Flickering Adversarial Attacks against Video Recognition Networks

Itay Naeh [*][1] **Roi Pony** [*][2] **Shie Mannor** [2]

Abstract

Deep neural networks for video classification, just like image classification networks, may be subjected to adversarial manipulation. The main difference between image classifiers and video classifiers is that the latter usually use temporal information contained within the video. In this work we present a manipulation scheme for fooling video classifiers by introducing a flickering temporal perturbation that is practically unnoticeable by human observers and is implementable in the real world. After demonstrating the manipulation of action classification of single videos, we generalize the procedure to make universal adversarial perturbation by using only dozens of videos, achieving high fooling ratio. In addition, we generalize the universal perturbation and produce a temporal-invariant perturbation, which can be applied to the video without synchronizing the perturbation to the input. These properties allow us to bridge the gap between simulated environment and real-world application.

1. Introduction

In recent years, Deep Neural Networks (DNNs) have shown phenomenal performance in a wide range of tasks, such as image classification (Krizhevsky et al., 2012), object detection (Ren et al., 2015), semantic segmentation (Shelhamer et al., 2017) etc. Despite their success, DNNs have been found vulnerable to adversarial attacks. Many works (Szegedy et al., 2014; Goodfellow et al., 2015a; Papernot et al., 2015) have shown that a small (sometimes imperceptible) perturbation added to an image, can make a given DNNs prediction false. These findings have raised many concerns, particularly for critical systems such as face recognition systems (Sun et al., 2014), surveillance cameras (Sultani et al., 2018), autonomous vehicles and medical applications (Litjens et al., 2017). In recent years most of the attention was given to the study of adversarial patterns in images and less in video action recognition. Only in the last two years works on adversarial video attacks were published (Wei et al., 2019; Inkawhich et al., 2018; Zhipeng Wei, 2019; Jiang et al., 2019), even though DNNs have been applied to video-based tasks for several years, in particular video action recognition (Carreira & Zisserman, 2017; Wang et al., 2017; Feichtenhofer et al., 2018). In video action recognition networks temporal information is of the essence in categorizing actions, in addition to per-frame image classification. In some of the proposed attacks the emphasis was, beyond adversarial categorization, the sparsity of the perturbation. In our paper, we consider adversarial attacks against video action recognition under white-box setting, with an emphasis on the imperceptible nature of the perturbation in the spatio-temporal domain to the human observer and implementability of the generalized adversarial perturbation in the real-world. We introduce flickering perturbation by applying a uniform RGB perturbation to each frame, thus constructing a temporal adversarial pattern.

Unlike previous works, in our case sparsity of the pattern is undesirable, because it helps the adversarial perturbation to be detectable by human observers for its unnatural pattern, and to image based adversarial perturbation detectors for the exact same reason. The adversarial perturbation presented in this work does not contain any spatial information on a single frame other than a constant (and usually small) offset, as presented in Figure 1. This type of perturbation occurs in natural videos by changing lighting (interlacing or natural), auto-gain corrections, scene changes, etc. In this paper, we aim to attack the video action recognition task (Kay et al., 2017). For the threat model we choose the I3D (Carreira & Zisserman, 2017) model based on InceptionV1 (Szegedy et al., 2015). Specifically we attack the RGB stream of the model, rather than on the easier to influence optical flow stream. The attacked network trained on the Kinetics-400 Human Action Video Dataset (Kay et al., 2017).

In order to make the adversarial perturbation unnoticed by human observers, we reduce the thickness and temporal roughness of the adversarial perturbation, which will be defined precisely later on. In order to do so we apply three

[*] Equal contribution [1] Rafael - Advanced Defense Systems Ltd., Israel [2] Department of Electrical Engineering, Technion Institute of Technology, Haifa, Israel. Correspondence to: Itay Naeh <itay@naeh.us>, Roi Pony <roi.pony@campus.technion.ac.il>.

regularization terms during the optimization process, each corresponds to a different effect of the perceptibly of the adversarial pattern. In addition, we introduce an improved adversarial-loss function that allows better integration of these regularization terms with the adversarial loss.

We will first introduce a flickering attacks on a single video and present the trade-off between the different regularization terms. We construct universal perturbation which generalize over classes and achieves 60% fooling rate, using only 50 video examples, and 90% using 500 examples. In addition, we introduce time invariant perturbation, which can be applied to the classifier without synchronization. This allows the perturbation to be relevant for real world scenarios, for frame synchronization is rarely possible. The main contributions of this work are:

- A methodology for developing flickering adversarial attack against video action recognition networks which incorporate a new type of regularization for affecting the visibility of the adversarial pattern.
- A universal, time-invariant adversarial perturbations that does not require frame synchronization.
- Adversarial pattern that generalizes with only dozens of examples.
- Adversarial perturbation which is implementable in the real world by using not spatial perturbation, but a temporal one.

The rest of the paper is organized as follows: In Section 2 we briefly review related work. In Section 3 we present the flickering adversarial attack. Section 4 shows experimental results. Section 4.4 presents generalization of the adversarial attack. Finally, we present conclusions and future work in Section 5.

We encourage the readers to view the adversarial videos[1] and additional material in the project page[2].

2. Related Work

2.1. Video Action Recognition

With deep convolutional neural networks (CNN) achieving state-of-the-art performance on image recognition task, many works propose to adapt this achievement to video-based computer vision tasks. In particular, DNNs for video action recognition. The most straightforward approach for acheiving this is to add temporally-recurrent layers such as LSTM (Sak et al., 2014) models to the traditional 2D-CNN. This way, long term temporal dependencies can be assigned to spatial features (Wang et al., 2018; Simonyan & Zisserman, 2014). Another approach implemented in C3D (Ji et al., 2013; Tran et al., 2014; Varol et al., 2016) extends the 2D CNN (image-based) to 3D CNN (video-based) kernels and learns hierarchical spatio-temporal representations directly from the raw videos. Despite the simplicity of this approach, it is very difficult to train this network due to its huge parameter space. To address this, (Carreira & Zisserman, 2017) proposes the Inflated 3D CNN (I3D) with inflated 2D pre-trained filters (Russakovsky et al., 2014). In addition to the RGB pipeline, the optical flow is also useful for temporal information encoding, and indeed several architectures greatly improved their performance by incorporating an optical-flow stream (Carreira & Zisserman, 2017).

2.2. Adversarial Attack on Video Models

The research of the vulnerability of video-based classifiers to adversarial attacks emerged only in the last years. The following attacks were perform under the white-box attack settings: (Wei et al., 2019) were the first to investigate a white-box attack on video action recognition. They proposed an $L_{2,1}$ norm based optimization algorithm to compute sparse adversarial perturbations. Unlike our threat model, they choose the networks with a CNN+RNN architecture in order to investigate the propagation properties of perturbations. (Li et al., 2019) generated an offline universal perturbation using GAN-based model, that they applied to the learned model on unseen input for real-time video recognition models. (Rey-de-Castro & Rabitz, 2018) proposed a nonlinear adversarial perturbation by using another neural network model (besides the attacked model), which was optimized to transform the input into adversarial pattern under the L_1 norm. (Inkawhich et al., 2018) proposed both white and black box untargeted attacks on two-stream model (optical-flow and RGB), based on the original and the iterative version of FGSM (Goodfellow et al., 2015b; Kurakin et al., 2017), and use FlowNet2 (Ilg et al., 2016) to estimates optical flow in order to provide gradients estimation.
Several black-box attacks were proposed (Jiang et al., 2019; Zhipeng Wei, 2019). Our attack follows the white-box setting therefore those are beyond the scope of this paper.

3. Flickering Adversarial attack

The flickering adversarial attack consists of a uniform offset added to the entire frame that changes with each frame. This novel approach is desirable for several reasons. First, it contains no spatial pattern within individual frames but a RGB offset. Second, this type of perturbation can easily be mistaken (if visible by the human eye at all) as changing lighting conditions of the scene or typical sensor behaviour.

[1] https://bit.ly/Patternless_videos
[2] https://github.com/anonymous-p/Flickering_Adversarial_Video

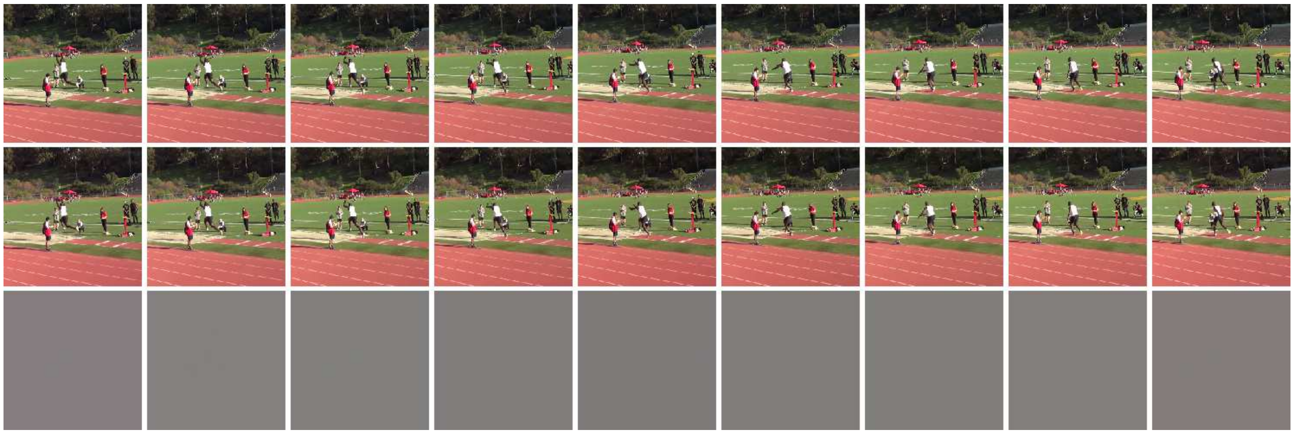

Figure 1. Top: Consecutive frames of the original video from the ”Triple jump” category. Middle: Consecutive frames of the miss-classified adversarial video. The adversarial perturbation is practically unnoticed by the human observer. Bottom: The flickering adversarial perturbation of each frame. The perturbation is a constant offset applied to the entire frame. Due to the fact that the perturbation can be negative, it is displayed over gray background. The gentle hue changes displayed at each frame are the adversarial pattern.

Third, the nature of the perturbation allowing the generalization of the attack by using small amount of training examples.

3.1. Preliminaries

Video action recognition is a function $F_\theta(X) = y$ that accepts an input $X = [x_1, x_2, .., x_T] \in \mathbb{R}^{T\times H\times W\times C}$ from T consecutive frames with H rows, W columns and C color channels, and produces an output $y \in \mathbb{R}^K$ which can be treated as probability distribution over the output domain, where K is the number of classes. The model F implicitly depends on some parameters θ that are fixed during the attack. The classifier assigns the label $A_\theta(X) = \text{argmax}_i\, y_i$ to the input X. We denote $adversarial$ video by $\hat{X} = X + \delta$ where the video $perturbation$ $\delta = [\delta_1, \delta_2, .., \delta_T] \in \mathbb{R}^{T\times H\times W\times C}$, and each individual adversarial frame by $\hat{x}_i = x_i + \delta_i$

3.1.1. Targeted and Untargeted Attack

Untargeted attack cause the target model to classify adversarial examples to any label but the correct class, i.e. $A_\theta(X) \neq A_\theta(\hat{X})$. These attacks minimize the probability of the targeted class until a different class becomes most probable.

Targeted attack cause the target model to classify adversarial examples to a specific predetermined incorrect class. The objective become $A_\theta(\hat{X}) = t$, where t is the target class. This attack usually lowers the probability of the original class by raising the probability of the adversarial class.

3.1.2. Video Adversarial Attack

Given video $X = [x_1, x_2, .., x_T]$, we would like to construct another video $\hat{X} = [\hat{x}_1, \hat{x}_2, .., \hat{x}_T]$ that meets the following requirements: $\hat{X}$ is adversarial i.e. $A_\theta(X) \neq A_\theta(\hat{X})$ (Untargeted Attack in this example), X and $\hat{X}$ look similar, in particular x_i and $\hat{x}_i$ look similar $\forall i$

3.2. Threat Model

Our threat model follows the white-box setting, which assumes the complete knowledge of the targeted model, its parameter values and architecture. In the experiments, video recognition model I3D (Carreira & Zisserman, 2017) is used as target model, focused on the RGB pipeline. The adversarial attacks described in this work are both targeted and untargeted, and the theory and implementation can be easily adapted accordingly. This network was selected for targeting because common video classification networks are based upon it’s architecture. Therefore, the insights derived from this work will be relevant for these networks.

3.3. Dataset

We use Kinetics-400 (Kay et al., 2017) for our experiments. Kinetics is a standard benchmark for action recognition in videos. It contains about 275K video of 400 different human action categories (220K - training split, 18K - validation split, 35K - test split). This Dataset was selected for it is used for pretraining several SOTA video classification networks on various datasets such as (Soomro et al., 2012; Kuehne et al., 2011). In Sections 4.3 and 4.4.2 we used the validation split. In Section 4.4.1 we train on the training split and evaluate on the validation and test split. We preprocess the dataset by excluding the movies on which the network misclassified to begin with. Each video contains 90-frame snippets.

3.4. Methodology

In our attack δ_i is designed to be spatial-constant on the three color channels of the frame, meaning for each pixel in image x_i offset is added with same value (RGB). Thus, the i^{th} perturbation, which corresponds with the i^{th} frame of the video, can be represented by three scalars, hence $\delta \in \mathbb{R}^{T\times 1\times 1\times 3}$, having in total $3T$ parameter to optimize. To generate adversarial perturbation (in the general case), we use the following objective function

$$\operatorname*{argmin}_{\delta} \lambda \sum_j \beta_j D_j(\delta) + \frac{1}{N}\sum_{n=1}^{N} \ell(F_\theta(X_n+\delta), t_n) \quad (1)$$

$$s.t\ \hat{x}_i \in [V_{min}, V_{max}]^{H\times W\times C}. \quad (2)$$

where N is total number of training videos, X_n is the n^{th} video, $F_\theta(X_n+\delta)$ is the classifier output (probability distribution or logits) and t_n is the targeted label in targeted attack or the original label in untargeted attack. The first terms in Equation (1) are the regularization terms, while the second are the adversarial classification loss (Section 3.5).

The parameter λ weights the relative importance of being adversarial and the regularization terms. The set of functions $D_j(\cdot)$ controls the regularization terms that allows us to achieve better imperceptibility for the human observer. The parameter β_j weights the relative importance of each regularization term. The constrain in Equation (2) makes sure that after applying the adversarial perturbation, the perturbed video will be clipped between the valid values: V_{min}, V_{max}, that represents the minimum and maximum allowed gray-level values.

3.5. Improved Adversarial loss function

For achieving a more stable convergence, we have developed on top (Carlini & Wagner, 2016) a new loss mechanism, which reaches the adversarial goal only to the desired extent, leaving space for other regularization terms. For untargeted attack:

$$\ell(y,t) = \max\left(0, \min\left(\frac{1}{m}\ell_m(y,t)^2, \ell_m(y,t)\right)\right) \quad (3)$$

$$\ell_m(y,t) = y_t - \max_{i\neq t}(y_i) + m. \quad (4)$$

$m > 0$ is the desired margin of the original class probability below the adversarial class probability. A more detailed explanation of the motivation in defining the above loss function is found in the section A.1.

3.6. Regularization terms

We quantify the distortion introduced by the perturbation δ with $D(\delta)$ in the spatio-temporal domain. This metric will be constrained in order for the perturbation δ to be imperceptible to the human observer while remaining adversarial. Unlike previously published work on adversarial patches in images, in video domain imperceptible may reference thin patches in gray-level space or slow changing patches in temporal frame space. In contrast to previously related work (Wei et al., 2019; Zhipeng Wei, 2019) in our case temporal sparsity is not of the essence but the unnoticability to the human observer. In order to achieve the most imperceptible perturbation we introduce three regularization terms, each one controls different aspects that may influence human perception mechanisms.

In order to simplify the following sections, we define these notations: Given tensor $X \in \mathbb{R}^{T\times H\times W\times C}$.

Roll operator. $Roll(X,\tau)$ produce the time shifted tensor, whose elements are τ-cyclic shifted along the first axis (time):

$$Roll(X,\tau) = [x_{(\tau \bmod T)+1}, x_{(1+\tau \bmod T)+1}, \cdots \\ , x_{(T-1+\tau \bmod T)+1}]$$

Tensor p-norm

$$\|X\|_p = \left(\sum_{i_1=1}^{T}\cdots\sum_{i_4=1}^{C} |x_{i_1\ldots i_4}|^p\right)^{1/p}.$$

1^{st} ***and*** 2^{nd} ***order temporal derivatives***. We approximate the 1^{st} and 2^{nd} order temporal derivatives by finite differences as follows.

$$\frac{\partial X}{\partial t} = Roll(X,1) - Roll(X,0)$$

$$\frac{\partial^2 X}{\partial t^2} = Roll(X,-1) - 2Roll(X,0) + Roll(X,1)$$

3.6.1. Thickness regularization

This loss term forces the adversarial perturbation to be as small as possible in gray-level over the three color channels (per-frame), having no temporal constraint and can be related to the ”thickness” of the adversarial pattern.

$$D_1(\delta) = \frac{1}{3T}\|\delta\|_2^2$$

3.6.2. Roughness regularization

We introduce temporal loss functions which incorporate two different terms,

$$D_2(\delta) = \frac{1}{3T}\left\|\frac{\partial \delta}{\partial t}\right\|_2^2 + \frac{1}{3T}\left\|\frac{\partial^2 \delta}{\partial t^2}\right\|_2^2 \quad (5)$$

The first order temporal difference shown in the Equation (5) (first term) controls the difference between each two consecutive frame perturbations. This term penalize for temporal changes of the adversarial pattern. Within the context of human visual perception, this term is perceived as ”flickering”, thus we wish to minimize it.

The second order temporal difference shown in Equation (5) (second term) controls the trend of the adversarial perturbation. Visually, this term inflicts penalty on fast trend changes, such as spikes, and may be considered as scintillation reducing term.

The weighting between the D_1 and D_2 will be noted by β_1 and β_2, respectively throughout the rest of the paper and also in the YouTube videos.

4. Experiments

4.1. Implementation Details

Experiment codes are implemented in TensorFlow[3] and based on I3D source code[4]. The code is executed on a server with four Nvidia Titan-X GPUs, Intel i7 processor and 128GB RAM. For optimization we adopt the ADAM (Kingma & Ba, 2014) optimizer with learning rate of 1e-3 and with batch size of 8 for the generalization section and 1 for a single video attack. Except where explicitly stated $\beta_1 = \beta_2 = 0.5$. For single video attack $\lambda = 1$ and for generalization sections $\lambda = 100$. In the kinetics Dataset $V_{min} = -1, V_{max} = 1$. In the I3D configuration $T = 90, H = 224, W = 224, C = 3$.

4.2. Metrics

. Let us define several metrics in order to quantify the performance of our adversarial attacks.

- ***Fooling ratio***: is defined as the percentage of adversarial videos that are successfully misclassified.
- ***Mean Absolute Perturbation per-pixel***:

$$thickness_{gl}(\delta) = \frac{1}{3T} \left\| \delta \right\|_1^2 .$$

- ***Mean Absolute Temporal-diff Perturbation per-pixel***:

$$roughness_{gl}(\delta) = \frac{1}{3T} \left\| \frac{\partial \delta}{\partial t} \right\|_1^2 .$$

The thickness and roughness values in this paper will be presented as percents from the full applicable values of the image span, e.g.

$$thickness(\delta) = \frac{thickness_{gl}(\delta)}{V_{max} - V_{min}} * 100.$$

4.3. Single Video Attack

In order to perform the flickering adversarial attacks on single videos, as demonstrated in Figure 1, we have selected a random sub-sample from the validation section of the kinetics dataset and solve Equation (1) for each one of the selected sample ($N = 1$). Table 1 shows the results of a single-video attack using $\beta_1 = \beta_2 = 0.5$, reaching 100% fooling ratio with low roughness and thickness values. Video examples of the attack can be found here[1]. Section A.2l contains a detailed description of the convergence process regarding this attack.

4.3.1. Thickness Vs. Roughness

In Figure 2, we see the temporal amplitude of the adversarial perturbation of each frame and for each color channel, respectively. The extreme case of minimizing only D_1 (given success of the untargeted adversarial attack) and leaving D_2 unconstrained (β_1 = 1, β_2 = 0) is presented at the top. The signal of the RGB channels fluctuates strongly with a thickness value of 0.87% and a roughness of 1.24%. The other extreme case is when D_2 is constrained and D_1 is not (β_1 = 0, β_2 = 1), leading to a thickness value of 1.66% and a roughness value of 0.6%. The central image displays all the gradual cases between the two extremities: β_1 goes from 1 to 0, and β_2 from 0 to 1 on the y-axis. The row denoted by β_1 = 0 corresponds to the upper graph and the row denoted by β_1 = 1 corresponds to the lower graph. Both D_1 and D_2 are very dominant in the received perturbation, as desired. Visualization of the path taken by our loss mechanisms at different β_1 and β_2 values can be found in section A.2.1.

4.4. Adversarial Attack Generalization

In order to obtain universal and class generalization of adversarial perturbation across videos we solve the optimization problem in Equation (1).

4.4.1. Class generalization: Untargeted Attack

Adversarial attack on a single video has limited applicability in the real world. In this section we have generalized the attack to cause misclassification to any videos from a specific class with a single generalized adversarial pattern. The results presented were averaged on 100 random classes out of 400. The training was as described in previous sections with the main difference of training using the training-split of each class and validating using the validation-split of the

[3] https://www.tensorflow.org/

[4] https://github.com/deepmind/kinetics-i3d

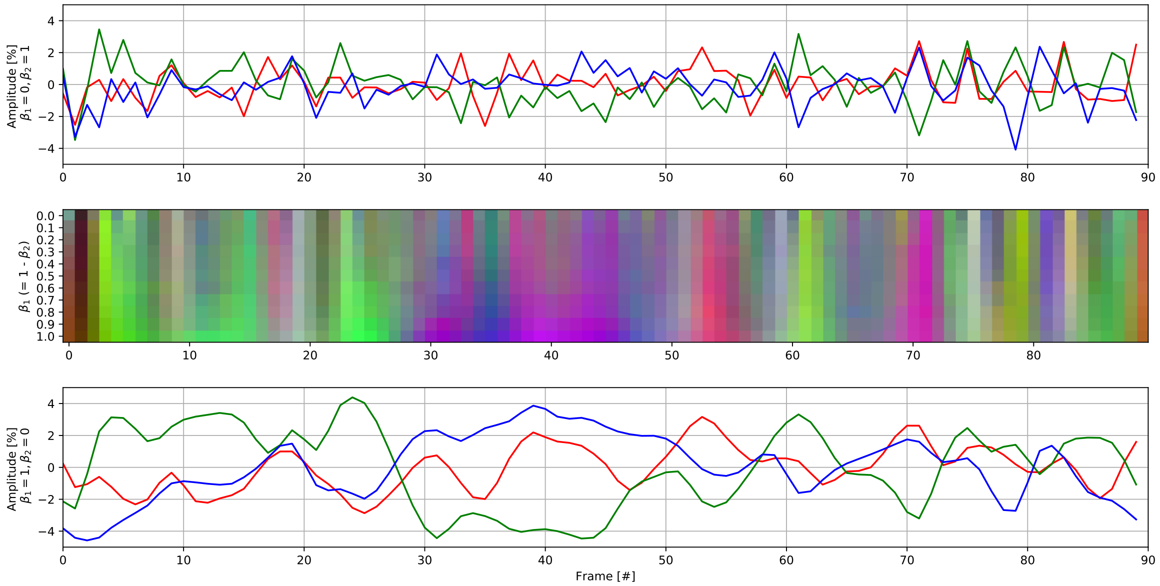

Figure 2. Top: The adversarial perturbation of the RGB channels (color represents relevant channel) as a function of the frame number at the case that $\beta_1 = 0$ and $\beta_2 = 1$ (D_2 minimization is preferred). Bottom: The adversarial perturbation of the RGB channels as a function of the frame number at the case that $\beta_1 = 1$ and $\beta_2 = 0$ (D_1 minimization is preferred). Top and bottom graphs are presented in percents from the full scale of the image. Middle: The gradual change of the adversarial pattern between the two extreme cases where $\beta_1 = 0$ corresponds to the top graph and $\beta_1 = 1$ corresponds to the bottom graph. Color (stretched for visualization purposes) represents the RGB parameters of the adversarial pattern of each frame.

Table 1. Results and standard deviation over several types of attacks

ATTACK	TIME INVARIANCE	FOOLING RATIO[%]	THICKNESS[%]	ROUGHNESS[%]
SINGLE VIDEO	×	100	1.0±0.5	0.83± 0.4
SINGLE CLASS	×	90.2± 11.72	13.0± 3.6	10.6± 2.2
UNIVERSAL	×	93.0	15.5	15.7
UNIVERSAL	√	83.1	18.0	14.0

same class in the kinetics dataset (Kay et al., 2017). The initially miss-classified videos within the dataset were removed in order to start with zero miss-ratio. Table 1 shows the statistics of success ratio when β_1=$\beta_2 = 0.5$. It is obvious that generalization produces an adversarial pattern with larger thickness and roughness. When applying this pattern, in average 90.2% of the videos from each class were misclassified. Selecting different parameters to the regularization terms can produce different adversarial patterns.

4.4.2. UNIVERSAL UNTARGETED ATTACK

We take one step toward real world implementation of the flickering attack by devising a universal perturbation that will attack videos from any class. Constructing such flickering attacks is not trivial due to the small number of trainable parameters. Training an adversarial perturbation requires a large video database since the free parameters space is very large: each pixel of each channel of each frame. In comparison, the number of free parameters of the flickering perturbation is relatively small and the training requires a modest amount of videos in order to generalize the attack. This advantage adds greatly to the implementability of this attack in the real world. In order to demonstrate this, we apply **Sparse Universal Perturbations** (SUP) attack (Wei et al., 2019) adapted to our settings. The original SUP (Wei et al., 2019) attacks a CNN+RNN architecture as threat model and experiment on the UCF101 dataset (Soomro et al., 2012). We choose the non-mask version s.t. $\delta_{SUP} \in \mathbb{R}^{T\times H\times W\times C}$ has in total $T \times H \times W \times C$ ($13M$ with I3D) parameters to optimize, while our attack has $T \times C$ (270 with I3D) parameters. The SUP applied on the I3D (Carreira & Zisserman, 2017) architecture and the Kinetics-400 (Kay et al., 2017) dataset while using the same loss terms from (Wei et al., 2019) and the adversarial loss. The training set defined as the entire evaluation-split ($20K$ videos) of the Kinetics-400 and the validation is a random sub-sample of $5K$ videos from the test-split. Before evaluation, each misclassified video was removed from the dataset, providing an initial fooling ratio of 0%. Figure 3 demonstrates our series of experiments on flickering and SUP attacks, by showing the fooling ratio as a function of the number of videos used for training. Our attack reaches a 60% fooling ratio with only 50 videos to train on, while SUP stays close to 0%. Furthermore, the attack introduced in (Wei et al., 2019) reaches a slightly higher fooling ratio over our attack with the current setup (95.1% vs 93%), but it requires 10 times more training videos. The adversarial pattern presented in this paper requires only 500 videos to reach a 90% fooling ratio, where the other attack requires 5000 for the same ratio. One might implement the universal flickering attack as a class-targeted attack using the presented method. In this case, the selected class may affect the efficiency of the adversarial perturbation.

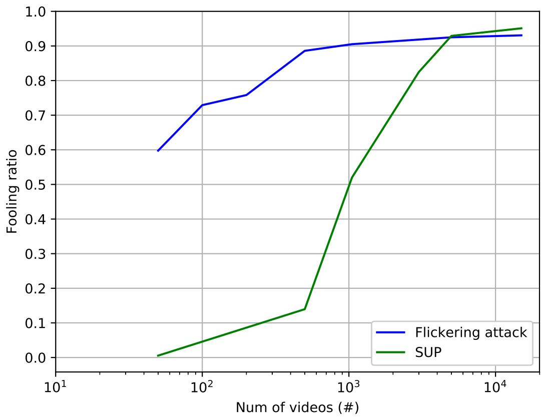

Figure 3. Fooling ratio Vs. number of videos in the training set. In order to reach 90% fooling ratio, the SUP attack requires almost 10 times more videos in the training set than the flickering attack.

4.5. Time Invariance

Practical implementation of adversarial attacks on video classifiers can not be subjected to prior knowledge regarding the frame numbering or temporal synchronization of the attacked video. In this section we present a time-invariant adversarial attack that can be applied to the recorded scene without assuming that the perturbation of each frame is applied at the right time. This time-invariant attack can be projected to the scene in a cyclic repeating manner, and regardless of any arbitrary frame the user would select as a first frame, the adversarial pattern would be effective. Similar to the generalized adversarial attacks described in previous subsections, a random shift between the adversarial pattern and the model input was applied during training. The adversarial perturbation in Equation (1) modified by adding the $Roll$ operator defined in Section 3.6 s.t. $F_\theta(X_n + Roll(\delta, \tau))$ for randomly sampled $\tau \in \{1, 2, \cdots, T\}$ in each iteration and on each video during training and evaluation. This time invariance generalization of universal adversarial flickering attack reaches 83% fooling ratio, but this is a small price to pay in order to approach real-world implementability. In order to avoid performance penalty due to temporal interlacing between the camera's integration time and frame rate and the adversarial pattern, it would be beneficial to increase the smoothness regularization of the adversarial pattern, in accordance with the specific parameters of the application at hand.

5. Conclusions and future work

The flickering adversarial attack was presented for the first time, over several scenarios summarized in Table 1. This attack has several benefits, such as the relative unperceptability to the human observer, achieved by small and smooth perturbations as can be seen in the videos we have posted[1]. For example, in the single video attack, the adversarial pattern adds only $0.4 \pm 0.3\%$ in average to the roughness of the unperturbed videos, which is almost half of the roughness of the average perturbation itself, for in some cases the addition of the perturbation reduces the total roughness. In addition, this adversarial pattern is probably the most applicable video attack of any adversarial perturbation thus far. This is due to the simplicity and uniformity of the perturbation across the frame. The flickering perturbation can be implemented in real world scenarios since it does not require a complex spatial adversarial pattern to be projected upon the scene, but a simple temporal one. This can be achieved by subtle lighting changes to the recorder scene by small illumination changes.

In extreme cases where regularization causes the pattern to be thick enough to be unnoticed by human observers, the usage of such perturbation can be relevant for non-man-in-the-loop systems or cases where the human observer will see image-flickering without realizing that the system is being fooled. In the future, we will expand the current research to include adversarial attacks with black box setting.

A. Appendix

A.1. Improved Adversarial loss function

For achieving a more stable convergence, we have developed a new loss mechanism. Similar to (Carlini & Wagner, 2016), our loss reaches the adversarial goal only to the desired extent, leaving space for other regularization terms. For untargeted attack:

$$\ell(y,t) = \max\left(0, \min\left(\frac{1}{m}\ell_m(y,t)^2, \ell_m(y,t)\right)\right) \qquad (6)$$

$$\ell_m(y,t) = y_t - \max_{i\neq t}(y_i) + m. \qquad (7)$$

$m > 0$ is the desired margin of the original class probability below the adversarial class probability. When loss values are within the desired margin, the quadratic loss term relaxes the relatively steep gradients and momentum of the optimizer, and the difference between the first and second class probabilities approach the desired margin m. When the loss starts rising, the quadratic term gently maintains the desired difference between these two classes, therefore preventing overshoot effects. In order to apply the suggested mechanism on targeted attack, the loss term changed to $\ell_m(y,t) = \max_{i\neq t}(y_i) - y_t + m$, while this time, t is the targeted adversarial class.

In some cases it would be beneficial to follow (Carlini & Wagner, 2016) and use the logits instead of the probabilities for calculating the loss. We suggest adapting this method partially by keeping the desired margin in probability space, normalized at each iteration accordingly, for margin defined in logit space may be less intuitive as a regularization term.

A.2. Convergence Process

In the learning process of the improved loss mechanism several trends can be observed (Figure 4). At first, the adversarial perturbation rises in thickness and roughness. At iteration 40 the top-probability class switches from the original to the adversarial class, which until now was not plotted, for this adversarial attack is untargeted. At that iteration, the adversarial loss is m. When the difference between the probability of the adversarial and original class is larger then m the adversarial loss is zero and the regularization starts to be prominent, causing the thickness and roughness to decay. This change of trend occurs slightly after the adversarial class change due to the momentum of the Adam optimizer and remaining intrinsic gradients. At iteration 600 the difference between the probability of the adversarial and original class is $m = 0.05$, the quadratic loss term maintaining the desired difference between these classes while diminishing the thickness and roughness. The binary loss changes at the interface between adversarial success and failure caused convergence issues, and the implementation of the quadratic term, as defined in Equation (6) handled this issue.

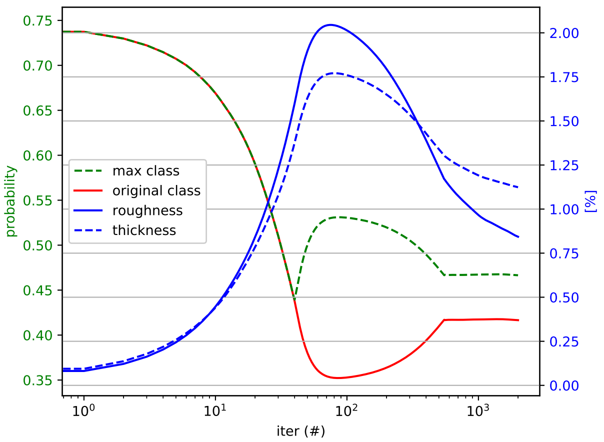

Figure 4. Learning process of the improved loss mechanism. Probabilities (green and red lines) corresponds to the left y-scale. Roughness and thickness (blue lines) are in percents from the full gray-level range of the image (right y-scale). *original class* is the probability of the actual class of the unperturbed video. *max class* is the probability of the most probable class as the classifier predicts.

A.2.1. Thickness Vs. Roughness

In order to visualize the path taken by our loss mechanisms at different β_1 and β_2, we have plotted a 3D representation in probability-thickness-roughness space for 10 different experiments (10 different single video attack). Figure 5 shows the probability of the maximal class at 10 different scenarios as described in the legend. One can see that at the beginning the maximal probability (original class) drops from the initial probability (upper section of the graph) on the same path for all of the described cases, until the adversarial perturbation takes hold of the top class. From there, the β's parameters takes the lead. At this point, each different case is converging along a different path to a different location on the thickness-roughness plane. The user may choose the desired ratios for each specific application.

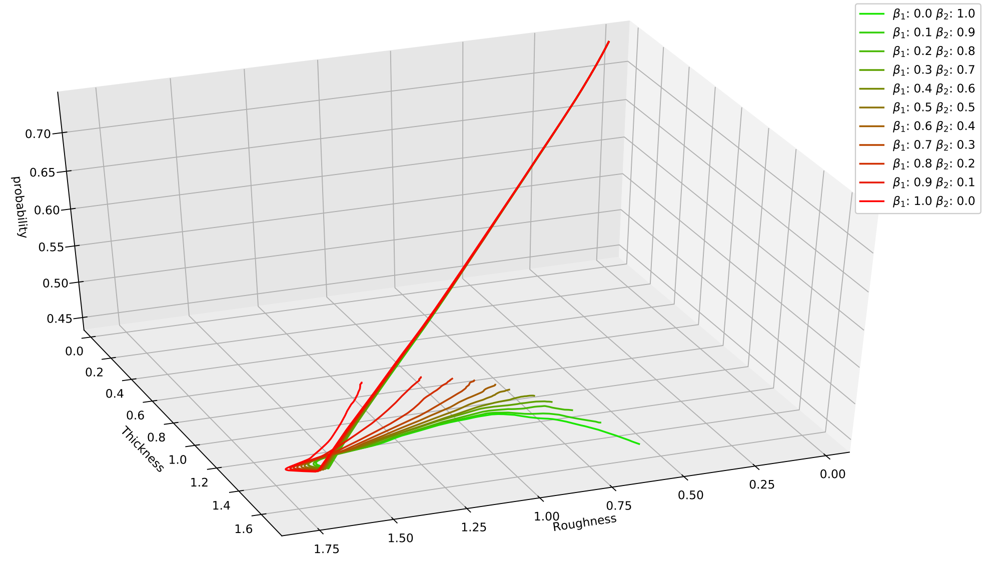

Figure 5. Convergence curve in probability-thickness-roughness space of an untargeted adversarial attack with different β_1 and β_2 parameters.

References

Carlini, N. and Wagner, D. A. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016.

Carreira, J. and Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. *CoRR*, abs/1705.07750, 2017.

Feichtenhofer, C., Fan, H., Malik, J., and He, K. Slowfast networks for video recognition. *CoRR*, abs/1812.03982, 2018.

Goodfellow, I., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015a.

Goodfellow, I., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015b.

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., and Brox, T. Flownet 2.0: Evolution of optical flow estimation with deep networks. *CoRR*, abs/1612.01925, 2016.

Inkawhich, N., Inkawhich, M., Chen, Y., and Li, H. Adversarial attacks for optical flow-based action recognition classifiers. *CoRR*, abs/1811.11875, 2018.

Ji, S., Xu, W., Yang, M., and Yu, K. 3d convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35:221–231, 2013.

Jiang, L., Ma, X., Chen, S., Bailey, J., and Jiang, Y.-G. Black-box adversarial attacks on video recognition models. *ArXiv*, abs/1911.09449, 2019.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., and Zisserman, A. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. Hmdb: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pp. 2556–2563, 2011.

Kurakin, A., Goodfellow, I. J., and Bengio, S. Adversarial machine learning at scale. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Li, S., Neupane, A., Paul, S., Song, C., Krishnamurthy, S. V., Roy-Chowdhury, A. K., and Swami, A. Stealthy adversarial perturbations against real-time video classification systems. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*, 2019.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Snchez, C. I. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42: 60 – 88, 2017. ISSN 1361-8415.

Papernot, N., McDaniel, P. D., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. The limitations of deep learning in adversarial settings. *CoRR*, abs/1511.07528, 2015.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28*, pp. 91–99. Curran Associates, Inc., 2015.

Rey-de-Castro, R. and Rabitz, H. Targeted nonlinear adversarial perturbations in images and videos. *CoRR*, abs/1809.00958, 2018.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. S., Berg, A. C., and Li, F. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

Sak, H., Senior, A. W., and Beaufays, F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, pp. 338–342, 2014.

Shelhamer, E., Long, J., and Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017.

Simonyan, K. and Zisserman, A. Two-stream convolutional networks for action recognition in videos. *CoRR*, abs/1406.2199, 2014.

Soomro, K., Zamir, A. R., and Shah, M. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

Sultani, W., Chen, C., and Shah, M. Real-world anomaly detection in surveillance videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Sun, Y., Chen, Y., Wang, X., and Tang, X. Deep learning face representation by joint identification-verification. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 1988–1996. Curran Associates, Inc., 2014.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Tran, D., Bourdev, L. D., Fergus, R., Torresani, L., and Paluri, M. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014.

Varol, G., Laptev, I., and Schmid, C. Long-term temporal convolutions for action recognition. *CoRR*, abs/1604.04494, 2016.

Wang, L., Xu, Y., Cheng, J., Xia, H., Yin, J., and Wu, J. Human action recognition by learning spatio-temporal features with deep neural networks. *IEEE Access*, 6:17913–17922, 2018.

Wang, X., Girshick, R. B., Gupta, A., and He, K. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2017.

Wei, X., Zhu, J., Yuan, S., and Su, H. Sparse adversarial perturbations for videos. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 8973–8980, 2019.

Zhipeng Wei, Jingjing Chen, X. W. L. J. T.-S. C. F. Z. Y.-G. J. Heuristic black-box adversarial attacks on video recognition models. *ArXiv*, abs/1911.09449, 2019.