Adversarial Attacks on Black Box Video Classifiers: Leveraging the Power of Geometric Transformations

Anonymous Author(s)
Affiliation
Address
email

Abstract

When compared to the image classification models, black-box adversarial attacks against video classification models have been largely understudied. This could be possibly because, with video, the temporal dimension poses significant additional challenges in gradient estimation. Query-efficient black-box attacks rely on effective estimated gradients towards maximizing the probability of misclassifying the target video. In this work, we demonstrate that such effective gradients can be searched for by parameterizing the temporal structure of the search space with geometric transformations. Specifically, we design a novel iterative algorithm Geometric TRAnsformed Perturbations (GEO-TRAP), for attacking video classification models. GEO-TRAP employs standard geometric transformation operations to reduce the search space for effective gradients into searching for a small group of parameters that define these operations. This group of parameters describes the geometric progression of gradients, resulting in a reduced and structured search space. Our algorithm inherently leads to successful perturbations with surprisingly few queries. For example, adversarial examples generated from GEO-TRAP have better attack success rates with $\sim 73.55\%$ fewer queries compared to the state-of-the-art method for video adversarial attacks on the widely used Jester dataset. Overall, our algorithm exposes vulnerabilities of diverse video classification models and achieves new state-of-the-art results under black-box settings on two large datasets.

1 Introduction

Adversarial attacks are designed to expose vulnerabilities of Deep Neural Networks (DNNs). With real-world applications of video classification based on DNNs emerging [1, 2], a key question that arises is "*what type of adversarial inputs can mislead, and thus render video classification networks vulnerable*?" Designing such adversarial attacks not only helps expose security flaws of DNNs, but can also potentially stimulate the design of more robust video classification models.

Adversarial attacks against image classification models have been studied in both *white-box* [3–7] and *black-box* [8–12] settings. In the white-box setting, an adversary has full access to the model under attack, including its parameters and training settings (hyper-parameters, training data, etc.) In the black-box setting, an adversary only has partial information about the victim model, such as the predicted labels of the model. In the case of video classification models, adversarial attacks in both white-box and black-box settings have garnered some interest [13–21], although the body of work here is more limited than the case of image classification models.

A common black-box attack paradigm is query-based, wherein the attacker can send queries to the victim model to collect the corresponding predicted labels, and thereby estimate the gradients needed for curating the adversarial examples. Unlike static images, videos naturally include additional information from the temporal dimension. This high dimensionality (i.e., sequence of frames instead

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Table 1: **Comparison with state-of-the-art.** GEO-TRAP, compared to current black-box attack methods for videos, doesn't train a different network to craft perturbations, and parameterizes the temporal dimension of videos in searching for effective perturbation directions.

Methods	WITHOUT training a "perturbation" network	CONSIDER temporal dimension?	PARAMETERIZE temporal dimension?
PATCHATTACK[18]	✗	✗	✗
HEURISTICATTACK [19]	✓	✗	✗
SPARSEATTACK [20]	✗	✗	✗
MOTION-SAMPLER ATTACK [21]	✓	✓	✗
GEO-TRAP (**Ours**)	✓	✓	✓

of one image) poses challenges to black-box adversarial attacks against video classification models; in particular, significantly more queries are typically needed for estimating the gradients for crafting adversarial samples [18–21]. [18] reduces the number of queries by adding perturbations on the patch level instead of at the pixel level; [19, 20] propose to add perturbations only on key pixels. [21] considers the intrinsic differences between images and videos (i.e., the temporal dimension), and proposes to use the optical-flow of clean videos as the motion prior for adversarial video generation. Similar to [21], we also explicitly consider the temporal dimension of video. However, rather than fixing the temporal search space using the motion prior of clean videos, we propose to parameterize the temporal structure of the space with geometric transformations. This results in a better structured and reduced search space, which allows us to generate successful attacks with much fewer queries in black-box settings than the state-of-the-art methods, including [21].

Contributions. In this paper, we propose a novel query-efficient black-box attack algorithm against video classification models. Due to the extra temporal dimension, generating video perturbations by searching for effective gradients remains a challenging task given the exceedingly large search space. These gradients are estimated by searching for 'directions' that maximize the probability of the victim model mis-classifying the crafted inputs. Our approach drastically reduces this large search space by defining this space with a small set of parameters that describe the geometric progression of gradients in the temporal dimension, resulting in a reduced and temporally structured search space. Conceptually, this parameterization of the temporal structure of the search space is performed using geometric transformations (e.g. affine transformations). We refer to our algorithm as Geometrically TRAnsformed Perturbations, or GEO-TRAP. Despite this surprisingly simple strategy, GEO-TRAP outperforms existing black-box video adversarial attack methods by significant margins ($\sim 1.8\%$ improvement in attack success rate with $\sim 73.55\%$ fewer queries for targeted attacks in comparison to the state-of-the-art [21] on the Jester dataset [22]).

2 Related Works

In this section, we review different black-box adversarial attacks strategies, and categorize our proposed method with respect to state-of-the-art black-box attacks designed for video classifiers.

In most real-world attacks, the adversary only has partial information about the victim models, such as the predicted labels. In such black-box settings, the adversary can first attack a local surrogate model and then transfer these attacks to the target victim model [23, 24], formally called as *transferability-based* black-box attack. Alternatively, they may estimate the adversarial gradient with zero-order optimization methods such as Finite Differences (FD) or Natural Evolution Strategies (NES) by querying the victim model [11, 12, 25], which is called *query-based* black-box attack. GEO-TRAP falls under the category of query-based black-box attacks (designed for videos).

Whilst several white-box attacks have been proposed for video classification models [13–17], black-box video attacks are relatively under explored. PATCHATTACK (V-BAD)[18] is the first to propose a black-box video attack framework which uses a hybrid attack strategy of first generating initial perturbations for each video frame by attacking a local image classifier, and then updating the perturbations by querying the victim model. Compared to PATCHATTACK[18] , GEO-TRAP does not require training a local classifier. PATCHATTACK[18] crafts video perturbations by treating each frame as a separate image, but reduces the search space of the gradient estimation by morphing the perturbations in patches/partitions. However, its attack performance has been shown to be inferior to that of a more recent approach [21] (discussed below). HEURISTICATTACK [19] uses a query-

based attack strategy, and reduces the search space by generating adversarial perturbations only on heuristically selected key frames and salient regions. SPARSEATTACK [20] reduces the search space by adding perturbations only on key frames using a reinforcement learning based framework. MOTION-SAMPLER ATTACK [21] proposed a query-based attack strategy that utilizes a motion excited sampler to obtain *motion-aware* perturbation prior by using the optical-flow of the clean video. This motion-aware prior reduces the search space for gradients resulting in fewer queries. Similar to [21] but different from [18–20], GEO-TRAP explicitly considers the temporal dimension of video in order to search for effective gradients. However unlike [21], GEO-TRAP does not fix the temporal structure of the search space using a *pre-computed fixed* motion prior, but parameterize it with simple geometric transformations. These black-box video attack methods are summarized in Table 1.

3 Attacking via Geometrically TRAnsformed Perturbations (GEO-TRAP)

Notation. We denote the tuple of a video clip and its corresponding label as $(\boldsymbol{x}, y)$, which represents a data-point in the distribution $\mathcal{X}$. Each video sample $\boldsymbol{x} \in \mathbb{R}^{T\times H\times W\times C}$ has T frames of H height, W width, and C channels. We denote the victim video classification model as $\boldsymbol{f_\theta} : \mathcal{X} \rightarrow \mathcal{Y}$, where $\boldsymbol{\theta}$ represents the model's parameters learned from the training subset of $\mathcal{X}$, via a mapping to the label space $\mathcal{Y}$. We further assume $\mathcal{X}$ consists of videos from $|\mathcal{Y}| = K$ categories. To make the perturbations imperceptible to humans, we impose the perturbation budget $\rho_{\max}$ with the $\|\cdot\|_p$ norm. Throughout this paper, we consider $\|\cdot\|_\infty$ norm following [14, 18, 21] (the method can be extended to $p = 1, 2$ norms). To constrain $\|\cdot\|_\infty$ of perturbation below a budget $\rho_{\max}$, we use the $\texttt{clip}(\cdot)$ function to keep the perturbation pixel value in $[-\rho_{\max}, \rho_{\max}]$. The function $\texttt{sign}(\cdot)$ extracts the sign of given input variable. The superscript i, throughout the paper, denotes the iteration i. The subscript t denotes the frame index. For clarity, we represent vectors/tensors with the bold font and scalars with the regular font.

Problem Statement. We consider the scenario of attacking a standard video classification model using a query-based paradigm under **black-box settings** (assuming no access to $\boldsymbol{\theta}$ nor the training subset of $\mathcal{X}$). Specifically, we aim to craft perturbed videos $\boldsymbol{x}_{\text{adv}}$ with imperceptible differences from $\boldsymbol{x}$, in order to alter the decision of the target model $\boldsymbol{f_\theta}$ via multiple queries to guide the gradient estimation. This problem can be mathematically formulated as follows.

$$\underset{\boldsymbol{x}_{\text{adv}}}{\text{argmin}} \quad \mathcal{L}\big(\boldsymbol{f_\theta}(\boldsymbol{x}_{\text{adv}}), y\big) \quad \text{s.t.} \quad \|\boldsymbol{x}_{\text{adv}} - \boldsymbol{x}\|_\infty \leq \rho_{\max} \tag{1}$$

$\mathcal{L}\big(\boldsymbol{f_\theta}(\boldsymbol{x}_{\text{adv}}), y\big)$ is the objective function, capturing the similarity between the classifier's output and the ground truth label y, and varies with different attack goals (targeted or untargeted). The challenge is to obtain $\boldsymbol{x}_{\text{adv}}$ with as few queries as possible by estimating gradient $\boldsymbol{g}^\star = \nabla_{\boldsymbol{x}_{\text{adv}}}\mathcal{L}\big(\boldsymbol{f_\theta}(\boldsymbol{x}_{\text{adv}}), y\big)$, which is unknown in the considered black-box setting.

Overview of GEO-TRAP. We propose a novel iterative video perturbation framework that follows the principle of the Basic Iterative Method [26] in order to fool $\boldsymbol{f_\theta}$ under $\|\cdot\|_\infty$ norm as follows.

$$\boldsymbol{x}^{(0)}_{\text{adv}} = \boldsymbol{x}, \quad \boldsymbol{x}^{(i)}_{\text{adv}} = \texttt{clip}\big(\boldsymbol{x}^{(i-1)}_{\text{adv}} - h\,\texttt{sign}(\boldsymbol{g}^{(i)})\big) \tag{2}$$

where h is a hyperparameter and $\boldsymbol{g}^{(i)}$ is the gradient estimated by querying the black-box victim model at the ith iteration using our proposed GEO-TRAP algorithm. As shown in (2), effective perturbations rely on the guidance of the gradient $\boldsymbol{g}^{(i)}$. Therefore, efficiently estimating $\boldsymbol{g}^{(i)}$ is at the core of GEO-TRAP for successfully subverting video classifiers. We execute the following two steps in each iteration to estimate $\boldsymbol{g}^{(i)}$.

1. For any input video $\boldsymbol{x}^{(i)}$, a random noise tensor $\boldsymbol{r}_{\text{frame}} \in \mathbb{R}^{H\times W\times C}$ and a set of geometric transformation parameters $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times D}$ are chosen with each element sampled from a standard normal distribution. D represents the number of parameters needed for the geometric transformation of a single frame (details are provided in Section 3.2). In this setup, our search space for estimating $\boldsymbol{g}^{(i)}$ consists of $\boldsymbol{r}_{\text{frame}}$ and $\boldsymbol{\Phi}_{\text{warp}}$.
2. We then warp $\boldsymbol{r}_{\text{frame}}$ with $\boldsymbol{\Phi}_{\text{warp}}$ to get the candidate direction $\boldsymbol{\pi} = [\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots, \boldsymbol{r}_T] \in \mathbb{R}^{T\times H\times W\times C}$ (see Algorithm 2.1 TRANS-WARP). $\boldsymbol{\pi}$ is then employed to compute a gradient estimator $\boldsymbol{\Delta}$ by querying the black-box victim model with a standard gradient estimation algorithm (see Algorithm 2 GRAD-EST). The gradient estimator $\boldsymbol{\Delta}$ is then used to update $\boldsymbol{g}^{(i)}$.

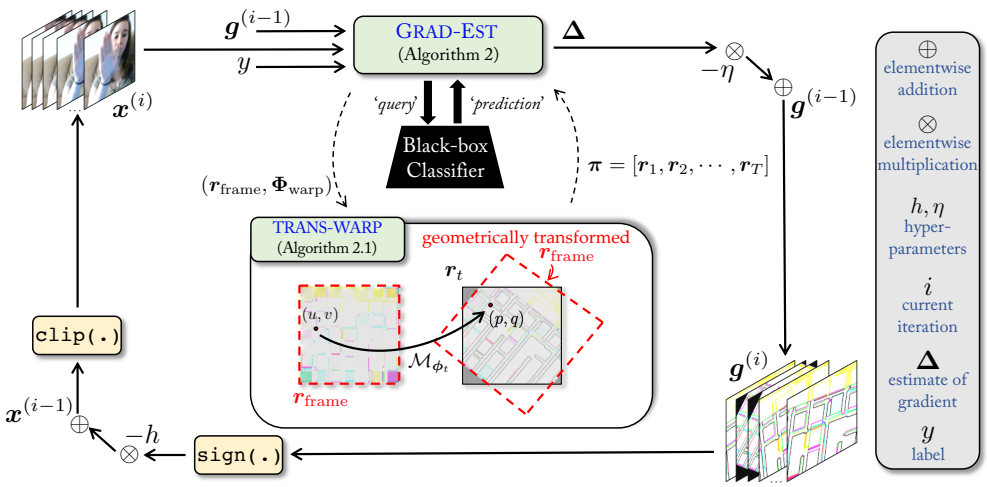


Figure 1: **Overview of Geo-Trap.** Geo-TRAP is a black-box attack algorithm guided by the key observation that strong gradients $\boldsymbol{g}^{(i)}$ can be computed by finding better gradient search direction candidates $\boldsymbol{\pi}$. We propose to search each frame of the directions $\boldsymbol{r}_t$ by warping a randomly sampled $\boldsymbol{r}_{\text{frame}}$ using a geometric transformation $\mathcal{M}_{\phi_t}$; different $\boldsymbol{r}_t$ in $\boldsymbol{\pi}$ are warped by the same $\boldsymbol{r}_{\text{frame}}$, thus have geometric progression among frames.

The overall attack strategy is summarized in Algorithm 1 and pictorially illustrated in Figure 1. Since in Step 2 above, the gradient estimation (i.e., GRAD-EST) procedure includes the geometric transformation strategy (i.e., TRANS-WARP), we will next describe GRAD-EST and then move on to TRANS-WARP. For the simplicity of exposition, we drop the superscript i and shorten the loss function to $\mathcal{L}(\boldsymbol{x}_{\text{adv}}, y)$ to $\mathcal{L}$ (since the model parameters $\boldsymbol{\theta}$ remain unchanged) in rest of this section.

3.1 GEO-TRAP Gradient Estimation (GRAD-EST)

Let $\boldsymbol{g}^\star = \nabla_{\boldsymbol{x}}\mathcal{L}$ be the ideal value of the gradient of $\mathcal{L}$ at $\boldsymbol{x}$, required to create $\boldsymbol{x}_{\text{adv}}$ in (2). To find an efficient estimator $\boldsymbol{g}$ for $\boldsymbol{g}^\star$, a (new) surrogate loss $\ell(\boldsymbol{g}) = -\langle \boldsymbol{g}^\star, \boldsymbol{g}\rangle$ is defined such that the estimator $\boldsymbol{g}$ has a sufficiently large inner product with the actual gradient $\boldsymbol{g}^\star$ ($\boldsymbol{g}$ is normalized to a unit vector; we ignore the normalization operation for ease of explanation). The loss function definition and the algorithm to estimate $\boldsymbol{g}$ follow [12].

As $\boldsymbol{g}^\star$ is unknown in the black-box setting, this surrogate loss function can be estimated as

$$\ell(\boldsymbol{g}) = -\langle \boldsymbol{g}^\star, \boldsymbol{g}\rangle = -\langle \nabla_{\boldsymbol{x}}\mathcal{L}, \boldsymbol{g}\rangle \approx -\frac{\mathcal{L}(\boldsymbol{x}+\epsilon\boldsymbol{g}, y) - \mathcal{L}(\boldsymbol{x}, y)}{\epsilon}. \tag{3}$$

To iteratively estimate $\boldsymbol{g}$, we need to, in turn, estimate the gradient of $\ell(\boldsymbol{g})$, i.e., $\boldsymbol{\Delta} = \nabla_{\boldsymbol{g}}\ell(\boldsymbol{g})$. With antithetic sampling [27], $\boldsymbol{\Delta}$ can be estimated as

$$\boldsymbol{\Delta} = \frac{\ell(\boldsymbol{g}+\delta\boldsymbol{\pi}) - \ell(\boldsymbol{g}-\delta\boldsymbol{\pi})}{\delta}\boldsymbol{\pi}, \tag{4}$$

where δ is a small number adjusting the magnitude of the loss variation and $\boldsymbol{\pi} \in \mathbb{R}^{T\times H\times W\times C}$ is a random candidate direction. Our core contribution lies in the fact that instead of randomly sampling $\boldsymbol{\pi}$ in the search space [12], we reduce the search dimensionality by warping a randomly sampled tensor $\boldsymbol{r}_{\text{frame}} \in \mathbb{R}^{H\times W\times C}$ with another randomly sampled geometric (e.g., affine) transformation parameter tensor $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times D}$ to get $\boldsymbol{\pi}$. The search space is then reduced from $T\times H\times W\times C$ to $(H\times W\times C)+(T\times D)$ and D is a relatively small number, $D \ll H\times W\times C$. With $\boldsymbol{w}_1 = \boldsymbol{g}+\delta\boldsymbol{\pi}$ and $\boldsymbol{w}_2 = \boldsymbol{g}-\delta\boldsymbol{\pi}$ and combining (3) with (4), we get

$$\boldsymbol{\Delta} = \frac{\mathcal{L}(\boldsymbol{x}+\epsilon\boldsymbol{w}_2, y) - \mathcal{L}(\boldsymbol{x}+\epsilon\boldsymbol{w}_1, y)}{\epsilon\delta}\boldsymbol{\pi}. \tag{5}$$

Note that by querying the victim model $\boldsymbol{f}_{\boldsymbol{\theta}}$ with $\boldsymbol{x}+\epsilon\boldsymbol{w}_1$, we are able to retrieve the value of $\mathcal{L}(\boldsymbol{x}+\epsilon\boldsymbol{w}_1, y)$; similarly we can obtain the value of $\mathcal{L}(\boldsymbol{x}+\epsilon\boldsymbol{w}_2, y)$ ($\mathcal{L}(\cdot)$ is defined following [16].). In summary, we estimate $\boldsymbol{\Delta}$ with these two queries to the victim model. The resulting algorithm for estimating gradient of $\nabla_{\boldsymbol{g}}\ell$ or $\boldsymbol{\Delta}$ for consequently estimating $\boldsymbol{g}$ is shown in Algorithm 2. Eventually at every iteration, we use $\boldsymbol{\Delta}$ to update $\boldsymbol{g}$ by applying a one-step gradient descent as $\boldsymbol{g} \leftarrow \boldsymbol{g} - \eta\boldsymbol{\Delta}$, where η is a hyperparameter to update $\boldsymbol{g}$. This updated $\boldsymbol{g}$ is later used to obtain $\boldsymbol{x}_{\text{adv}}$ using (2).

Algorithm 1: GEO-TRAP: Query-based Iterative attack for Video Classifiers

Input : video $\boldsymbol{x}$, corresponding label y, step-size η for updating the gradient, step-size h for updating adversarial video.

Output : adversarial video $\boldsymbol{x}_{\text{adv}}$

Initialize: $\boldsymbol{x}^{(0)} = \boldsymbol{x}$, $\boldsymbol{g}^{(0)} = \mathbf{0}$, $i = 1$

while $\operatorname{argmax}\left(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})\right) = y$ **do**

  $\boldsymbol{\Delta} = \texttt{GRAD-EST}(\boldsymbol{x}^{(i-1)}, \boldsymbol{g}^{(i-1)}, y)$ /* Gradient Estimation */

  $\boldsymbol{g}^{(i)} \leftarrow \boldsymbol{g}^{(i-1)} - \eta\boldsymbol{\Delta}$

  $\boldsymbol{x}^{(i)} \leftarrow \texttt{clip}(\boldsymbol{x}^{(i-1)} - h\texttt{sign}(\boldsymbol{g}^{(i)}))$

  $i \leftarrow i + 1$

end

return $\boldsymbol{x}_{\text{adv}} = \boldsymbol{x}^{(i)}$

3.2 Noise Warping using Geometric Transformation (TRANS-WARP)

To tackle the challenge of the high-dimensionality of the search space, we propose to parameterize the search space with a single random noise tensor $\boldsymbol{r}_{\text{frame}} \in \mathbb{R}^{H\times W\times C}$ and a sequence of geometric transformations $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times D}$. Apart from the reduction of the search space of gradient estimation, our geometric transformation provides a temporal structure to $\boldsymbol{\pi}$, which we discuss next.

At every iteration, $\boldsymbol{\pi} = [\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_T]$ represents the candidate direction for $\boldsymbol{\Delta}$. These directions $\boldsymbol{r}_t \in \mathbb{R}^{H\times W\times C}$ are used to compute $\boldsymbol{\Delta}$ in order to update gradient $\boldsymbol{g}$. To obtain $\boldsymbol{\pi}$, we use a sequence of transformation vectors $\boldsymbol{\Phi}_{\text{warp}} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_T]$ where $\boldsymbol{\phi}_t \in \mathbb{R}^D$. The dimensionality D, chosen by the attacker, can vary depending on the transformation type that is populated from $\boldsymbol{\phi}_t$, *e.g.*, $D = 6$ for affine transformation. We take affine transformation as an example to describe the warping process. We start by randomly sampling $\boldsymbol{r}_{\text{frame}}$ and the sequence of $\boldsymbol{\phi}_t$ along with initializing each element in the sequence of $\boldsymbol{r}_t$ with zero in **every** iteration. TRANS-WARP then computes $\boldsymbol{r}_t$ by warping $\boldsymbol{r}_{\text{frame}}$ using the parameters in $\boldsymbol{\phi}_t = [\phi^t_{11}, \phi^t_{12}, \phi^t_{13}, \phi^t_{21}, \phi^t_{22}, \phi^t_{23}] \in \mathbb{R}^6$ of $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times 6}$ as follows. For all C channels, let (p, q) and (u, v) be the target and source coordinates in $\boldsymbol{r}_t$ and $\boldsymbol{r}_{\text{frame}}$, respectively. $\boldsymbol{r}_t$ (for all channels) is computed as

$$\boldsymbol{r}_t(p, q) \leftarrow \boldsymbol{r}_{\text{frame}}(u, v), \quad 1 \le p, u \le H,\ 1 \le q, v \le W. \tag{6}$$

Location (p, q) is computed using the affine transform matrix $\mathcal{M}_{\boldsymbol{\phi}_t}$ created with ϕ_t in homogeneous coordinates [28] as shown below. t is dropped for simplicity.

$$\begin{pmatrix} p \\ q \\ 1 \end{pmatrix} = \mathcal{M}_{\boldsymbol{\phi}} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{7}$$

We compactly denote this warping operation in (6) and (7) with $\boldsymbol{r}_t = \mathcal{T}(\boldsymbol{r}_{\text{frame}}, \boldsymbol{\phi}_t)$. Affine transformation allows translation, rotation, scaling, and skew to be applied to $\boldsymbol{r}_{\text{frame}}$ to get each $\boldsymbol{r}_t$. Therefore, the sequence of $\boldsymbol{r}_t$ have affine geometric progression among its temporal dimension. Other examples of geometric transformations may be more constrained, such as the similarity transformation $\mathcal{M}^{\text{S}}_{\boldsymbol{\phi}}$ (that allows translation, dilation (uniform scale) and rotation with $D = 4$) and translation-dilation $\mathcal{M}^{\text{TD}}_{\boldsymbol{\phi}}$ (that allows translation and uniform dilation with $D = 3$) as shown below.

$$[\phi_{11}, \phi_{12}, \phi_{13}, \phi_{23}] \rightarrow \mathcal{M}^{\text{S}}_{\boldsymbol{\phi}} = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ -\phi_{12} & \phi_{11} & \phi_{23} \\ 0 & 0 & 1 \end{bmatrix}, [\phi_{11}, \phi_{13}, \phi_{23}] \rightarrow \mathcal{M}^{\text{TD}}_{\boldsymbol{\phi}} = \begin{bmatrix} \phi_{11} & 0 & \phi_{13} \\ 0 & \phi_{11} & \phi_{23} \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

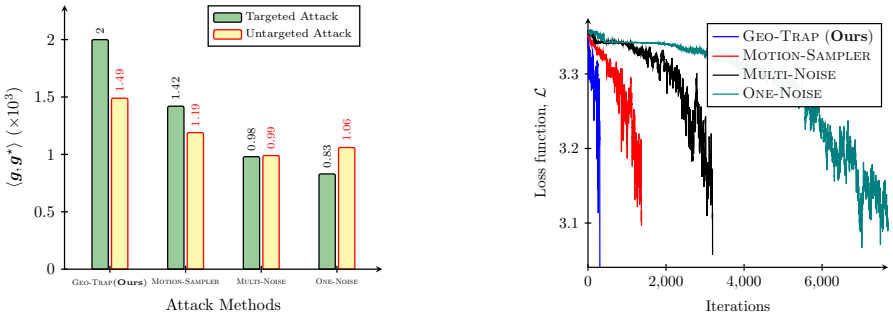
4 What Makes GEO-TRAP Effective?

Potent iterative algorithms should rely on few queries for crafting successful perturbations for time efficiency. To minimize the number of queries, iterative algorithms need to find strong gradients in their early iterations. As discussed earlier, videos inherently incur a larger search space due to the temporal dimension and thus, pose challenges in searching for effective gradients. In this section, we provide empirical evidence to show that by parameterizing the temporal dimension, GEO-TRAP finds better gradients, in general, than previous works. We use three baselines in this analysis.

Algorithm 2: GRAD-EST($\boldsymbol{x}^{(i-1)}, \boldsymbol{g}^{(i-1)} \in \mathbb{R}^{T\times H\times W\times C}, y$)$\rightarrow$ Estimate $\boldsymbol{\Delta} = \nabla_{\boldsymbol{g}}\ell(\boldsymbol{g}) \in \mathbb{R}^{T\times H\times W\times C}$

Input : video $\boldsymbol{x}^{(i)}$, label y, gradient estimator $\boldsymbol{g}^{(i-1)}$, δ for loss variation, ϵ for approximation.
Output : estimation of $\boldsymbol{\Delta} = \nabla_{\boldsymbol{g}}\ell(\boldsymbol{g})$

Sample $\boldsymbol{r}_{\text{frame}} \in \mathbb{R}^{H\times W\times C}$, $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times D}$ (each element from a normal distribution $\mathcal{N}(0,1)$)
$\boldsymbol{\pi} = \texttt{TRANS-WARP}(\boldsymbol{r}_{frame}, \boldsymbol{\Phi}_{warp})$ `/* Use Geometric Transformations */`
$\boldsymbol{w}_1 = \boldsymbol{g}^{(i-1)} + \delta\boldsymbol{\pi}$, $\boldsymbol{w}_2 = \boldsymbol{g}^{(i-1)} - \delta\boldsymbol{\pi}$
$L_1 = \mathcal{L}(\boldsymbol{x}^{(i-1)} + \epsilon\boldsymbol{w_2}, y)$, $L_2 = \mathcal{L}(\boldsymbol{x}^{(i-1)} + \epsilon\boldsymbol{w_1}, y)$ `/* Query victim model twice */`
$\boldsymbol{\Delta} = (L_2 - L_1)\boldsymbol{\pi}/\epsilon\delta$
return $\boldsymbol{\Delta}$



(a) (b)

Figure 2: **Gradient Analysis of GEO-TRAP. (a)** GEO-TRAP's high query-efficiency is a direct implication of good quality gradient estimation (for both targeted and untargeted attack), shown here with higher cosine similarity with $\boldsymbol{g}^\star$ compared to other methods. **(b)** Better quality of estimated gradients by GEO-TRAP results in a successful attack with fewer queries compared to other attacks.

- MULTI-NOISE ATTACK [12] which computes search directions $\boldsymbol{r}_t$ separately for each frame by sampling each element of $\boldsymbol{r}_t$ from a standard normal distribution, resulting in a search space dimension of $T \times H \times W \times C$. It does not explicitly consider the temporal dimension; temporal progression in any arbitrary direction is possible between a sequence of perturbation frames.
- ONE-NOISE ATTACK which computes $\boldsymbol{r}_1$ by sampling each element from a standard normal distribution and applies the same $\boldsymbol{r}_1$ across all $\boldsymbol{r}_t(t = 1, 2, \cdots, T)$. ONE-NOISE ATTACK reduces the search space but completely ignores the temporal dimension when generating the perturbation.
- MOTION-SAMPLER ATTACK [21] which uses the optical flow of the original video $\boldsymbol{x}$ to warp $\boldsymbol{r}_{\text{frame}}$ to get each $\boldsymbol{r}_t$. It reduces the search space by using the motion prior of $\boldsymbol{x}$ as the temporal progression between perturbation frames. In contrast, rather than fixing the temporal search space using a motion prior, GEO-TRAP parameterizes the temporal structure of the space with $\boldsymbol{\Phi}_{\text{warp}}$.

We measure the gradient estimation quality by calculating the cosine similarity between the ground truth $\boldsymbol{g}^\star$ and the estimated gradient $\boldsymbol{g}$ following [18] for the aforementioned baselines. For each attack, we average over 1000 randomly selected videos with their cosine similarity values in the first attack iteration. We choose the first iteration because the initial $\boldsymbol{g}^\star$ is the same for the different attack methods, ensuring a fair comparison. As shown in Figure 2(a), our proposed method for estimating the gradients, yields $\boldsymbol{g}$ of the best quality for both untargeted and targeted attacks among all evaluated approaches. This leads to faster loss convergence / few queries as shown in Figure 2(b). We validate such trends with different loss functions and more datasets in the Supplementary Material.

The empirical results validate that by carefully considering the temporal dimension and parameterizing the temporal structure of the gradient search space with geometric transformations, GEO-TRAP finds better gradients. GEO-TRAP and MOTION-SAMPLER ATTACK [21] are better than the other two; the reason could be that temporally structured perturbations are more likely to disrupt the motion context of videos. However, the gradients estimated by MOTION-SAMPLER ATTACK [21] are not as effective as our proposed approach; the reason could be that the motion-prior of the clean video does not necessarily represent the temporal behavior of effective video perturbations. By allowing flexibility of the temporal progression while maintaining only a minimally sufficient space through

Algorithm 2.1: TRANS-WARP($\boldsymbol{r}_{\text{frame}} \in \mathbb{R}^{H\times W\times C}$, $\boldsymbol{\Phi}_{\text{warp}} \in \mathbb{R}^{T\times D}$)$\rightarrow$ Estimate $\boldsymbol{\pi} \in \mathbb{R}^{T\times H\times W\times C}$

Input : noise tensor $\boldsymbol{r}_{\text{frame}}$, warp tensors $\boldsymbol{\Phi}_{\text{warp}}$, transformation operation $\mathcal{T}_{\boldsymbol{\phi}}(\cdot)$.
Output : candidate directions $\boldsymbol{\pi} = [\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots, \boldsymbol{r}_T]$.

Initialize $\boldsymbol{\pi} = \emptyset$
for $t = [1, 2, \cdots, T]$ **do**
  $\boldsymbol{\phi}_t = \boldsymbol{\Phi}_{\text{warp}}[t]$
  $\boldsymbol{r}_t = \mathcal{T}(\boldsymbol{r}_{\text{frame}}, \boldsymbol{\phi}_t)$ /* Warping Operation */
  $\boldsymbol{\pi} \leftarrow$ append $\boldsymbol{r}_t$
end
return $\boldsymbol{\pi}$

its geometric parameterization, GEO-TRAP generates effective temporally structured perturbations. Note that one could use other, potentially better ways to parameterize the temporal progression of the video perturbation; this is regarded as future works.

5 Experiments

Datasets. Following previous work like [14], we use the human action recognition dataset UCF-101 [29] and the hand gesture recognition dataset 20BN-JESTER (Jester) [22] to validate our attacks. *UCF-101* includes 13320 videos from 101 human action categories (e.g., applying lipstick, biking, blow drying hair, cutting in the kitchen). Given the diversity it provides, we consider the dataset to validate the feasibility of our attacks on *coarse-grained* actions. *Jester*, on the other hand, includes hand gesture videos that are recorded by crowd-sourced workers performing 27 kinds of gestures (e.g., sliding hand left, sliding two fingers left, zooming in with full hand, zooming out with full hand). The appearance of different hand gestures is similar; it is the motion information that matters in the video classification. We use this dataset to validate our attack with regard to *fine-grained* actions.

Baselines. Among the four state-of-the-art black-box video attack methods [18–21] described in Section 2, we use [20, 21] as baselines for following reasons. Our first baseline is MOTION-SAMPLER ATTACK [21], which has been shown to outperform PATCHATTACK[18] , ONE-NOISE and MULTI-NOISE attacks (introduced in Section. 4). Our second baseline is HEURISTICATTACK [19]. We note that SPARSEATTACK [20] is not included in our analysis as we couldn't replicate their results.

Attack Settings. We consider four state-of-the-art video classification models representing diverse methodologies of learning from videos, i.e., C3D [30], SlowFast [31], TPN [32] and I3D [33], as our black-box victim models to attack. More details about the four video models are provided in the Supplementary Materials. For UCF-101, we randomly select one video from each category following the setting in [18, 21]. For Jester, since the number of categories is small, we randomly select four videos from each category. All attacked videos are correctly classified by the black-box model. For targeted attack, a random target class is chosen for each video. The maximum noise value $\rho_{\max}$ is 10 pixel values (out of 255) following [14, 34, 35]. We provide more results for different $\rho_{\max}$ in Supplementary Material. Note that since the perturbation generated by HEURISTICATTACK [19] is sparse and thus more imperceptible, we do not impose a perturbation budget on it. We set the maximum query limit to $Q = 60,000$ for untargeted attack and $Q = 200,000$ for targeted attack. The other hyper-parameters, i.e., ϵ, δ, η, and h take the same values as mentioned in [21]. Unless otherwise specified, a transformation-dilation transformation (with $D = 3$) is used for our query-based attack. We provide the implementation of GEO-TRAP in the Supplementary Material as `code.zip`.

Metrics. Following [18, 21], we evaluate GEO-TRAP, in terms of *(a)* Success Rate (SR), i.e., the total success rate in attacking within query and perturbation budgets; and *(b)* Average Number of Queries (ANQ) i.e., the average total queries from attacks for all videos (including failed ones).

5.1 Comparison to State-of-the-Art

Untargeted Attack. We report the untargeted attack performance of our attack method and the baseline methods in Table 2. We observe that, in general, GEO-TRAP requires fewer average number of queries when attacking different black-box victim models: on average over 45 % fewer queries than MOTION-SAMPLER ATTACK [21]. At the same time, GEO-TRAP yields higher attack success rates: on average about 6% higher than HEURISTICATTACK [19]. When attacking SlowFast model

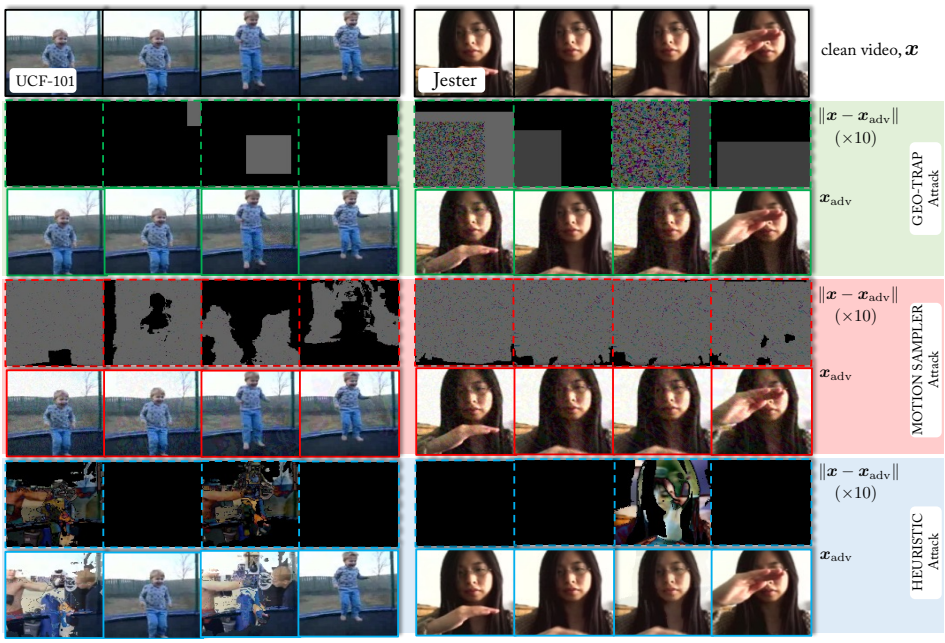


Figure 3: **Visualization of Perturbations and Perturbed Video.** We visualize the generated perturbations and perturbed video for GEO-TRAP and other baselines for UCF-101 (*left*) and Jester (*right*) datasets for untargeted attack against SlowFast classifier with $\rho_{\max} = {}^{10}/_{255}$.

on the Jester dataset, GEO-TRAP achieves 100% successful rate with only 521 queries while the baseline methods need at least 1906 queries. We also observe that the TPN model is more robust towards black-box attacks compared with the other three video recognition models.

Visualization. We show two visualizations of adversarial frames on Jester and UCF-101 in Fig. 3. We observe that the generated adversarial frames have little difference from the clean ones but can lead to a failed classification. Also, our attack method could lead to *sparse* perturbations in the spatial and temporal dimension as the perturbations are sometimes zoomed out (thus get very small), and sometimes are translated out of the sight with choice of geometric transformation. More examples are in the Supplementary Material.

Targeted Attack. We report the targeted attack performance of our method and the baseline methods in Table. 3. We observe that in some cases, HEURISTICATTACK [19] requires fewer number of queries than GEO-TRAP, but its attack success rates are pretty low in those cases. For example, when attacking the TPN model on the Jester dataset, although HEURISTICATTACK [19] requires only 12k average number of queries, its attack success rate is less than half of ours, 44.4% v.s. 92.6%. GEO-TRAP consistently yields higher attack success rates, on average over 30% higher than HEURISTICATTACK [19] and over 8% higher than MOTION-SAMPLER ATTACK [21]. In addition, in most cases, GEO-TRAP requires fewer average number of queries than the two baseline attacks, on average over 45 % fewer queries than MOTION-SAMPLER ATTACK [21]. The targeted attack performance further validates the effectiveness of our method.

Table 2: **Untargeted Attacks**. GEO-TRAP demonstrates highly successful untargeted attacks (high Success Rate (SR)) with fewer queries (low Average Number of Queries (ANQ))

Datasets	Methods	Black-box Video Classifiers							
		C3D		SlowFast		TPN		I3D	
		ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)
Jester	HEURISTICATTACK [19]	4699	99.0%	3572	98.1%	4679	82.0%	4248	98.1%
	MOTION-SAMPLER ATTACK [21]	4549	99.0%	1906	100%	6269	91.3%	3029	99.4%
	GEO-TRAP (**Ours**)	**1602**	**100%**	**521**	**100%**	**3315**	**92.4%**	**1599**	**100%**
UCF-101	HEURISTICATTACK [19]	**5206**	70.2%	3507	87.2%	**6539**	71.8%	6949	84.7%
	MOTION-SAMPLER ATTACK [21]	14336	81.6%	4673	97.2%	20369	75.8%	7400	94.4%
	GEO-TRAP (**Ours**)	11490	**86.2%**	**1547**	**98.8%**	17716	**76.1%**	**4887**	**97.4%**

Table 3: **Targeted Attacks**. GEO-TRAP demonstrates highly successful targeted attacks (high Success Rate (SR)) with fewer queries (low Average Number of Queries (ANQ))

Datasets	Methods	Black-box Video Classifiers							
		C3D		SlowFast		TPN		I3D	
		ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)	ANQ (↓)	SR (↑)
Jester	HEURISTICATTACK [19]	15595	46.3%	30768	98.1%	**12006**	44.4%	31088	77.8%
	MOTION-SAMPLER ATTACK [21]	26704	98.2%	33087	100%	63721	80.9%	39037	90.7%
	GEO-TRAP (**Ours**)	**6198**	**100%**	**7788**	**100%**	41294	**92.6%**	**19542**	**98.2%**
UCF-101	HEURISTICATTACK [19]	**26741**	29.0%	22152	61.4%	**71828**	36.4%	92244	43.7%
	MOTION-SAMPLER ATTACK [21]	100467	71.1%	57126	86.0%	151409	31.6%	96498	59.6%
	GEO-TRAP (**Ours**)	71820	**85.8%**	**21878**	**95.0%**	141629	**40.0%**	**76708**	**74.6%**

5.2 Different Geometric Transformations in TRANS-WARP

As discussed in Section 3.2, different kinds of geometric transformations could be used in the TRANS-WARP function. In addition to the translation-dilation transformation ($\mathcal{M}_\phi^{\text{TD}}$ in (8), $D = 3$) employed throughout the paper, we report the performance of GEO-TRAP with two other different geometric transformations, i.e., similarity transformation ($\mathcal{M}_\phi^{\text{S}}$ in (8), $D = 4$) and affine transformation ($\mathcal{M}_\phi$ in (7), $D = 6$). Figure 4 shows the untargeted attack performance on Jester with these different geometric transformations (more results are available in the Supplementary Material). We observe that the transformation with fewer degrees of freedom (DOF) (translation-dilation transformation) tends to require fewer queries while having the same or higher attack success rates (the attack success rates are available in the Supplementary Material). We believe that $D = 3$ provides enough temporal flexibility to disrupt the motion context of the videos; additional degrees of freedom seemingly increase the search space unnecessarily, resulting in more queries.

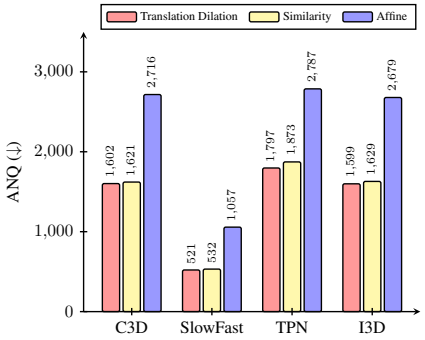


Figure 4: **Performance with different $\mathcal{M}_\phi$.** GEO-TRAP results in best performance when $\mathcal{M}_\phi$ is set as translation-dilation operation.

6 Conclusion

Black-box adversarial attacks on video classifiers is a challenging problem that has been largely understudied. In this work, we demonstrate that searching for effectual gradients in a reduced but structured search space for crafting perturbations leads to highly successful attacks with fewer queries compared to state-of-the-art attack strategies. In particular, we propose a novel iterative algorithm that employs Geometric transformations to parameterize and reduce the search space, for estimating gradients that maximize the probability of mis-classification of the perturbed video. This simple and novel strategy exposes the vulnerability of widely used video classification models. For instance, GEO-TRAP decreases average query numbers by 64.78%, 72.66% and 47.21% to attack C3D, SlowFast, and I3D, respectively, for close to a 100% success rate in untargeted attacks.

7 Broader Impact

In this work, by leveraging geometric transformations for effective gradient estimations, we propose a highly query-efficient adversarial attack on video classification models which demonstrates state-of-the-art results. As more and more safety-critical systems (e.g., perceptual modules in autonomous vehicles) nowadays rely on video models, we are hopeful that our work, in addition to future research, can eventually help build sufficiently robust video models to best avoid malicious sub-versions. On one hand, we believe our algorithm could allow further research in adversarial robustness and data augmentation strategies of deep vision models. It should also give a direction to researchers to design counter defense methodologies. On the other hand, it highlights a key drawback of different video classifiers which will allow adversaries to design more sophisticated attacks, both in white-box and black-box settings. Addressing such fallacies in designing deep neural networks is of utmost importance before introducing them in real-world scenarios.

References

[1] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for automatic human action recognition, January 1 2013. US Patent 8,345,984.

[2] Wongun Choi, Yuanqing Lin, Yu Xiang, and Silvio Savarese. Subcategory-aware convolutional neural networks for object detection, May 8 2018. US Patent 9,965,719.

[3] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing Properties of Neural Networks. *arXiv preprint arXiv:1312.6199*, 2013.

[4] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv preprint arXiv:1412.6572*, 2014.

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[6] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *12th {USENIX} Workshop on Offensive Technologies ({WOOT} 18)*, 2018.

[7] Chaowei Xiao, Ruizhi Deng, Bo Li, Fisher Yu, Mingyan Liu, and Dawn Song. Characterizing adversarial examples based on spatial consistency information for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–234, 2018.

[8] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

[9] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.

[10] Arjun Nitin Bhagoji, Warren He, Bo Li, and Dawn Song. Practical black-box attacks on deep neural networks using efficient query mechanisms. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 154–169, 2018.

[11] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26, 2017.

[12] Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *arXiv preprint arXiv:1807.07978*, 2018.

[13] Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8973–8980, 2019.

[14] Shasha Li, Ajaya Neupane, Sujoy Paul, Chengyu Song, Srikanth V Krishnamurthy, Amit K Roy-Chowdhury, and Ananthram Swami. Stealthy adversarial perturbations against real-time video classification systems. In *NDSS*, 2019.

[15] Zhikai Chen, Lingxi Xie, Shanmin Pang, Yong He, and Qi Tian. Appending adversarial frames for universal video attack. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3199–3208, 2021.

[16] Roi Pony, Itay Naeh, and Shie Mannor. Over-the-air adversarial flickering attacks against video recognition networks. *arXiv preprint arXiv:2002.05123*, 2020.

[17] Shao-Yuan Lo and Vishal M Patel. Multav: Multiplicative adversarial videos. *arXiv preprint arXiv:2009.08058*, 2020.

[18] Linxi Jiang, Xingjun Ma, Shaoxiang Chen, James Bailey, and Yu-Gang Jiang. Black-box adversarial attacks on video recognition models. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 864–872, 2019.

[19] Zhipeng Wei, Jingjing Chen, Xingxing Wei, Linxi Jiang, Tat-Seng Chua, Fengfeng Zhou, and Yu-Gang Jiang. Heuristic black-box adversarial attacks on video recognition models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12338–12345, 2020.

[20] Huanqian Yan, Xingxing Wei, and Bo Li. Sparse black-box video attack with reinforcement learning. *arXiv preprint arXiv:2001.03754*, 2020.

[21] Hu Zhang, Linchao Zhu, Yi Zhu, and Yi Yang. Motion-excited sampler: Video adversarial attack with sparked prior. In *European Conference on Computer Vision*. Springer, 2020.

[22] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The jester dataset: A large-scale video dataset of human gestures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[23] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *ICLR*, 2017.

[24] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4733–4742, 2019.

[25] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pages 2137–2146. PMLR, 2018.

[26] Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world, 2017.

[27] Hongyu Ren, Shengjia Zhao, and Stefano Ermon. Adaptive antithetic sampling for variance reduction. In *International Conference on Machine Learning*, pages 5420–5428. PMLR, 2019.

[28] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[29] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[30] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[31] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6202–6211, 2019.

[32] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal pyramid network for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 591–600, 2020.

[33] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

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

[35] Konda Reddy Mopuri, Utkarsh Ojha, Utsav Garg, and R Venkatesh Babu. Nag: Network for adversary generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 742–751, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] *The main claim we made is that we propose a query-efficient black-box video attack method by parameterizing the temporal structure of the video perturbation, which is the paper's contribution and scope.*
   (b) Did you describe the limitations of your work? [Yes] *We discussed in Section 5 that using geometric transformation to parameterize the temporal structure of the video perturbation is just one way of the many potential temporal parameterizations. Better ways of parameterizations could be explored in the future.*
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] *We discussed in Section 8 that our work could possibly be employed by adversaries to launch attacks on real-world applications and to design more sophisticated attacks.*
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] *We provide the python implementation of our method with instruction file in the supplementary material as `code.zip`. The datasets we used are public ones and how we use them are explained in the experimental section.*
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] *We provide all such details in the experimental section.*
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] *We include the error bar results in the Supplementary Material.*
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] *Our attack is query-based attack and the deep neural network computation happens in the victim black-box systems. The attacker side runs codes that calculate the equations described in the paper. No GPU is needed.*
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] *We use existing datasets and video classification models and we cite the creators.*
   (b) Did you mention the license of the assets? [N/A] *No licenses are involved.*
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] *We are including the code of the proposed attack method in the Supplemental Material.*
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] *We are using publicly available data.*
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] *We discuss in the Supplementary Material that the datasets we used do not contain personally identifiable information and offensive contents.*
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]