psi \mathcal{S}$ and $y = \Phi \Psi \mathcal{S}$. The challenge in VCS involves reconstructing x from y by addressing an optimization problem to enhance the sparsity in the sparse representation $\mathcal{S}$ of x:

$$\min_{\mathcal{S}} \|\mathcal{S}\|_1 \text{ subject to } y = \Phi \Psi \mathcal{S} \tag{1}$$

This renders most $\mathcal{S}$ coefficients zero or near-zero [14].

Table 1: Comparing SecVID with the state of the art in countering adversarial image/video attacks.

Method	Video-Oriented	Temporal Dynamics Considered	Black-Box	Correction Capability	No Requirement for Prior Adversarial Data	No Model Retraining	No Requirement for Original Dataset
Adversarial Training [27, 42, 71, 78]	✗	✗	✗	✗	✗	✗	✗
Random Smoothing [34]	✗	✗	✗	✗	✓	✗	✗
ComDefend [37]	✗	✗	✓	✓	✓	✓	✗
Compressed&Restore [26]	✗	✗	✗	✓	✗	✓	✗
SESR [3]	✗	✗	✓	✓	✓	✓	✗
DiffPure [60]	✗	✗	✓	✓	✓	✓	✗
FakeDetector [83]	✗	✗	✓	✗	✗	✗	✓
Input Transformations [28]	✗	✗	✓	✗	✓	✓	✗
OUDefend [51]	✓	✓	✗	✗	✓	✓	✗
DP [46]	✓	✓	✗	✗	✗	✓	✗
AdvIT [89]	✓	✓	✓	✗	✓	✓	✗
SecVID [This Paper]	✓	✓	✓	✓	✓	✓	✗

3 Threat Model

A DNN for video classification is represented as:

$$f : x \in \mathbb{R}^{T\times H\times W\times C} \quad \rightarrow \quad y \in \mathbb{R}^{X} \tag{2}$$

where T, H, W, and C are the number of frames, height, width, and channels (typically 3 for RGB) of a video sequence. Given x, f predicts a class y from a set of class labels, X. The adversary seeks an adversarial x^{adv} of x satisfying:

$$f(x^{adv}) = y_t \quad \text{if targeted} \tag{3}$$

$$f(x^{adv}) \neq y_0 \quad \text{if untargeted} \tag{4}$$

where y_t is the target class and y_0 the original class of x. The perturbed video x^{adv} can be defined as follows:

$$x^{adv} = x + \varepsilon \cdot \mathcal{P} \tag{5}$$

with $\mathcal{P}$ and ε representing perturbations and intensity, respectively. In general, x_{adv} is optimized using model gradients [12, 76, 80] while maintaining:

$$B_p(r) = \{\varepsilon \in \mathbb{R} : \|\varepsilon\|_p \leq r\} \tag{6}$$

where r limits the perturbation magnitude under the ℓ_p-norm.

We assume that attackers possess comprehensive knowledge of the target model, encompassing its architecture, parameters, gradients, and training data. They may employ diverse perturbation patterns, including random, Perlin [62], and Gabor [44] noise, alongside various textures and colors. We also consider different attack strategies, including targeted (T) and untargeted attacks (U), varying perturbation intensities, and both universal and one-on-one attacks. Specifically, we focus on five primary attack types: StyleFool (U) [5], U3D (U) [90], Geo-Trap (U) [48], StyleFool (T) [5], and Geo-Trap (T) [48]. Additionally, we consider sparse attacks tailored specifically against SecVID by targeting its sparse transformation using adversarial patches [4]. Further details will be discussed in our experimental evaluation (Section 6.1).

4 Defense Mechanisms

Table 1 outlines leading defenses against image and video attacks, primarily centered on static images [3, 16, 26–28, 37, 60, 71, 83]. DiffPure [60] employs a stochastic diffusion process to purify input images, though it is computationally costly. Adversarial training, initially for images [27, 71], has been adapted to videos [42, 78], but it relies on known adversarial samples and involves high training costs. AdvIT [89], the first black-box defense for videos based on temporal consistency, is susceptible to sophisticated attacks like U3D [90] and StyleFool [5]. OUDefend [51] and DP [46] are defenses tailored to specific video classifier architectures or known perturbation types, reducing their versatility across scenarios.

SecVID serves as a *black-box* pre-processing module for video classifiers, utilizing original training datasets without direct classifier access. Unlike image-centric methods (Table 1), SecVID surpasses DiffPure [60] by employing discretized compression and leveraging a temporal loss to preserve essential consistency and flow in video analysis [5, 33, 81]. Compared to other video-specific defenses listed in Table 1, SecVID utilizes a correction-based approach that exceeds the detection capabilities of solutions like AdvIT [89] and OUDefend [51]. It achieves this enhanced effectiveness without model modifications, unlike Random Smoothing [34], or reliance on adversarial sample learning, typical of adversarial training methods [27, 71], while also avoiding extensive retraining and preserving the original model's integrity.

5 The SecVID Framework

SecVID, illustrated in Figure 1, comprises three main components: sparse transformation, discretized compression, and reconstruction, each described below.

5.1 Sparse Transformation

Sparsity is key for effective signal recovery in VCS [14, 85, 92, 94]. VCS mitigates attacks by converting videos, including adversarially perturbed ones, into a sparse domain. Sparse

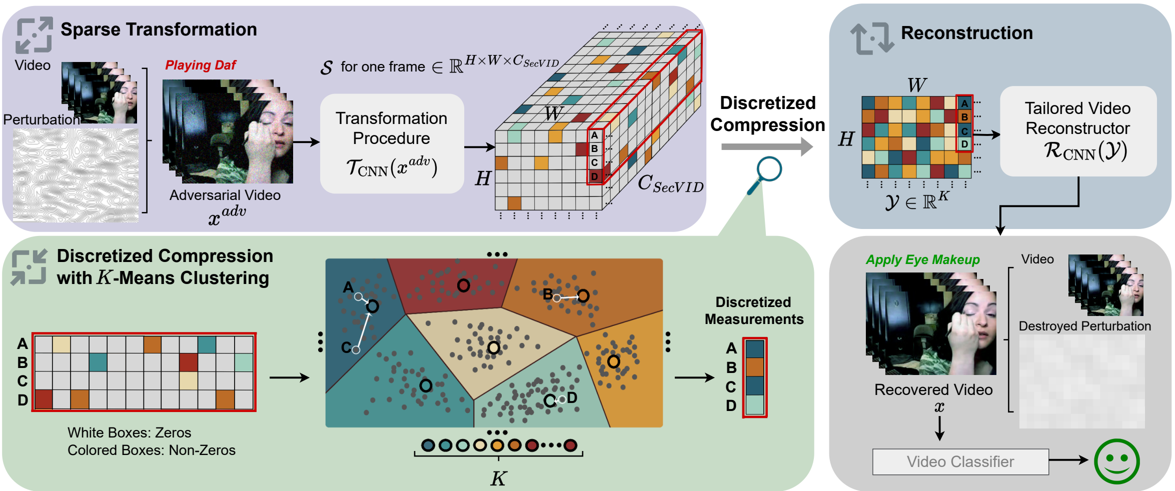


Figure 1: Architecture of SECVID (illustrated for one video frame). An adversarial video $x^{adv} = x + \varepsilon\mathcal{P}$ (Equation (5)) is transformed to a sparse representation $\mathcal{S} = \mathcal{T}_{\mathrm{CNN}}(x^{adv})$, and then undergoes discretized compression, where $\mathcal{S}$ is compressed and discretized via K-Means clustering, disrupting continuous adversarial perturbations and producing compact measurements $\mathcal{Y}$. A CNN reconstructor, guided by content, temporal, perceptual, and sparse transformation losses, recovers the clean video x.

transformation's hallmark is its ability to disperse data. Stemming from VCS's profound dimensionality reduction, this process scatters densely packed data into a format filled with zeros or near-zeros. This shift from compact to expanded, sparse representation drastically alters data distribution in the measurement space, embodying the essence of dispersal.

Definition 1 (Dispersal Effect). *Given an adversarial video x^{adv} represented by a signal consisting of N elements and its transformation $\mathcal{T}(x^{adv})$, the dispersal effect of $\mathcal{T}$ in VCS is characterized by the following three properties [20, 84, 98]:*

- ***Sparsity Change (SC).*** *This reflects the variation in the number of significant coefficients resulting from the transformation, where a coefficient is deemed* significant *if its absolute value exceeds a small positive threshold τ:*

$$SC_{\mathcal{T}}(x^{adv}) = \frac{|\{i \mid |\mathcal{T}(x_i^{adv})| > \tau\}|}{N} - \frac{|\{i \mid |x_i^{adv}| > \tau\}|}{N} \quad (7)$$

- ***Intensity Redistribution (IR).*** *This quantifies the shift in energy or intensity distribution of the transformed signal compared to the original signal:*

$$IR_{\mathcal{T}}(x^{adv}) = \frac{1}{N}\sum_{i=1}^{N} |\mathcal{T}(x_i^{adv})|^2 - \frac{1}{N}\sum_{i=1}^{N} |x_i^{adv}|^2 \quad (8)$$

- ***Positional Redistribution (PR).*** *This metric evaluates the positional shifts of non-zero elements in a signal post-transformation, using, for example, the Wasserstein distance W [32] to calculate the minimal "work" needed to transform one distribution into another. It specifically applies to sets $P_{x^{adv}}$ and $P_{\mathcal{T}(x^{adv})}$, which represent non-zero elements in x^{adv} and $\mathcal{T}(x^{adv})$, respectively:*

$$PR_{\mathcal{T}}(x^{adv}) = W(P_{x^{adv}}, P_{\mathcal{T}(x^{adv})}) \quad (9)$$

Finally, the overall dispersal effect of $\mathcal{T}$ on a given signal x is a composite of SC, IR, and PR [20, 84].

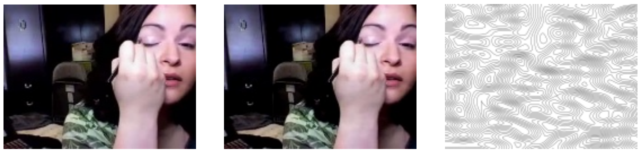
Sparse transformation in VCS efficiently reduces the structural information of perturbations, weakening their effectiveness and rendering their adversarial intent less discernible.

In line with Equation (5), the adversarial video x^{adv} is produced by adding perturbations $\varepsilon \cdot \mathcal{P}$ to a clean video x. We explore how the sparse transformation $\mathcal{T}$ disperses these adversarial perturbations, affecting their structural integrity as indicated by changes in SC, IR, and PR. Experimental validation of these effects is detailed in Section 6.

The structural change in $\varepsilon\mathcal{P}$ induced by $\mathcal{T}$ is quantified by $SC_{\mathcal{T}}(x^{adv})$. This alteration represents a significant shift in the effectiveness of the perturbations in the transformed domain. $SC_{\mathcal{T}}(x^{adv})$ typically exhibits a marked reduction in the density of critical elements in x^{adv} post-transformation. This reduction undermines the ability of the perturbations $\varepsilon\mathcal{P}$ to impact specific features or patterns in the original video x, as critical elements become more dispersed.

$IR_{\mathcal{T}}(x^{adv})$ usually exhibits a noticeable decrease in the energy of x^{adv} after transformation. This reduction in energy aids in mitigating the impact of perturbations $\varepsilon\mathcal{P}$ by more evenly distributing their energy post-transformation. Consequently, their concentrated effect on specific areas of the original video x is lessened, reducing their potential to adversely affect particular segments of the video.

$PR_{\mathcal{T}}(x^{adv})$ reflects a significant positional change of non-zero elements in x^{adv} after transformation. This shift disrupts the alignment of perturbations $\varepsilon\mathcal{P}$ with specific patterns in the original video x, diminishing their intended effect. The Wasserstein Distance typically shows a significant increase, indicating a greater degree of positional change and highlighting



(a) Clean frame (b) Perturbed frame (c) Perturbations

Figure 2: Continuity in adversarial perturbations generated by U3D [90]. The visualized pattern emphasizes the smooth transition and continuous pixel-level variation.

the transformation's effectiveness in displacing perturbation elements from their original, strategically targeted locations.

Together, the changes captured by $SC_{\mathcal{T}}(x^{adv})$, $IR_{\mathcal{T}}(x^{adv})$, and $PR_{\mathcal{T}}(x^{adv})$ culminate in a thorough structural breakdown of the perturbations $\varepsilon\mathcal{P}$ in the sparse domain. This comprehensive disintegration plays a vital role in diminishing their intended impact, consequently enhancing the resilience of the original video x against these adversarial attacks.

To realize a sparse transformation $\mathcal{T}$, traditional methods like dictionary learning [98] are useful but may lack dynamic adaptability [14, 88] and hierarchical feature extraction. Utilizing CNNs' deep layered structure enables extracting hierarchical features from video data, resulting in a more adaptive and detailed sparse representation [14]. Unlike static dictionaries [87], CNNs are trainable, retrainable, and fine-tunable, allowing them to evolve with the data.

In SECVID, a CNN-based sparse transformation is applied:

$$\mathcal{S} = \mathcal{T}_{\text{CNN}}(x^{adv}) \tag{10}$$

where $\mathcal{T}_{\text{CNN}}$ is a CNN-powered sparse transformation function, and $\mathcal{S}$ is its resultant latent sparse representation. The transformation's sparsity degree is key to diminishing adversarial perturbations. As x^{adv} becomes sparser post-transformation, the adversarial perturbations' dispersal effect intensifies, increasing their effective elimination. To sparsify $\mathcal{S}$, we utilize a sparse transformation loss (Equation (18)), detailed in Section 5.3, and *Degree of Sparsity (DoS)* as a critical metric to guide the sparse transformation in SECVID:

$$DoS = C_{\text{SECVID}}/C \tag{11}$$

where C and C_{SECVID} represent the channel counts before and after the transformation, respectively.

5.2 Discretized Compression

Adversarial video attacks generate pixel-level perturbations within a set magnitude ε, as detailed in Equation (6) [54]. These perturbations, stealthy and hard to detect due to their continuous and bounded nature [18, 29, 47, 54], are shown in Figure 2. It includes (a) an original video frame, (b) the frame with subtle alterations, and (c) the smooth alteration gradient, showing a strategy that exploits model vulnerabilities.

We exploit the inherent continuity of adversarial perturbations—often their Achilles' heel—by employing *discretized compression*. This process involves discretization, which transforms the data from a smooth continuum into a distinct, jagged discrete space, and compression, which further compacts the data, effectively neutralizing perturbations.

We employ the K-Means clustering algorithm [17, 19] for our discretized compression component, chosen for its efficiency in discretizing continuous data and its lightweight characteristics. This method quantizes the continuous sparse representation by assigning each data point to the nearest cluster centroid. We refer to this *discretized compressor* as $\mathcal{Q}$, which contains K cluster centroids for the dataset. The quantization process is mathematically modeled as follows:

$$\mathcal{Y} = \mathcal{Q}(\mathcal{S}) \text{ subject to } |\mathcal{Y}| = |\mathcal{Q}| = K \tag{12}$$

where $\mathcal{Y}$ signifies the discretized and compressed measurements from $\mathcal{S}$ as shown in Equation (10). K-Means is particularly effective here for its minimal computational overhead and its disruption of adversarial perturbation continuity.

Our discretized compression strategy interrupts pixel-level continuity, as illustrated in Figure 1. Through sparse transformation, a video frame is transformed into sparse coefficients, capturing essential pixel-level details like high-level features for sharp transitions and smoother, uniform areas. For example, a slice from the frame's 3D sparse representation, displayed at the bottom-left corner, includes vectors A, B, C, and D, each containing zeros and non-zeros. With K-means for discretized compression, all values within a vector are replaced by the nearest cluster centroid's value (for this particular illustrating example). Notably, vectors A and C are in the same cluster. This discretized compression strategy disrupts the smooth flow of data, leading to abrupt transitions that significantly impact pixel-level continuity.

The size, i.e., *cluster count* of $\mathcal{Q}$, i.e., K, is crucial in discretized compression. A smaller $\mathcal{Q}$ means more data discretization and compression, aiding in perturbation defense by intensifying data discretization and eliminating more perturbations. However, too much discretization can lower video recovery quality by filtering out vital original video features. In SECVID, K is adjustable, letting users choose values based on their needs. The effects of different K values on performance are detailed in Section 6. Empirically, for the evaluated video recognition systems, $K = 1024$ is a balanced trade-off.

5.3 Video Reconstruction

Following sparse transformation and discretized compression, the main challenge is to restore the original video quality and counter adversarial perturbations. The reconstruction component, $\mathcal{R}_{\text{CNN}}$, seeks to reconstruct x from $\mathcal{Y}$:

$$x = \mathcal{R}_{\text{CNN}}(\mathcal{Y}) \tag{13}$$

This is achieved with a composite loss function combining four crucial components: *content loss*, based on VGG-19 [74] features, *temporal loss* leveraging optical flow [100], *perceptual loss* for pixel-level detail [37], and *sparse transformation loss* to minimize $\mathcal{S}$ in line with VCS theory (Equation (1)).

Therefore, the total loss function L_{loss} is given by:

$$L_{\text{loss}} = \alpha L_{\text{cont}} + \beta L_{\text{temp}} + \gamma L_{\text{per}} + \delta L_{\mathcal{S}} \tag{14}$$

where α, β, γ, and δ are weight coefficients turned to balance the four losses for a harmonized reconstruction.

For optimal SecVID performance, we co-train $\mathcal{T}_{\text{CNN}}$ and $\mathcal{R}_{\text{CNN}}$, as detailed in Algorithm 2. Using the notations from Section 3, for a video $x \in \mathbb{R}^{T\times H\times W\times C}$, x_t denotes its t-th frame, and x'_t refers to its reconstructed version. $\mathcal{S} \in \mathbb{R}^{T\times H\times W\times C_{\text{SecVID}}}$ represents the sparse version of x.

Content Loss. We select VGG-19 [74] for content extraction due to its depth and proficiency in capturing complex content hierarchies [64, 82], particularly through its deep layers, even with subtle perturbations:

$$L_{\text{cont}}(x_t, x'_t) = \sum_{t=1}^{T}\sum_{l}\frac{1}{H_lW_lC_l}\left\|\phi_l^{\text{VGG-19}}(x_t) - \phi_l^{\text{VGG-19}}(x'_t)\right\|_2^2 \tag{15}$$

where $\phi_l^{\text{VGG-19}}$ denotes the feature map of the l-th layer in VGG-19, while W_l, H_l, and C_l denote the height, width and number of channels of that layer, respectively.

Temporal Loss. Temporal coherence, essential for videos [33, 81], demands natural frame transitions. We use SpyNet [67] for efficient optical flow extraction. As a pre-processing module, SecVID prioritizes speed with minimal accuracy loss. SpyNet's rapid computation meets our needs for swift, accurate temporal checks:

$$L_{\text{temp}}(x'_t, x'_{t+1}) = \sum_{i=1}^{T-1}\frac{1}{HWC}\left\|x'_{t+1} - \text{warp}(x'_t, \Omega^{\text{SpyNet}}(x'_t, x'_{t+1}))\right\|_2^2 \tag{16}$$

where the warp function from [91] simulates pixel motion in x'_t using the optical flow between x'_t and x'_{t+1}, denoted $\Omega^{\text{SpyNet}}(x'_t, x'_{t+1})$, to ensure temporal coherence.

Perceptual Loss. This metric goes beyond mere basic pixel-level accuracy, capturing essential stylistic elements like textures and colors in the video, thereby ensuring a high-fidelity representation of its intrinsic qualities [65]:

$$L_{\text{per}}(x_t, x'_t) = \sum_{t=1}^{T}\frac{1}{HWC}\left\|x_t - x'_t\right\|_2^2 \tag{17}$$

Sparse Transformation Loss. In compressive sensing [98], the goal is to efficiently compress original data while preserving key features. The sparse transformation loss $L_{\mathcal{S}}$, crucial for minimizing the sparse representation as shown in Equation (1), aims to retain only significant data features:

$$L_{\mathcal{S}} = \sum_{t=1}^{T}\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{k=1}^{C_{\text{SecVID}}}\left\|\mathcal{S}_{tijk}\right\|_1 \tag{18}$$

which represents the sum of $\mathcal{S}$'s absolute values, as per the ℓ_1-norm used. A lower $L_{\mathcal{S}}$ indicates a more sparse and significant representation, essential for effective adversarial resisting and effective data recovery in compressive sensing [98].

5.4 SecVID: A Pre-Processing Module

In Algorithm 1, the integration of SecVID as a pre-processing defense module for a video classifier is demonstrated. The co-training of $\mathcal{T}_{\text{CNN}}$ and $\mathcal{R}_{\text{CNN}}$ (Equations (10) and (13)) and the development of $\mathcal{Q}$ from $\mathcal{T}_{\text{CNN}}$ (Equation (12)) are specified in Algorithm 2. In both algorithms, $\oplus$ represents an operator that combines video frames together in the standard manner. The training, described in Section 6.1, involves system parameters like *DoS* (Equation (11)) and the cluster count K of $\mathcal{Q}$ (Equation (12)), and hyper-parameters such as number of epochs N, and loss weights α, β, γ, and δ (Equation (14)). SecVID ensures temporal coherence between consecutive frames using L_{temp} in line 7 of Algorithm 2 and operates on a frame-by-frame basis in Algorithm 1.

6 Evaluation

We show that SecVID, as a black-box pre-processing module, outperforms seven leading baselines: two video-specific defenses—AdvIT [89] and OUDefend [51], and five image-centric frameworks—AT (Adversarial Training) [27, 42, 71, 78], IT (Input Transformations) [28], RS (Random Smoothing) [34], ComDefend [37], and DiffPure [60]. Additionally, SecVID effectively counters sparse adversarial attacks [4] specifically targeting it. Our evaluation addresses four RQs:

RQ1. Does SecVID outperform the seven advanced baselines in defending against adversarial videos?

RQ2. Does SecVID maintain video quality and fidelity through its discretization-enhanced video compressive sensing, thereby enhancing its defense capabilities?

RQ3. Are the costs of SecVID's security manageable?

RQ4. How robust is SecVID against sparse attacks?

6.1 Experiment Setup

Video Recognition Systems. We evaluate two prominent classifiers, C3D [79] and I3D [8]. Both models, employing distinct strategies, deliver top-tier video classification [5, 90]. C3D uses 3D convolution for spatial-temporal feature learning, while I3D employs optical flow for frame relationships. Given that C3D requires 16-frame input videos, we divide all video samples into segments of 16-frame each.

Datasets. We utilize UCF-101 [75] and HMDB-51 [43], two prominent datasets for video classifiers. UCF-101, from

```
Algorithm 1: SecVID defense framework.
Input: Adversarial video sample $x^{adv}$, $\mathcal{T}_{\text{CNN}}$, $\mathcal{Q}$, $\mathcal{R}_{\text{CNN}}$
       from Algorithm 2, and a black-box video
       classifier $f$.
Output: Recovered video $x$.
$\mathcal{S} \leftarrow \varnothing$; $\mathcal{Y} \leftarrow \varnothing$; $x \leftarrow \varnothing$;
foreach frame $x_t^{adv}$ in $x_{adv}$ do
    $\mathcal{S}_t \leftarrow \mathcal{T}_{\text{CNN}}(x_t^{adv})$;
    $\mathcal{S} \leftarrow \mathcal{S} \oplus \mathcal{S}_t$;
foreach sparse representation $\mathcal{S}_t$ in $\mathcal{S}$ do
    $\mathcal{Y}_i \leftarrow \mathcal{Q}(\mathcal{S}_t)$;
    $\mathcal{Y} \leftarrow \mathcal{Y} \oplus \mathcal{Y}_t$;
foreach discretized measurement $\mathcal{Y}_t$ in $\mathcal{Y}$ do
    $x_t \leftarrow \mathcal{R}_{\text{CNN}}(\mathcal{Y}_t)$;
    $x \leftarrow x \oplus x_t$;
Perform inference on downstream $f(x)$;
```

YouTube, contains 13,320 videos in 101 action classes, including archery and haircut. HMDB-51, from varied sources, comprises 6,849 videos in 51 classes like sword and climb.

Adversarial Attacks. We examine a wide range of advanced attacks, including both universal and one-on-one types across different norms and perturbation intensities, in targeted and untargeted scenarios:

- *Perturbation Norms.* Perturbations are often characterized by their compliance with specific ℓ_p-norms. However, some, like StyleFool [5], employ unrestricted perturbations driven by video styles. Our evaluation encompasses both ℓ_∞-norm bounded and StyleFool-type unrestricted perturbations.
- *Procedural Noises.* Adversarial attacks often employ strategic noises such as U3D [90] using *Perlin* [62] or *Gabor* [44] to craft perturbations. StyleFool [5] utilizes style transfer as the basis for its attacks, subsequently refining them with smoothly generated PGD noise during execution.
- *Perturbation Intensity* ε. In Equation (5), ε represents the intensity of adversarial alterations in videos. Within digital media's 8-bit range, a higher ε means stronger perturbations, but attackers often opt for subtler, less noticeable changes. In our main setup, following U3D attack methodology [90], ε is set to 8/255. SecVID's consistent performance across various intensities is briefly discussed in Section 7.
- *Targeted vs. Untargeted Attacks.* Adversarial attacks can be targeted [5, 48], aiming to deceive the model into predicting a specific, incorrect class, or untargeted [49, 90], seeking to cause any misclassification without a specified target.
- *Universal vs. One-on-One Attacks.* Universal attacks like C-DUP [49] and U3D [90] cause broad disruptions effective across multiple videos, while one-on-one attacks such as StyleFool [5] and Geo-Trap [48] deliver precise, targeted or untargeted effects.

```
Algorithm 2: SecVID Training.
Input: Training dataset $\mathcal{D}$, $DoS$, cluster count $K$,
       Number of epochs $N$, and the four loss weights
       $\alpha$, $\beta$, $\gamma$, and $\delta$ from Equation (14).
Output: Trained parameters $\Theta_{\mathcal{T}}$ and $\Theta_{\mathcal{R}}$ as well as $\mathcal{Q}$.
Initialize $\Theta_{\mathcal{T}}$ for $\mathcal{T}_{\text{CNN}}$ with $DoS$, $\Theta_{\mathcal{R}}$ for $\mathcal{R}_{\text{CNN}}$,
  and $\mathcal{Q}$;
for $epoch = 1$ to $N$ do
    foreach video $x$ in $\mathcal{D}$ do
        foreach frame $x_t$ in $x$ do
            $\mathcal{S}_t \leftarrow \mathcal{T}_{\text{CNN}}(x_t)$;
            $x'_t \leftarrow \mathcal{R}_{\text{CNN}}(\mathcal{S}_t)$;
            $L_{\text{loss}} \leftarrow \alpha L_{\text{cont}}(x_t, x'_t) + \beta L_{\text{temp}}(x'_t, x'_{t+1}) +$
              $\gamma L_{\text{per}}(x_t, x'_t) + \delta L_{\mathcal{S}}$;
            $\Theta_{\mathcal{T}}, \Theta_{\mathcal{R}} \leftarrow$ Minimize $L_{\text{loss}}$;
foreach video $x$ in $\mathcal{D}$ do
    foreach frame $x_t$ in $x$ do
        $\mathcal{S}_t \leftarrow \mathcal{T}_{\text{CNN}}(x_t)$;
        $\mathcal{S} \leftarrow \mathcal{S} \oplus \mathcal{S}_t$;
$\mathcal{Q} \leftarrow$ $K$-Means($\mathcal{S}$);
```

In our evaluation, we consider a variety of state-of-the-art attack strategies, each posing unique challenges:

- *U3D [90].* This prominent universal attack framework employs Perlin or Gabor noise under ℓ_∞-norm. Notably stealthy, U3D makes its perturbations hard to detect, attaining an 87.8% success rate against diverse video classifiers.
- *StyleFool [5].* The StyleFool framework initiates its adversarial strategy with a style transfer, followed by using PGD to optimize and create unrestricted perturbations for both targeted and untargeted attacks. It consistently achieves a 100% success rate, underscoring its effectiveness.
- *Geo-Trap [48].* A black-box framework for both untargeted and targeted attacks optimizes video gradients using geometric transformations within the ℓ_∞-norm.

Using open-source frameworks, we follow their protocols to generate adversarial video samples. We introduce 3,911 samples, as detailed in Table 2, covering five attack types across both UCF-101 and HMDB-51, targeting C3D and I3D. To evaluate SecVID's defense, we ensure that both classifiers accurately identified original videos but failed with their adversarial counterparts. In terms of generation speed, U3D is the quickest, while Geo-Trap takes the longest. Sample creation takes 1 week for StyleFool (U), 3 weeks for StyleFool (T), 1 week each for Geo-Trap (U) and (T), and 6 hours for U3D (U), aligning with studies like [5, 90] that use around 50 samples per scenario. Except for C3D-HMDB51, all 10 model-dataset pairs are documented [5, 48, 90]. Notably, C3D-HMDB51 has the fewest samples for StyleFool (U), reflecting the challenge of generating adversarial samples in this case.

These five attack types are all black-box attacks. For certain

Table 2: Adversarial videos (**U**ntargeted and **T**argeted).

Attack Method	C3D		I3D	
	UCF-101	HMDB-51	UCF101	HMDB-51
StyleFool (U)	109	32	144	206
U3D (U)	602	602	301	301
Geo-Trap (U)	76	53	97	63
StyleFool (T)	573	143	143	198
Geo-Trap (T)	84	51	74	59

white-box attacks [30, 49, 63], their implementations are not open-sourced, limiting our ability to use them. As detailed in Section 5, SECVID is capable of defending against white-box attacks similarly to black-box attacks, provided there are no model modifications (Section 3). This is possible because SECVID corrects adversarial videos for a video classifier by leveraging discretization-enhanced VCS [14, 85, 92, 94], needing only access to the classifier's original training dataset.

In addition, we also assess SECVID's robustness against sparse attacks [4] targeting its sparse transformation.

Baselines. We evaluate SECVID's performance against seven baselines: AdvIT [89], AT [27, 42, 71, 78], IT [28], RS [34], OUDefend [51], ComDefend [37] and DiffPure [60].

AdvIT [89] and OUDefend [51] are two defenses for video. AdvIT uses black-box temporal consistency, estimating optimal flow between a target frame and its m preceding frames, creating m pseudo frames, and computing an inconsistency score c to detect adversarial content. Gaussian noise enhances detection, with $m = 3$ set due to minimal performance variation [89, 90]. SpyNet [67] is used for flow estimation. A video is flagged as adversarial if the averaged inconsistency score over five frames exceeds 1, ensuring consistent detection across various frame counts [89]. OUDefend [51] utilizes over/undercomplete features in a restoration network to combat adversarial videos, per its established implementation.

The five other baselines—AT [27, 42, 71, 78], IT [28], RS [34], ComDefend [37], and DiffPure [60]—are leading image-based defense schemes adapted for videos. AT, enhancing model robustness, integrates adversarial samples into training; we implemented it as outlined in [71], and took 16-frame as input for each video, following [5]. IT, which applies manipulations like cropping and rotating to reduce adversarial effects on frames, was implemented based on its outdated open-source code [28]. RS, which uses Gaussian noise to enhance model resilience within the ℓ_2-norm space, we use its open-source version [34]. ComDefend employs compression and reconstruction CNNs for image reconstruction; we used its open-source implementation [36] for frame-by-frame video processing. DiffPure uses diffusion models for noise reduction to counter adversarial perturbations, also using its open-source implementation [59].

Computing Platform. We conduct all experiments, including training and inferencing, on four NVIDIA RTX 4090 cards with 24GB RAM each, utilizing CUDA version 12.0.

SECVID Training Protocols. SECVID, composed of sparse transformation $\mathcal{T}_{\text{CNN}}$, discretized compression $\mathcal{Q}$, and reconstruction $\mathcal{R}_{\text{CNN}}$, is trained using the video classifier's dataset without accessing the classifier itself. Both $\mathcal{T}_{\text{CNN}}$ and $\mathcal{R}_{\text{CNN}}$ utilize ten 2D convolution layers with 3×3 filters, each followed by LeakyReLU activation. Training uses the Adam optimizer with an initial learning rate of 0.001 and is set for 100 epochs. Default values for $\mathcal{T}_{\text{CNN}}$ are $DoS = 4$ and $K = 1024$. In $\mathcal{R}_{\text{CNN}}$, the weights in Equation (14) are $\alpha = 10^4$, $\beta = 10^4$, $\gamma = 1$, and $\delta = 1$. Grid search determines the optimal weights, selecting the combination for best performance. Co-training of $\mathcal{T}_{\text{CNN}}$ and $\mathcal{R}_{\text{CNN}}$ typically converges in 30 epochs for UCF-101 and 60 for HMDB-51. For content loss L_{cont}, we use VGG-19's `ReLU4_2` layer output, following [5].

6.2 RQ1. SECVID's Defense Capabilities

SECVID significantly outperforms contemporary defense baselines, maintaining a low false positive rate. In Section 6.2.1, we show that SECVID, with its default setting of $DoS = 4$ and $K = 1024$, surpasses AdvIT [89], AT [27, 42, 71, 78], IT [28], RS [34], OUDefend [51], ComDefend [37], and DiffPure [60]. We investigate how modifying the sparse transformation (Equation (10)) affects SECVID's performance, including varying DoS levels (Equation (11)) and enabling or disabling the sparse transformation loss (Equation (18)). We also analyze VCS's effects on three key properties: SC, IR, and PR (Definition 1). Additionally, we consider the influence of varying cluster counts in discretized compression (Equation (12)) and including or excluding the temporal loss function (Equation (16)). The impacts of these adjustments are analyzed in Section 6.2.2 – Section 6.2.4.

To evaluate defense mechanisms, we use the Detection Success Rate (DSR)—the ratio of correctly detected adversarial videos to the total adversarial samples. AdvIT uniquely focuses on detection, calculating DSR as the ratio of correctly detected adversarial videos to total adversarial samples. For SECVID, as with the other six baselines, DSR is the ratio of correctly classified adversarial videos to total samples.

6.2.1 Defense Performance

Table 3 highlights SECVID's improved DSRs compared to AdvIT, AT, IT, RS, OUDefend, ComDefend, and DiffPure for C3D and I3D using UCF-101 and HMDB-51 across five attack types. For C3D, SECVID's DSR improvements on UCF-101 are 399.2%, 53.5%, 315.8%, 478.6%, 88.3%, 62.2%, and 38.5% over AdvIT, AT, IT, RS, OUDefend, ComDefend, and DiffPure, respectively, and 471.0%, 77.3%, 286.8%, 558.3%, 265.0%, 136.5%, and 50.1% on HMDB-51. For I3D, the improvements on UCF-101 are 523.5%, 55.4%, 866.2%, 521.0%, 145.8%, 88.5%, and 51.3%, and 433.6%, 60.8%, 716.9%, 592.4%, 273.4%, 90.0%, and 46.3% on HMDB-51. Relative to AT and DiffPure, the two best-performing base-

Table 3: Comparing SECVID with AdvIT, AT, IT, RS, OUDefend, ComDefend, and DiffPure for their **DSRs (Detection Success Rates)** on adversarial videos, as shown in Table 2. The highest DSR for each attack type and model is highlighted in **bold**.

Model	Attack Method	AdvIT [89]		AT [71]		IT [28]		RS [34]		OUDefend [51]		ComDefend [37]		DiffPure [60]		SECVID	
		UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	StyleFool (U)	31.2%	25.0%	62.4%	53.1%	8.3%	25.0%	13.8%	6.3%	37.6%	25.0%	42.2%	18.8%	65.1%	59.4%	**83.5%**	**84.3%**
	U3D (U)	12.0%	15.9%	47.3%	38.2%	50.0%	12.5%	16.1%	30.2%	57.1%	32.4%	70.9%	46.0%	71.3%	66.4%	**92.3%**	**83.3%**
	Geo-Trap (U)	19.7%	10.5%	64.5%	58.5%	25.0%	0.0%	23.2%	32.1%	48.7%	22.6%	65.8%	56.6%	63.2%	66.0%	**100.0%**	**100.0%**
	StyleFool (T)	17.5%	15.4%	53.2%	46.9%	20.4%	11.2%	11.3%	8.4%	43.6%	29.4%	54.1%	45.4%	68.6%	53.8%	**79.2%**	**81.8%**
	Geo-Trap (T)	16.7%	19.6%	61.9%	60.8%	0.0%	0.0%	14.3%	25.5%	48.8%	19.6%	63.1%	51.0%	58.3%	64.7%	**83.3%**	**100.0%**
I3D	StyleFool (U)	13.7%	18.6%	57.0%	59.2%	16.7%	11.1%	9.0%	7.3%	43.1%	18.4%	75.0%	63.1%	60.4%	55.8%	**88.9%**	**83.0%**
	U3D (U)	12.3%	9.6%	49.8%	41.2%	20.2%	26.2%	25.2%	29.2%	34.9%	28.6%	18.3%	22.6%	68.1%	67.4%	**83.1%**	**94.4%**
	Geo-Trap (U)	19.7%	9.2%	64.9%	61.9%	38.1%	0.0%	21.6%	27.0%	28.9%	23.8%	82.5%	63.5%	53.6%	57.1%	**91.8%**	**88.8%**
	StyleFool (T)	15.5%	13.9%	57.3%	53.8%	2.8%	3.0%	10.5%	6.3%	39.2%	13.6%	38.5%	43.9%	58.7%	65.7%	**96.5%**	**83.3%**
	Geo-Trap (T)	11.9%	9.8%	58.1%	61.0%	41.9%	0.0%	18.9%	23.7%	24.3%	20.3%	85.1%	33.9%	62.2%	67.8%	**83.8%**	**83.1%**

Table 4: Comparing SECVID with AdvIT, AT, IT, RS, OUDefend, ComDefend, and DiffPure in **managing clean videos** from UCF-101 and HMDB-51, using false positive rate for AdvIT (detection-only), and accuracy for the others (protection-oriented).

Model	False Positive Rates		Recognition Accuracy															
	AdvIT [89]		Unprotected		AT [71]		IT [28]		RS [34]		OUDefend [51]		ComDefend [37]		DiffPure [60]		SECVID	
	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	3.2%	4.1%	78.3%	60.2%	76.2%	59.2%	49.7%	38.2%	65.6%	51.2%	59.7%	48.4%	69.3%	55.7%	71.5%	52.3%	73.7%	56.9%
I3D	3.2%	4.1%	87.6%	62.5%	87.3%	62.5%	58.4%	42.3%	53.8%	54.3%	55.3%	42.9%	78.6%	56.3%	82.9%	56.4%	85.2%	60.3%

Table 5: Average inference time (ms) per video of classifiers protected by SECVID, AdvIT, OUDefend, ComDefend, DiffPure, and "Unprotected" (representing AT, IT, and RS), with 300 clean videos randomly selected from each of UCF-101 and HMDB-51.

Model	Unprotected		AdvIT [89]		OUDefend [51]		ComDefend [37]		DiffPure [60]		SECVID	
	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	6.21	6.99	434.17**(69.91×)**	436.29**(62.42×)**	11.93**(1.92×)**	18.76**(2.68×)**	12.50**(2.01×)**	23.69**(3.39×)**	458.40**(73.82×)**	446.24**(63.86×)**	13.72**(2.20×)**	30.60**(4.40×)**
I3D	10.75	8.40	438.71**(40.81×)**	437.70**(52.11×)**	19.58**(1.82×)**	24.63**(2.93×)**	17.04**(1.59×)**	25.10**(2.99×)**	462.94**(43.06×)**	447.65**(53.29×)**	18.34**(1.71×)**	34.12**(4.06×)**

lines, SECVID's average gains across all model-dataset-attack scenarios are 61.8% and 46.6%, respectively. IT's 0.0% DSR against certain attacks, due to undetected adversarial videos, is excluded from SECVID's accuracy improvements over IT.

SECVID excels in defending against both targeted and untargeted attacks. For Geo-Trap (U), it secures a 100% DSR with C3D on both UCF-101 and HMDB-51 datasets. With I3D, SECVID achieves 91.8% DSR on UCF-101 and 88.8% on HMDB-51, outperforming all three baseline defenses. In the case of StyleFool (T), SECVID attains 79.2% DSR for UCF-101 and 81.8% for HMDB-51 with C3D, surpassing the baselines, and shows even stronger performance with I3D.

AdvIT demonstrates low DSRs across all five attack types, consistent with earlier findings [5,90], highlighting the limitations of relying solely on temporal consistency, especially in temporally coherent attacks. Similarly, OUDefend performs poorly, as its use of over/undercomplete features in a restoration network fails to address subtle perturbations without disrupting overall completeness. Among the five image-centric baselines, AT, ComDefend, and DiffPure show relatively high DSRs, with DiffPure only slightly outperforming AT. However, AT requires costly model retraining with known adversarial samples, ComDefend neglects temporal consistency, and DiffPure incurs exceptionally high computational costs (Table 5). Conversely, RS [34] avoids these issues by smoothing predictions but ranks lowest in effectiveness.

In contrast, SECVID uses discretization-enhanced VCS theory to counter these attacks, disrupting perturbations while preserving temporal coherence (Equation (16)).

SECVID excels not only in defense against adversarial videos, surpassing the seven baselines, but also handles clean videos more effectively overall, as shown in Table 4. The values in Table 3 are higher because they include adversarial samples from Table 2, where classifiers correctly identify clean videos but fail on adversarial ones, maintaining 100% accuracy on clean videos. In contrast, Table 4 notes potential misclassification for clean videos under "Unprotected".

AdvIT, with its sole focus on detection, has its precision gauged using the false positive rate (FPR). This metric reflects the percentage of clean videos that a classifier can accurately classify but are erroneously flagged as adversarial by AdvIT. On the other hand, SECVID, along with the other six baselines, add a defense layer to a downstream classifier, have their effectiveness assessed by the recognition accuracy of the protected classifier. Thus, the FPR for a specific dataset can be interpreted as the product of the discrepancy in accuracy between an unprotected and its protected counterpart, and the accuracy of the unprotected classifier itself.

SECVID significantly outperforms IT, RS, OUDefend, ComDefend, and DiffPure in terms of accuracy but is marginally less accurate than AT, attributed to AT's comprehensive adversarial training that generally maintains the

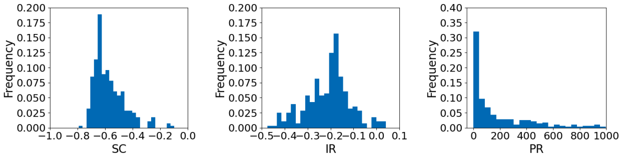


Figure 3: Effects of VCS on SC, IR, and PR for adversarial samples for StyleFool (U) (Table 2) on UCF-101 with C3D.

accuracy of its protected classifiers. In comparison with AdvIT, SecVID shows slightly better performance in all four settings, except for C3D on UCF-101. This is evident considering SecVID's FPRs are 3.6% for C3D and 2.1% for I3D on UCF-101, and 2.0% for C3D and 1.4% for I3D on HMDB-51, respectively. Note that AdvIT's detection rates are not dependent on the video classifiers. In Section 6.4, we will delve deeper into the moderate accuracy loss experienced when using SecVID, as compared to its absence.

In Table 5, we reported the average inference times for video classifiers protected by SecVID and seven baselines, using 300 clean videos from each dataset. For AT, IT, and RS, the post-integration times are similar to those of the unprotected classifier, used as the "Unprotected" reference point. SecVID significantly outperforms AdvIT and DiffPure, with AdvIT incurring high costs due to multiple pseudo frames and inference operations, and DiffPure's iterative frame purification demanding significant resources [60]. Although SecVID has slightly higher computational demands than OUDefend and ComDefend, it achieves superior DSRs.

These findings show SecVID's superior effectiveness against advanced attacks, surpassing current solutions. While ideal for urban safety, anomaly detection, and online videos, SecVID is less suited for real-time applications like autonomous driving or patient monitoring. Targeted optimizations can enhance its efficiency in these contexts (Section 7).

6.2.2 Impact of Sparse Transformation

Sparse transformation in VCS [14, 85, 92, 94] modifies the sparsity, intensity, and positions of non-zero elements in videos. We assess VCS's impact on SC (Equation (7) with $\tau = 0.05$), IR (Equation (8)), and PR (Equation (9)) using five types of adversarial samples from UCF-101 against C3D (Table 2). For StyleFool (U), shown in Figure 3, we consistently observe a decrease in SC and IR, and an increase in PR. Minimum, maximum, and average values are −0.791, −0.098, and −0.572 for SC (a decrease in 100% of adversarial samples), −0.476, 0.04, and −0.210 for IR (a decrease in 97.9%), and 0.972, 994.422, and 184.581 for PR (an increase in 100%). Similar trends are seen for the other four attack types: StyleFool (T), U3D, Geo-Trap (T), and Geo-Trap (U).

Impact of DoS. Table 6 shows how changing *DoS* from 1 to 4 (Equation (11)), while keeping $K = 1024$ constant, affects SecVID's effectiveness. Increasing the *DoS* generally enhances its adversarial resilience. For C3D, SecVID achieves its highest or equal highest DSR at $DoS = 4$ for all five attack types, marking significant defense improvements. For example, with StyleFool (U) on UCF-101, SecVID's DSR rises from 69.7% at $DoS = 1$ to 83.5% at $DoS = 4$. A similar increase is noted for Geo-Trap (U) on HMDB-51, where DSR jumps from 75.0% at $DoS = 1$ to a perfect 100% at $DoS = 4$.

Table 6: Impact of ***DoS*** on SecVID's DSR, evaluated with four ***DoS*** **levels**, at $K = 1024$ on adversarial videos (Table 2).

Model	Attack Method	$DoS = 1$		$DoS = 2$		$DoS = 3$		$DoS = 4$	
		UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	StyleFool (U)	69.7%	75.0%	72.5%	71.9%	74.3%	75.0%	**83.5%**	**84.3%**
	U3D (U)	74.9%	50.0%	84.6%	66.7%	88.5%	68.8%	**92.3%**	**83.3%**
	Geo-Trap (U)	75.0%	**100.0%**	87.5%	**100.0%**	82.9%	**100.0%**	**100.0%**	**100.0%**
	StyleFool (T)	75.2%	73.4%	73.8%	74.8%	76.6%	78.3%	**79.2%**	**81.8%**
	Geo-Trap (T)	70.0%	**100.0%**	71.1%	**100.0%**	77.8%	**100.0%**	**83.3%**	**100.0%**
I3D	StyleFool (U)	87.5%	68.9%	86.1%	**83.5%**	**89.0%**	**83.5%**	88.9%	83.0%
	U3D (U)	69.8%	86.4%	81.7%	83.1%	78.1%	**89.4%**	83.1%	**94.4%**
	Geo-Trap (U)	87.6%	66.7%	89.7%	66.7%	89.7%	66.7%	**91.8%**	**88.8%**
	StyleFool (T)	**100.0%**	78.8%	88.1%	78.3%	90.9%	80.3%	96.5%	**83.3%**
	Geo-Trap (T)	71.6%	66.1%	81.1%	66.1%	**83.8%**	66.1%	**83.8%**	**83.1%**

Table 7: Impact of sparse transformation loss L_S (Equation (18)) on SecVID's DSR in the default setting of $DoS = 4$ and $K = 1024$ on adversarial videos, as shown in Table 2.

Model	Attack Method	Without L_S		With L_S	
		UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	StyleFool (U)	74.3%	81.3%	83.5%	84.3%
	U3D (U)	84.9%	81.1%	92.3%	83.3%
	Geo-Trap (U)	81.6%	86.8%	100.0%	100.0%
	StyleFool (T)	75.0%	78.3%	79.2%	81.8%
	Geo-Trap (T)	78.6%	86.3%	83.3%	100.0%
I3D	StyleFool (U)	75.0%	68.4%	88.9%	100.0%
	U3D (U)	81.4%	84.4%	83.1%	94.4%
	Geo-Trap (U)	86.7%	81.0%	91.8%	88.8%
	StyleFool (T)	93.7%	79.8%	96.5%	83.3%
	Geo-Trap (T)	78.4%	76.3%	83.8%	83.1%

For I3D, SecVID's response to different *DoS* levels is more nuanced compared to C3D, but it follows a similar overall trend. It achieves optimal performance for Geo-Trap (U) at $DoS = 4$ on both UCF-101 and HMDB-51. However, for StyleFool (T) on UCF-101, its best performance is seen at the lowest $DoS = 1$. This variation is influenced by I3D's focus on long-range temporal flow, along with spatial flow, enhancing C3D. In I3D, smaller *DoS* levels are more effective in mitigating spatial adversarial perturbations, while larger ones better disrupt temporal disruptions. This results in a more spread impact of *DoS* choices in I3D compared to C3D.

The findings show that *DoS* considerably affects SecVID's defense efficacy. In general, a larger *DoS* enhances data sparsity, improving the disruption of adversarial perturbations.

Impact of Sparse Transformation Loss L_S. Table 7 highlights the differences in SecVID's defense efficacy with and without the utilization of L_S, under its default setting of $DoS = 4$ and $K = 1024$. SecVID exhibits improved performance for both classifiers across two datasets against five types of attacks. This improvement underscores the importance of L_S in reducing sparse representation and enhancing its sparsity, as detailed in Sections 5.1 and 5.3.

Table 8: Impact of varying **cluster counts** on SecVID in terms of DSR, evaluated across four distinct cluster counts, at $DoS = 4$ on adversarial videos, as shown in Table 2. The highest DSR for each attack type and model is highlighted in **bold**.

Model	Attack Method	Without Discretized Compression		With Discretized Compression							
				$K = 128$		$K = 256$		$K = 512$		$K = 1024$	
		UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	StyleFool (U)	63.3%	40.0%	79.8%	78.1%	82.5%	75.0%	80.0%	81.3%	**83.5%**	**84.3%**
	U3D (U)	50.0%	54.8%	83.3%	85.5%	83.3%	84.7%	91.6%	**86.7%**	**92.3%**	83.3%
	Geo-Trap (U)	87.5%	95.5%	75.0%	**100.0%**	75.0%	**100.0%**	87.5%	**100.0%**	**100.0%**	**100.0%**
	StyleFool (T)	64.8%	61.2%	**79.2%**	73.3%	**79.2%**	75.2%	**79.2%**	76.1%	**79.2%**	**81.8%**
	Geo-Trap (T)	78.0%	66.7%	73.7%	**100.0%**	73.7%	**100.0%**	**83.8%**	66.7%	83.3%	**100.0%**
I3D	StyleFool (U)	85.4%	63.1%	86.1%	61.5%	**90.3%**	66.7%	88.9%	67.4%	88.9%	**83.0%**
	U3D (U)	59.8%	89.5%	69.4%	72.2%	**83.3%**	80.8%	**83.3%**	83.3%	83.1%	**94.4%**
	Geo-Trap (U)	83.8%	66.7%	**93.5%**	66.7%	**93.5%**	74.2%	90.3%	78.5%	91.8%	**88.8%**
	StyleFool (T)	84.7%	73.5%	94.4%	79.5%	94.4%	79.5%	94.4%	81.8%	**96.5%**	**83.3%**
	Geo-Trap (T)	78.4%	66.1%	73.0%	62.1%	73.0%	65.8%	**83.8%**	66.1%	**83.8%**	**83.1%**

6.2.3 Impact of Discretized Compression

Table 8 shows the impact of different cluster counts ($K \in \{128, 256, 512, 1024\}$) on SecVID's defense effectiveness, with $DoS = 4$ as the default setting. Generally, SecVID's performance decreases without discretized compression.

For C3D, under the StyleFool (U) attack, SecVID shows a significant DSR improvement with more clusters. DSR increases from 63.3% without discretization to 83.5% at $K = 1024$ on UCF-101. On HMDB-51, it rises from 40.0% to 84.3% at the same cluster count. This pattern is observed in other attacks as well; with U3D (U) on UCF-101, SecVID's DSR climbs from 50.0% to 92.3% at $K = 1024$, affirming that higher cluster counts consistently boost adversarial resilience.

For I3D, SecVID shows a similar trend, with its DSR increasing from 83.8% without discretization to 91.8% at $K = 1024$ for Geo-Trap (U) on UCF-101. However, its optimal performance for U3D (U) on UCF-101 peaks at 83.3% with K set to 256 or 512. Further reducing K compromises video quality due to over-compression. Generally, fewer clusters efficiently counter adversarial perturbations and reduce K-Means clustering costs, but at the expense of reconstructed video quality. As seen in Table 8, a cluster count of 1024 provides a balanced trade-off, enhancing SecVID's average DSR by 29.5% compared to the non-discretized scenario..

6.2.4 Impact of Temporal Loss

SecVID shows improved DSRs for both classifiers across two datasets against five attack types under its default setting of $DoS = 4$ and $K = 1024$, as shown in Table 9, by incorporating temporality L_{temp} (Equation (16)). These improvements are 5.2% for C3D-UCF-101, 8.3% for C3D-HMDB-51, 3.3% for I3D-UCF-101, and 9.1% for I3D-HMDB-51. This underscores the importance of boosting SecVID's defense capabilities by maintaining temporal consistency in videos.

Table 9: Impact of temporal loss L_{temp} (Equation (16)) on SecVID's DSR in the default setting of $DoS = 4$ and $K = 1024$ on adversarial videos, as shown in Table 2.

Model	Attack Method	Without L_{temp}		With L_{temp}	
		UCF-101	HMDB-51	UCF-101	HMDB-51
C3D	StyleFool (U)	78.9%	75.0%	83.5%	84.3%
	U3D (U)	89.5%	83.3%	92.3%	83.3%
	Geo-Trap (U)	90.8%	84.9%	100.0%	100.0%
	StyleFool (T)	76.3%	79.7%	79.2%	81.8%
	Geo-Trap (T)	81.0%	92.2%	83.3%	100.0%
I3D	StyleFool (U)	83.3%	93.7%	88.9%	100.0%
	U3D (U)	81.4%	84.4%	83.1%	94.4%
	Geo-Trap (U)	89.7%	76.2%	91.8%	88.8%
	StyleFool (T)	94.4%	79.8%	96.5%	83.3%
	Geo-Trap (T)	81.1%	78.0%	83.8%	83.1%

6.3 RQ2. SecVID's Video Quality

SecVID adopts a correction-focused strategy to combat adversarial video attacks, using discretization-enhanced VCS. Our experimental findings reveal that while SecVID may not fully return adversarial videos to their original condition, it significantly recovers their original quality and fidelity. This extent of restoration is sufficient for SecVID-enhanced video classifiers to precisely classify videos. We apply three standard metrics adapted from images to videos: SSIM [52], PSNR [70], and FID [39], averaging frame-wise metrics.

SecVID, by default configured to $DoS = 4$ and $K = 1024$, employs discretization-enhanced VCS to counter adversarial perturbations and reconstruct original videos. Figure 4 examines adversarial samples (as detailed Table 2) and their SecVID reconstructions compared to the originals, across varying cluster counts while maintaining $DoS = 4$. SSIM, PSNR, and FID are calculated as averages for each cluster count. In a similar approach, Figure 5 assesses clean videos from the UCF-101 and HMDB-51 datasets.

SecVID effectively maintains the naturalness and realism, i.e., indistinguishability [5] of content reconstructed from both adversarial and clean videos. SSIM scores, reflecting structural similarity, consistently surpass 0.9 for varying K values. PSNR, another measure of video quality, generally stays below 30, regardless of K variations. For adversarial videos, SSIM and PSNR increase, peaking at $K = 1024$, except for U3D (U). Here, SecVID's impact on video quality is more evident in PSNR, mainly due to the residual noise introduced.

FID trends vary; clean videos reconstructed by SecVID typically achieve FID scores below 100, aligning with their favorite SSIM and PSNR values. Conversely, FID scores for adversarial videos can reach up to 300, as depicted in Figure 4,

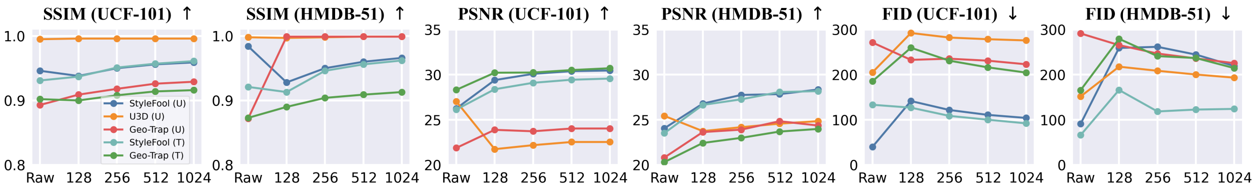


Figure 4: SECVID's restoration of video quality in adversarial videos as detailed in Table 2. This is achieved for both the UCF-101 and HMDB-51 datasets at four distinct cluster counts $K \in \{128, 256, 512, 1024\}$ ($DoS = 4$). 'Raw' denotes the original, unaltered adversarial videos. The evaluation uses the SSIM [52], PSNR [70], and FID [39] metrics.

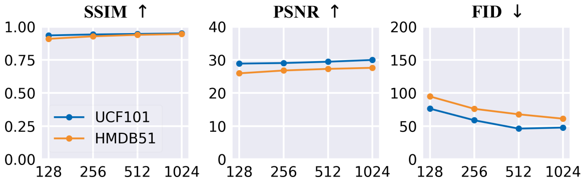


Figure 5: SECVID's effective maintenance of video quality and fidelity for clean videos from UCF-101 and HMDB-51 at four different cluster counts $K \in \{128, 256, 512, 1024\}$ using the PSNR [70], SSIM [52], and FID [39] metrics ($DoS = 4$).

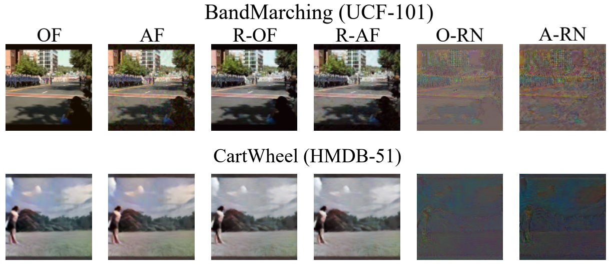


Figure 6: SECVID's differential effects on an original frame (OF) and its corresponding adversarial frame (AF). Typically, the residual noise (A-RN) between AF and its reconstruction (R-AF) is larger than the residual noise (O-RN) between OF and its reconstruction (R-OF). This disparity occurs as R-AF, when processed by SECVID, often retains some perturbations present in, and introduces distortions to, the AF.

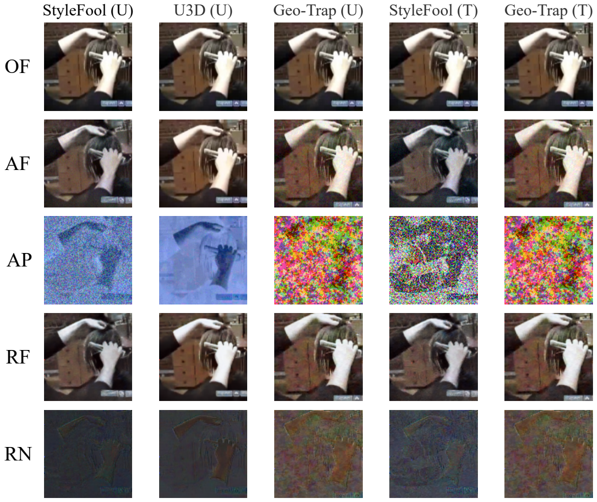


Figure 7: SECVID's video quality on UCF-101, showing adversarial perturbations (AP) between original frame (OF) and adversarial frame (AF), and residual noise (RN) between original frame (OF) and reconstructed frames (RF).

but these attacks are still classified as stealthy [48, 69, 90]. Elevated FID scores in SECVID-reconstructed adversarial videos are due to minor distortions of original features while neutralizing perturbations, a process depicted in Figure 6. Thus, clean videos tend to have lower FID scores compared to their adversarial counterparts. Nevertheless, as shown in Figure 7, SECVID effectively restores original videos and neutralizes perturbations, maintaining their indistinguishability.

6.4 RQ3. SECVID's Security-Related Costs

SECVID excels in defending against adversarial video attacks, ideal for security-focused video recognition systems. However, it involves trade-offs: a slight dip in recognition accuracy, extra training needs, and increased inference times (about two to four times longer) for improved security. Approaches to mitigate these drawbacks are detailed in Section 7.

Accuracy Loss. Figure 8 shows that SECVID slightly reduces accuracy in video classifiers on original datasets, a reasonable trade-off for the improved defense at different *DoS* levels and various cluster counts. As discussed in Section 6.2.1, SECVID's performance in minimizing accuracy loss is on par with four defense baselines. With SECVID, C3D's accuracy on UCF-101 slightly decreases from 78.3% to 73.7%, and on HMDB-51 from 60.2% to 56.9%. Similarly, I3D also sees a small decrease in the classification accuracy, from 87.6% to 85.2% on UCF-101 and from 62.5% to 60.3% on HMDB-51, with the lowest at the highest $DoS = 4$ for HMDB-51. As per Section 6.2.2, SECVID's temporal disruptions, designed to counter adversarial perturbations, might unintentionally impact long-range temporal flow [23, 72] in clean videos, slightly lowering I3D's classification accuracy.

As noted in Section 6.2.2, larger *DoS* levels in SECVID more effectively neutralize perturbations, but concurrently cause a minor reduction in accuracy for clean videos. This indicates SECVID's substantial boost in adversarial resilience at the cost of a small accuracy compromise on original unaltered videos. In high-risk situations like illegal activities

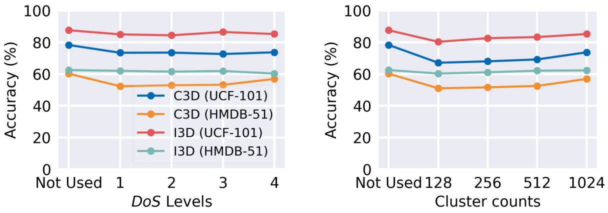


Figure 8: Accuracy of C3D and I3D on clean datasets with and without SECVID at various *DoS* with $K = 1024$ (left) and accuracy of C3D and I3D on clean datasets with and without SECVID at different cluster counts with $DoS = 4$ (right).

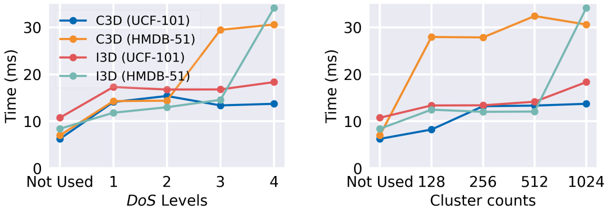


Figure 9: Average model inference time (ms) per sample with and without SECVID at various *DoS* with $K = 1024$ (left) and average model inference time per sample with and without SECVID at different cluster counts with $DoS = 4$ (right).

recognition [77], this trade-off underscores SECVID's effectiveness, where the demand for robust security outweighs the slight drop in video recognition accuracy.

SECVID's Extra Training Times. As per Algorithm 2, SECVID's dataset-specific training involves co-training $\mathcal{T}_{\text{CNN}}$ and $\mathcal{R}_{\text{CNN}}$, with K-Means clustering for Q. Training time depends on hyper-parameters like epochs, batch size, and learning rate. It converges in about 30 epochs for UCF-101 and 60 for HMDB-51. Each epoch lasts about 0.25 hours for both datasets, and constructing Q takes around 8 hours for $K = 1024$, leading to total training times of approximately 7.5 hours for UCF-101 and 15 hours for HMDB-51.

Increased Inference Times. SECVID's enhanced security leads to longer inference times (Figure 9). Using 300 samples from the training dataset per setting, the average per-video inference time increases with SECVID but remains moderate. In the default setting of $DoS = 4$ and $K = 1024$, C3D's time rose by 2.20× to 13.72 ms on UCF-101 and by 4.40× to 30.60 ms on HMDB-51. These times are on par with models like TSM at 29.0 ms [50], ECO at 32.9 ms [101], and InceptionResNet3D-V2 at 25.0 ms [96]. I3D also saw increases of 1.71× on UCF-101 and 4.06× on HMDB-51. Inference time usually increases with higher *DoS* (larger feature maps) and higher K (longer K-Means clustering times).

6.5 RQ4. Defending Against Sparse Attacks

We assess SECVID's robustness against adaptive attacks targeting its sparse transformation using adversarial patches [4].

Table 10: Comparing SECVID and DiffPure for their DSRs when SECVID is under sparse attacks using adversarial patches [4] on UCF-101 and HMDB-51 with C3D and I3D.

Defense	C3D		I3D	
	UCF-101	HMDB-51	UCF101	HMDB-51
SECVID	82.7%	74.7%	89.3%	72.0%
DiffPure [60]	69.3%	72.0%	66.7%	65.3%

Specifically, we compare SECVID's defense capabilities against DiffPure [60], which achieves the highest DSR (Table 3) but also has the longest average inference time per video (Table 5) among the seven baselines.

Adversarial patches exploit spatial sparsity by perturbing a strategically selected small area in each image [4]. We adapted the open-source implementation of this approach from images to videos [35], maintaining its default settings. Each video frame includes a square patch of random noise that occupies 5% of the area. We generated 75 untargeted adversarial samples for each model-dataset pair.

Table 10 demonstrates SECVID's resilience against sparse attacks, which yield slightly lower DSRs compared to non-adaptive scenarios (Table 3). Despite this, SECVID remains robust. Although such sparse attacks are problematic due to their human-perceptibility, as also noted earlier [90], SECVID, purposely designed to counter human-imperceptible perturbations (Section 6.1), effectively mitigates these effects through its discretized compression strategy.

Despite targeted sparse attacks on its sparse transformation, SECVID consistently outperforms DiffPure [60]—the best of the seven baselines—across all four model-dataset pairs. Specifically, SECVID exceeds DiffPure in DSR by 19.3% for C3D-UCF-101, 3.8% for C3D-HMDB-51, 33.9% for I3D-UCF-101, and 10.3% for I3D-HMDB-51, further highlighting its superior defense capabilities.

7 Discussion

Reducing SECVID-Enhanced Classifier Inference Times. Integrating SECVID into video classifiers significantly enhances defense against adversarial attacks and leads to increased inference times, as our evaluation shows—typically a two to fourfold rise. In addressing this, we propose several future strategies. Firstly, SECVID-specific optimizations, such as model pruning [97], quantization [99], and knowledge distillation [58], could expedite the pre-processing module while maintaining robust defense. Secondly, domain-specific hardware acceleration [2] offers a promising avenue to enhance SECVID's performance without compromising its defensive efficacy. Thirdly, SECVID could be selectively deployed for videos preliminarily identified as suspicious through lightweight techniques [89]. Lastly, a random application strategy, similar to ETD screenings at airports [41], may further optimize inference times. An additional benefit

Table 11: Impact of varying **perturbation intensity** (ε) on SecVID's DSR for three untargeted attack types, with $\varepsilon \in \{8/255, 16/255, 32/255, 64/255\}$ ($DoS = 4$ and $K = 1024$).

Attack Method	$\varepsilon = 8/255$	$\varepsilon = 16/255$	$\varepsilon = 32/255$	$\varepsilon = 64/255$
StyleFool (U)	83.5%	78.2%	77.4%	78.1%
U3D (U)	92.3%	95.5%	91.7%	92.3%
Geo-Trap (U)	100.0%	100.0%	100.0%	100.0%

of this approach is a potential further decrease in the already small accuracy loss for clean videos, as indicated in Figure 8.

Impact of Perturbation Intensity ε. In our experiments (Section 6.1), we use $\varepsilon = 8/255$, in line with previous studies [48, 90], as higher perturbation intensities are visible to the human eye. Yet, Table 11 shows SecVID's effective defense even at higher intensities, relevant for attacks like StyleFool [5]. We test C3D on UCF-101 against three untargeted attacks: StyleFool (U), U3D (U), and Geo-Trap (U), simulating real-world misclassification. With StyleFool (U), SecVID's DSR slightly decreased from 83.5% at $\varepsilon = 8/255$ to 78.1% at $\varepsilon = 64/255$. U3D (U)'s DSR varied, but returned to 92.3% at $\varepsilon = 64/255$, the same as at $\varepsilon = 8/255$. For Geo-Trap (U), SecVID maintained 100.0% DSR across all intensities, demonstrating its strong defense.

8 Related Work

The susceptibility of AI systems, especially in deep learning, to adversarial perturbations is a critical issue recognized by both the security and deep learning communities. Adversarial attacks on video classification are categorized into white-box [1, 10, 31, 49, 86] and black-box [5, 15, 38, 48, 57, 90]. Existing approaches mainly optimize adversarial perturbations within ℓ_p norms using methods like FGSM [22] or PGD [76], focusing on gradient estimation. In a white-box context, C-DUP [49] creates offline universal adversarial perturbations, drawing inspiration from GANs. V-BAD [38] targets black-box classifiers with limited-query attacks. A heuristic approach reducing perturbation intensity in black-box attacks was introduced, focusing on key frames' salient regions [86]. StyleFool [5] introduced black-box attacks on video classification through style transfer, utilizing texture and color changes. U3D [90] enhanced black-box attack transferability with universal three-dimensional perturbations.

Most existing defense mechanisms against adversarial attacks are tailored to images, focusing either on improving model robustness (like Adversarial Training (AT) [27, 42, 71, 78], Input Transformations (IT) [28], Random Smoothing (RS) [34]), ComDefend [37] and DiffPure [60]) or detecting attacks [25, 53, 55]. In particular, ComDefend, similar to [26], utilizes compression and reconstruction CNNs for image reconstruction. DiffPure employs diffusion models to purify images, resulting in high computational costs.

AdvIT [89], leveraging temporal consistency, is the first video-specific defense, but it only detects, not rectifies, adversarial videos. DP [46] and OUDefend [51] are two additional video-specific defenses. DP customizes defense patterns based on specific known adversarial perturbations. Meanwhile, OUDefend utilizes over/undercomplete features of videos within a feature restoration network, tailored to the video classifier's architecture, to defend against adversarial videos. Adversarial training [42, 78], adapted from images to videos, requires prior knowledge of adversarial perturbations. Random Smoothing [34], which uses Gaussian noise to avoid certain downsides, underperforms in defending videos, as noted in [5] and confirmed in Table 3. Finally, self-adaptive JPEG compression and optical texture analysis are used in video defenses [13]. This method is especially effective for large objects, where distinct textures and sizes facilitate better compression adjustments and more robust texture analysis.

9 Conclusion

We introduce SecVID in response to increasing attacks on DNN-based video recognition systems. This correction-based defense uses video compressive sensing to address adversarial threats through sparse transformation, discretization, and advanced reconstruction, maintaining video quality and integrity. While enhancing system defense, SecVID introduces trade-offs, including slightly reduced recognition accuracy, additional training for its pre-processing module, and longer inference times. These trade-offs pinpoint future research directions, especially in cost-effective methods that selectively impact security. As reliance on DNNs grows in security-critical areas, SecVID serves as an effective countermeasure against advanced adversarial video attacks, providing crucial insights for developing robust defenses.

Acknowledgments

We express our gratitude to the reviewers for their constructive feedback, which has significantly improved both the presentation and quality of our paper. This research is supported by ARC Grant DP230102871.

References

[1] Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. *NeurIPS'20*, 33:16048–16059.

[2] Mostafa Rahimi Azghadi, Corey Lammie, Jason K. Eshraghian, Melika Payvand, Elisa Donati, Bernabé Linares-Barranco, and Giacomo Indiveri. Hardware implementation of deep network accelerators towards healthcare and biomedical applications. *IEEE Transactions on Biomedical Circuits and Systems (TBCAS)*, 14:1138–1159, 2020.

[3] Kartikeya Bhardwaj, Dibakar Gope, James Ward, Paul Whatmough, and Danny Loh. Super-efficient super resolution for fast adversarial defense at the edge. In *DATE'22*, pages 418–423.

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

[5] Yuxin Cao, Xi Xiao, Ruoxi Sun, Derui Wang, Minhui Xue, and Sheng Wen. StyleFool: Fooling video classification systems via style transfer. In *S&P'23*, pages 1631–1648.

[6] Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David Wagner, and Wenchao Zhou. Hidden voice commands. In *USENIX Security'16*, pages 513–530.

[7] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *S&P'17*, pages 39–57, 2017.

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR'17*, pages 6299–6308.

[9] Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

[10] Jung-Woo Chang, Mojan Javaheripi, Seira Hidano, and Farinaz Koushanfar. Adversarial attacks on deep learning-based video compression and classification systems. *arXiv preprint arXiv:2203.10183*, 2022.

[11] Hao Chen, Matthew Gwilliam, Ser-Nam Lim, and Abhinav Shrivastava. HNeRV: A hybrid neural representation for videos. In *CVPR'23*, pages 10270–10279.

[12] Peng Chen and Omar Ghattas. Projected stein variational gradient descent. *NeurIPS'20*, 33:1947–1958.

[13] Yupeng Cheng, Xingxing Wei, Huazhu Fu, Shang-Wei Lin, and Weisi Lin. Defense for adversarial videos by self-adaptive jpeg compression and optical texture. In *MM Asia'21*, pages 1–7.

[14] Ziheng Cheng, Bo Chen, Guanliang Liu, Hao Zhang, Ruiying Lu, Zhengjue Wang, and Xin Yuan. Memory-efficient network for large-scale video compressive sensing. In *CVPR'21*, pages 16246–16255.

[15] Kenneth T Co, Luis Muñoz-González, Sixte de Maupeou, and Emil C Lupu. Procedural noise adversarial examples for black-box attacks on deep convolutional networks. In *CCS'19*, pages 275–289.

[16] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *ICML'19*, pages 1310–1320.

[17] Vincent Cohen-Addad, Kasper Green Larsen, David Saulpic, Chris Schwiegelshohn, and Omar Ali Sheikh-Omar. Improved coresets for euclidean *k*-means. *NeurIPS'22*, 35:2679–2694.

[18] John Collomosse, Tu Bui, and Hailin Jin. Livesketch: Query perturbations for guided sketch-based visual search. In *CVPR'19*, pages 2879–2887.

[19] Sanjoy Dasgupta, Nave Frost, Michal Moshkovitz, and Cyrus Rashtchian. Explainable k-means and k-medians clustering. In *ICML'20*, pages 12–18.

[20] Nelson Diaz, Carlos Hinojosa, and Henry Arguello. Adaptive grayscale compressive spectral imaging using optimal blue noise coding patterns. *Optics & Laser Technology*, 117:147–157, 2019.

[21] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR'15*, pages 2625–2634.

[22] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *CVPR'19*, pages 4312–4321.

[23] Zheng Fang, Qingqing Long, Guojie Song, and Kunqing Xie. Spatial-temporal graph ODE networks for traffic flow forecasting. In *SIGKDD'21*, pages 364–373.

[24] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR'16*, pages 1933–1941.

[25] Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

[26] Claudio Ferrari, Federico Becattini, Leonardo Galteri, and Alberto Del Bimbo. A robust defense against adversarial attacks on image classification. *IEEE Transactions on Mobile Computing (TMC)*, 19:1–16, 2023.

[27] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[28] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

[29] Sicong Han, Chenhao Lin, Chao Shen, Qian Wang, and Xiaohong Guan. Interpreting adversarial examples in deep learning: A review. *ACM Computing Surveys*, 55(14s), 2023.

[30] Nathan Inkawhich, Matthew Inkawhich, Yiran Chen, and Hai Li. Adversarial attacks for optical flow-based action recognition classifiers. *arXiv preprint arXiv:1811.11875*, 2018.

[31] Nathan Inkawhich, Matthew Inkawhich, Yiran Chen, and Hai Li. Adversarial attacks for optical flow-based action recognition classifiers. *arXiv preprint arXiv:1811.11875*, 2018.

[32] Ajil Jalal, Sushrut Karmalkar, Alexandros G Dimakis, and Eric Price. Instance-optimal compressed sensing via posterior sampling. *arXiv preprint arXiv:2106.11438*, 2021.

[33] Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *CVPR'16*, pages 3852–3861.

[34] JeremyCohen. Random smoothing, 2019. Accessed: 2023-11-07.

[35] Jhayes. Adversarial patch implementation, 2018. Accessed: 2024-05-07.

[36] Xiaojun Jia. ComDefend implementation, 2019. Accessed: 2024-05-07.

[37] Xiaojun Jia, Xingxing Wei, Xiaochun Cao, and Hassan Foroosh. ComDefend: An efficient image compression model to defend adversarial examples. In *CVPR'19*, pages 6084–6092.

[38] Linxi Jiang, Xingjun Ma, Shaoxiang Chen, James Bailey, and Yu-Gang Jiang. Black-box adversarial attacks on video recognition models. In *MM'19*, pages 864–872.

[39] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR'20*, pages 8110–8119.

[40] Minal S Khandare and Anjali Mahajan. Mobile monitoring system for smart home. In *ICETET'10*, pages 848–852.

[41] Mun Hwan Kim, Jin Woo Park, and Yu Jin Choi. A study on the effects of waiting time for airport security screening service on passengers' emotional responses and airport image. *Sustainability*, 12:10634, 2020.

[42] Kaleab A Kinfu and René Vidal. Analysis and extensions of adversarial training for video classification. In *CVPR'22*, pages 3416–3425.

[43] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV'11*, pages 2556–2563.

[44] Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. Procedural noise using sparse gabor convolution. 28:1–10, 2009.

[45] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *S&P'19*, pages 656–672.

[46] Hong Joo Lee and Yong Man Ro. Defending video recognition model against adversarial perturbations via defense patterns. *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 2023.

[47] Qian Li, Yuxiao Hu, Ye Liu, Dongxiao Zhang, Xin Jin, and Yuntian Chen. Discrete point-wise attack is not enough: Generalized manifold adversarial attack for face recognition. In *CVPR'23*, pages 20575–20584.

[48] Shasha Li, Abhishek Aich, Shitong Zhu, Salman Asif, Chengyu Song, Amit Roy-Chowdhury, and Srikanth Krishnamurthy. Adversarial attacks on black box video classifiers: Leveraging the power of geometric transformations. *NeurIPS'21*, 34:2085–2096.

[49] Shasha Li, Ajaya Neupane, Sujoy Paul, Chengyu Song, Srikanth V Krishnamurthy, Amit K Roy Chowdhury, and Ananthram Swami. Adversarial perturbations against real-time video classification systems. *arXiv preprint arXiv:1807.00458*, 2018.

[50] Ji Lin, Chuang Gan, and Song Han. TSM: Temporal shift module for efficient video understanding. In *ICCV'19*, pages 7083–7093.

[51] Shao-Yuan Lo, Jeya Maria Jose Valanarasu, and Vishal M Patel. Overcomplete representations against adversarial videos. In *ICIP*, pages 1939–1943, 2021.

[52] Kede Ma, Kai Zeng, and Zhou Wang. Perceptual quality assessment for multi-exposure image fusion. *IEEE Transactions on Image Processing (TIP)*, 24(11):3345–3356, 2015.

[53] Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*, 2018.

[54] Xiaofeng Mao, Yuefeng Chen, Ranjie Duan, Yao Zhu, Gege Qi, Xiaodan Li, Rong Zhang, Hui Xue, et al. Enhance the visual representation via discrete adversarial training. *NeurIPS'22*, 35:7520–7533.

[55] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *CCS'17*, pages 135–147.

[56] Yisroel Mirsky, Tom Mahler, Ilan Shelef, and Yuval Elovici. CT-GAN: Malicious tampering of 3D medical imagery using deep learning. In *USENIX Security'19*, pages 461–478.

[57] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *CVPR'17*, pages 1765–1773.

[58] Ravi Teja Mullapudi, Steven Chen, Keyi Zhang, Deva Ramanan, and Kayvon Fatahalian. Online model distillation for efficient video inference. In *CVPR'19*, pages 3573–3582.

[59] Weili Nie. Diffpure implementation, 2022. Accessed: 2024-05-07.

[60] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022.

[61] Tadashi Onishi, Toshiyuki Motoyoshi, Yuki Suga, Hiroki Mori, and Tsuya Ogata. End-to-end learning method for self-driving cars with trajectory recovery using a path-following function. In *IEEE International Joint Conference on Neural Networks (IJCNN)*, 2019.

[62] Ken Perlin. An image synthesizer. *ACM Siggraph Computer Graphics*, 19:287–296, 1985.

[63] Roi Pony, Itay Naeh, and Shie Mannor. Over-the-air adversarial flickering attacks against video recognition networks. In *CVPR'21*, pages 515–524.

[64] Gilles Puy and Patrick Pérez. A flexible convolutional solver for fast style transfers. In *CVPR'19*, pages 8963–8972.

[65] Mohammad Saeed Rad, Behzad Bozorgtabar, Urs-Viktor Marti, Max Basler, Hazim Kemal Ekenel, and Jean-Philippe Thiran. SROBB: Targeted perceptual loss for single image super-resolution. In *ICCV'19*, pages 2710–2719.

[66] DM Motiur Rahaman and Manoranjan Paul. Virtual view synthesis for free viewpoint video and multiview video compression using gaussian mixture modelling. *IEEE Transactions on Image Processing (TIP)*, 27:1190–1201, 2017.

[67] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *CVPR'17*, pages 4161–4170.

[68] Viktor Rausch, Andreas Hansen, Eugen Solowjow, Chang Liu, Edwin Kreuzer, and J Karl Hedrick. Learning a deep neural net policy for end-to-end control of autonomous vehicles. In *ACC'17*, pages 4914–4919.

[69] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, 128:2586–2606, 2020.

[70] De Rosal Igantius Moses Setiadi. PSNR vs SSIM: imperceptibility quality assessment for image steganography. *Multimedia Tools and Applications*, 80:8423–8444.

[71] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *NeurIPS'19*, 32.

[72] Mykhailo Shvets, Wei Liu, and Alexander C Berg. Leveraging long-range temporal relationships between proposals for video object detection. In *ICCV'19*, pages 9756–9764.

[73] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *NeurIPS'14*, page 568–576.

[74] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[75] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR'12*, abs/1212.0402.

[76] Gaurang Sriramanan, Sravanti Addepalli, Arya Baburaj, et al. Guided adversarial attack for evaluating and enhancing adversarial defenses. *NeurIPS'20*, 33:20297–20308.

[77] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *CVPR'18*, pages 6479–6488.

[78] Nupur Thakur and Baoxin Li. Pat: Pseudo-adversarial training for detecting adversarial videos. In *CVPR'22*, pages 131–138.

[79] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *ICCV'15*, pages 4489–4497.

[80] Trung Vu and Raviv Raich. On asymptotic linear convergence of projected gradient descent for constrained least squares. *IEEE Transactions on Signal Processing*, 70:4061–4076, 2022.

[81] Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. In *AAAI'19*, volume 33, pages 5232–5239, 2019.

[82] Huan Wang, Yijun Li, Yuehai Wang, Haoji Hu, and Ming-Hsuan Yang. Collaborative distillation for ultra-resolution universal style transfer. In *CVPR'20*, pages 1860–1869.

[83] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. CNN-generated images are surprisingly easy to spot... for now. In *CVPR'20*, pages 8695–8704.

[84] Zhe Wang, Songling Huang, Shen Wang, Shuangyong Zhuang, Qing Wang, and Wei Zhao. Compressed sensing method for health monitoring of pipelines based on guided wave inspection. *IEEE Transactions on Instrumentation and Measurement (TIM)*, 69:4722–4731, 2019.

[85] Zhengjue Wang, Hao Zhang, Ziheng Cheng, Bo Chen, and Xin Yuan. MetaSCI: Scalable and adaptive reconstruction for video compressive sensing. In *CVPR'21*, pages 2083–2092.

[86] Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *AAAI'19*, volume 33, pages 8973–8980.

[87] Jinming Wen, Rui Zhang, and Wei Yu. Signal-dependent performance analysis of orthogonal matching pursuit for exact sparse recovery. *IEEE Transactions on Signal Processing*, 68:5031–5046, 2020.

[88] Kaiguo Xia, Zhisong Pan, and Pengqiang Mao. Video compressive sensing reconstruction using unfolded lstm. *Sensors*, 22:7172, 2022.

[89] Chaowei Xiao, Ruizhi Deng, Bo Li, Taesung Lee, Benjamin Edwards, Jinfeng Yi, Dawn Song, Mingyan Liu, and Ian Molloy. AdvIT: Adversarial frames identifier based on temporal consistency in videos. In *CVPR'19*, pages 3968–3977.

[90] Shangyu Xie, Han Wang, Yu Kong, and Yuan Hong. Universal 3-dimensional perturbations for black-box attacks on video recognition systems. In *S&P'22*, pages 1390–1407.

[91] Jianjin Xu, Zheyang Xiong, and Xiaolin Hu. Frame difference-based temporal loss for video stylization. *arXiv preprint arXiv:2102.05822*, 2021.

[92] Jianbo Yang, Xin Yuan, Xuejun Liao, Patrick Llull, David J Brady, Guillermo Sapiro, and Lawrence Carin. Video compressive sensing using gaussian mixture models. *IEEE Transactions on Image Processing (TIP)*, 23:4863–4878, 2014.

[93] Jiaolong Yang, Peiran Ren, Dongqing Zhang, Dong Chen, Fang Wen, Hongdong Li, and Gang Hua. Neural aggregation network for video face recognition. In *CVPR'17*, pages 4362–4371.

[94] Di You, Jian Zhang, Jingfen Xie, Bin Chen, and Siwei Ma. COAST: Controllable arbitrary-sampling network for compressive sensing. *IEEE Transactions on Image Processing (TIP)*, 30:6066–6080, 2021.

[95] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR'15*, pages 4694–4702.

[96] Yifan Zhang, Lei Shi, Yi Wu, Ke Cheng, Jian Cheng, and Hanqing Lu. Gesture recognition based on deep deformable 3D convolutional neural networks. *Pattern Recognition*, 107:107416, 2020.

[97] Yihua Zhang, Yuguang Yao, Parikshit Ram, Pu Zhao, Tianlong Chen, Mingyi Hong, Yanzhi Wang, and Sijia Liu. Advancing model pruning via bi-level optimization. *NeurIPS'22*, 35:18309–18326.

[98] Yun Zhang, Huan Zhang, Mei Yu, Sam Kwong, and Yo-Sung Ho. Sparse representation-based video quality assessment for synthesized 3D videos. *IEEE Transactions on Image Processing (TIP)*, 29:509–524, 2019.

[99] Yiren Zhou, Seyed-Mohsen Moosavi-Dezfooli, Ngai-Man Cheung, and Pascal Frossard. Adaptive quantization for deep neural network. In *AAAI'20*.

[100] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *CVPR'19*, pages 989–997.

[101] Mohammadreza Zolfaghari, Kamaljeet Singh, and Thomas Brox. Eco: Efficient convolutional network for online video understanding. In *ECCV'18*, pages 695–712.