

Just One Moment: Structural Vulnerability of Deep Action Recognition against One Frame Attack

Jaehui Hwang, Jun-Hyuk Kim, Jun-Ho Choi, and Jong-Seok Lee

School of Integrated Technology, Yonsei University, Korea

{jaehui.hwang, junhyuk.kim, idearibosome, jong-seok.lee}@yonsei.ac.kr

Abstract

The video-based action recognition task has been extensively studied in recent years. In this paper, we study the structural vulnerability of deep learning-based action recognition models against the adversarial attack using the one frame attack that adds an inconspicuous perturbation to only a single frame of a given video clip. Our analysis shows that the models are highly vulnerable against the one frame attack due to their structural properties. Experiments demonstrate high fooling rates and inconspicuous characteristics of the attack. Furthermore, we show that strong universal one frame perturbations can be obtained under various scenarios. Our work raises the serious issue of adversarial vulnerability of the state-of-the-art action recognition models in various perspectives.

1. Introduction

Human action recognition using videos has been extensively studied in recent years thanks to the development of deep network-based algorithms based on the abundance of computational resources and data [6]. From a network design perspective, the distinguished main research topic of action recognition is how to model temporal information residing in video clips. Concerning this, various attempts have been made, such as utilizing the long short-term memory (LSTM) module [3] or the optical flow [1], but recently, 3D convolutional neural network (CNN)-based action recognition models are widely used. To improve the performance and efficiency of 3D CNN-based action recognition models, various mechanisms in the temporal dimension have been proposed, such as frame selection [4] and convolutional operations [1, 22, 20].

Many researchers have found the vulnerability of deep learning-based algorithms against so-called adversarial attacks, which add an inconspicuous perturbation to input data to mislead a target model to produce wrong output. It has been reported that many state-of-the-art deep image

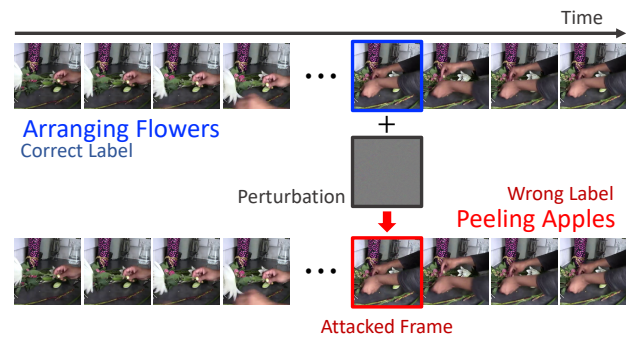


Figure 1: Overall illustration of one frame attack targeting a vulnerable frame. With only one attacked frame, the target action recognition model wrongly classifies a given video in Kinetics-400 [8] as “peeling apples” instead of “arranging flowers.”

classification methods are highly vulnerable to the adversarial attacks [16] and can raise severe security concerns [5]. On the other hand, there are not many studies on the vulnerability of video-based deep action recognition systems.

We argue that the ways of modeling temporal information in deep models for action recognition have significant impact on the vulnerability of the models, which we denote as *structural vulnerability*. Although there exist a few attempts for adversarial attacks on action recognition systems [23, 14], they do not sufficiently consider the structural vulnerability of recent action recognition models. As a result, they do not fully satisfy two criteria that a successful adversarial attack should meet: 1) achieving a high fooling rate and 2) keeping adversarial perturbation invisible to conceal that the video clip is attacked. Wei *et al.* [23] proposed an attack method of perturbing only a few frames of a video clip, in order to reduce computational resources and achieve inconspicuousness. However, this method targets only LSTM-based models and relies on the particular property of LSTM, i.e., temporal propagation of information. Thus, it does not achieve a high fooling rate on the latest CNN-based action recognition models (see Table 1). On

the other hand, Pony *et al.* [14] proposed an attack method to find a sequence of flickering perturbation that changes the overall color of a given video clip over time. However, it is known that artifacts changing in the temporal dimension are more detectable than spatial artifacts by human observers [13, 24]. Therefore, this adversarial attack does not sufficiently satisfy the second criterion, inconspicuousness (see Figure 7).

In this paper, we discover the structural vulnerability of recent CNN-based deep action recognition models, which has not been explored previously to the best of our knowledge. Using this vulnerability, we also show that perturbation in just a single vulnerable frame of a video clip can significantly degrade the accuracy of deep action recognition models, as illustrated in Figure 1. The attacked frame is shown only for 33 or 40 milliseconds when the target video clip has 30 or 25 frames per second (FPS), which is hardly perceivable to human observers. The main contributions of this work can be summarized as follows.

- We investigate the vulnerability caused by the structural property of deep models using three state-of-the-art CNN-based deep action recognition models and examine what factors of these models make them highly vulnerable against adversarial attacks. We show that the efforts to efficiently model temporal information induce the vulnerability issue.
- We show the possibility of so-called *one frame attack* on action recognition models in a white-box attack scenario. When only one vulnerable frame found by our analysis is perturbed with a gradient-based adversarial attack method, this perturbation can easily defeat deep learning-based action recognition systems. This one frame attack can fool the state-of-the-art video-based action recognition models with fooling rates of almost 100%. In addition, this adversarial attack is inconspicuous, which is demonstrated via a subjective experiment.
- We further explore video-agnostic universal perturbation based on the one frame attack. We show that the universal perturbation, which is found from a small number of videos, can affect other input video clips with high fooling rates. Besides, the one frame attack can be effectively applied to time-invariant scenarios where the perturbation is added to the input video clip with an unknown temporal offset.

2. Related work

2.1. Action recognition

Recently, the performance of action recognition has been significantly improved, along with the development of deep

neural networks [1, 3, 4, 7, 15, 19, 20, 22]. In the early attempts, the CNN+LSTM structure [3] achieved high performance by integrating two-dimensional convolutional layers, which has been popularly employed in image-related tasks, and a LSTM model targeting sequence data. On the other hand, three-dimensional (3D) convolutional layers that utilize features in both spatial and temporal dimensions have been proposed [7, 19]. Another approach to deal with videos is to employ two CNNs simultaneously (known as two-stream networks), where they process original RGB frames (to exploit spatial features) and their optical flows (to exploit temporal features), respectively [15]. These two approaches are sometimes combined to further improve the performance of action recognition [1]. In recent years, more advanced deep action recognition models have been developed. One of the widely used approaches is an inflated three-dimensional (I3D) network [1, 22], which is a fine-tuned version of a pre-trained image classification model by inflating two-dimensional kernels of the convolutional layers to three-dimensional. The two-stream approach also has been extended to a method named SlowFast [4], which takes video data having different temporal resolutions (i.e., frame rates) on each of the stream networks as inputs. Another trend in recent researches on action recognition is to employ kernel factorization (e.g., interaction-reduced channel-separated network (ir-CSN) [20]) for reducing computational complexity.

2.2. Adversarial attack

It has been shown that deep learning-based image classification models are highly vulnerable to adversarial attacks under a white-box scenario. Szegedy *et al.* [18] proposed an optimization-based attack method to minimize the amount of input perturbation that can change the classification result of a given model. Goodfellow *et al.* [5] developed the fast gradient sign method (FGSM), which calculates perturbation from the sign of the gradients obtained from a given model. Kurakin *et al.* [9] extended FGSM to an iterative approach, which is called iterative FGSM (I-FGSM) and showed higher fooling rates of the attack than FGSM. While those methods find perturbation in the whole region of a given input image, Su *et al.* [17] showed the feasibility of one-pixel attack, which tries to find perturbation of only one pixel to fool deep image classifiers.

The vulnerability of deep learning models has been further evaluated via several advanced methods beyond finding perturbation for each input image. Liu *et al.* [11] investigated transferability of perturbation, which is to examine whether a perturbation found for a model can also work for another model. Moosavi-Dezfooli *et al.* [12] found image-independent universal perturbation that can be applied to any images to fool a given classifier.

2.3. Adversarial attack on action recognition

There are a few studies on white-box adversarial attack of action recognition models. Li *et al.* [10] developed an adversarial attack for the convolutional 3D (C3D) model [19] by employing a generative adversarial network (GAN). Wei *et al.* [23] proposed an optimization-based method to generate adversarial perturbation for LSTM-based models. Pony *et al.* [14] developed a method that changes the overall color of each frame in a given video clip to obtain a flickering perturbation. However, these methods have been verified only on traditional action recognition models. Furthermore, they add perturbation to multiple frames of a given video, which may be visible to human observers.

3. Analysis of structural vulnerability

In this section, we show the existence of the structural vulnerability of action recognition models. For this, a single frame of a video sequence is perturbed by I-FGSM [9] and uniform random noise, and the recognition performance is examined for each frame. Then, we analyze what factors cause such vulnerability.

3.1. Analysis methods

Using I-FGSM. We use I-FGSM [9] to perturb a frame, which is one of the widely used strong adversarial attack methods.

This method iteratively finds a perturbation of the i -th frame in a video clip as follows. Let $X^0 = \{X^0(1), X^0(2), \dots, X^0(T)\}$ denote the original video clip (having T frames) that is correctly classified as y by a target action recognition model $M(\cdot)$, i.e., $M(X^0) = y$. The attack aims to find an attacked version of a video clip X from X^0 , where only the i -th frame $X^0(i)$ is changed to an attacked one $X(i)$ containing inconspicuous perturbation. To find $X(i)$, the I-FGSM update rule is applied, by finding the adversarial frame at iteration $n + 1$, $X^{n+1}(i)$, from the frame at the previous iteration, $X^n(i)$, iteratively:

$$\tilde{X}^{n+1}(i) = \text{Clip}_{0,255} \left(X^n(i) + \frac{\epsilon}{N} \text{sgn}(\nabla_{X^n(i)} J(X^n, y)) \right), \quad (1)$$

$$X^{n+1}(i) = \text{Clip}_{-\epsilon, \epsilon}(\tilde{X}^{n+1}(i) - X^0(i)) + X^0(i), \quad (2)$$

where ϵ regulates the amount of perturbation to be added, $\text{sgn}(\cdot)$ is the sign function, $\nabla_{X^n(i)} J(X^n, y)$ is the gradient of the target frame for the loss function $J(X^n, y)$, and

$$\text{Clip}_{a,b}(X) = \min(\max(X, a), b). \quad (3)$$

After N iterations, the final adversarial video clip is obtained by $X = X^N = \{X^0(1), X^0(2), \dots, X^N(i), X^0(i) +$

$1), \dots, X^0(T)\}$. We expect that the model outputs a wrong prediction when X is inputted (i.e., $M(X) \neq y$).

Using uniform random noise. We use uniform random noise within $[-64, 64]$ as perturbation, which is injected to a certain frame. This type of perturbation is tested in order to understand the structural vulnerability. Furthermore, since generating random noise perturbation is computationally efficient, it can be used to identify vulnerable frame indices for a given action recognition model.

3.2. Experimental setup

Dataset and models. We use Kinetics-400 [8], which is one of the widely used large-scale benchmark datasets for action recognition. From the test set of Kinetics-400, we randomly choose ten videos for each class. Therefore, a total of 4000 videos are chosen to evaluate the fooling rates of the attack methods. As target action recognition models, we consider three state-of-the-art models having various model structures, including I3D [22], SlowFast [4], and ir-CSN [20]. These models have shown outstanding recognition performance on the Kinetics-400 dataset. We employ the pre-trained models available on MMAction2 [2], which is the open-sourced repository that provides testing tools for the aforementioned action recognition models. Among the variants of the SlowFast models, we use the 8×8 SlowFast in the implementation of MMAction2.

Implementation details We conduct the I-FGSM method with various hyperparameters. For attacking a single frame, we set the number of iterations to $N \in \{30, 50, 100\}$ and the amount of perturbation to $\epsilon \in \{2, 4, 8, 16\}$. We only report the case of $N = 30$, which empirically found to be sufficient to attack the target models.

3.3. Vulnerability of action recognition models

Figures 2 and 3 show the fooling rates of the two types of perturbation for the three models. Surprisingly, we observe the existence of vulnerable frame indices (or, shortly, vulnerable frames) showing significantly higher fooling rates than the others, especially in the cases of I3D and SlowFast. We also confirm that those vulnerable frames are observed periodically. Specifically, I3D and SlowFast have vulnerable frames at $i \in \{3, 7, 11, 15, 19, 23, 27, 31\}$ and $i \in \{1, 5, 9, 13, 17, 21, 25, 29\}$, respectively. The ir-CSN model does not show such a trend. Compared to the other two models, it exhibits relatively high vulnerability overall, with relatively small variations between frames. These observations hold consistently across different values of ϵ .

Among the vulnerable frames, the most vulnerable frames for I3D, SlowFast, and ir-CSN are the 31st, 29th, and 1st frames, respectively. It can be seen that even by adding uniform random noise, the most vulnerable frame

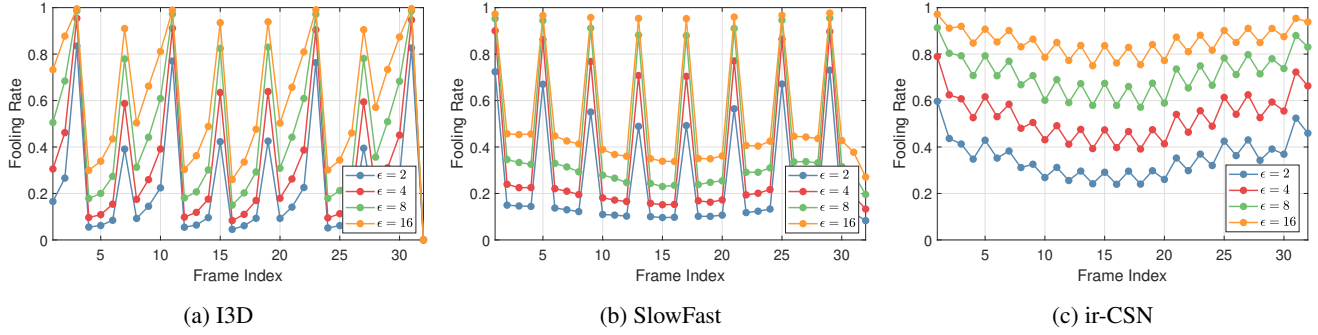


Figure 2: Fooling rates of perturbing a single frame by I-FGSM depending on the frame index.

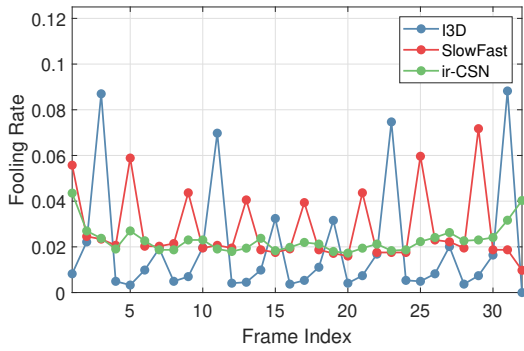


Figure 3: Fooling rates of perturbing a single frame by uniform random noise depending on the frame index.

can be identified. Note that while Figure 3 is obtained using all the video clips, we found that only 100 randomly chosen clips were enough to discover the most vulnerable frames.

We discuss the causes of these interesting observations by analyzing the structural properties of the models, which are illustrated in Figure 4.

I3D. As illustrated in Figure 4a, the I3D model [22] initially extracts features from a given video clip via a convolutional layer having a kernel size of $5 \times 7 \times 7$ and a temporal stride of 2. Then, they are processed by a max-pooling layer having a kernel size of $1 \times 3 \times 3$ and a temporal stride of 2. Through this process, the video clip having 32 frames is contracted to eight frames of features, i.e., an effective temporal stride becomes 4. And, there may exist asymmetric information extraction through the two layers, i.e., when the weights of the convolutional layer have different values across the temporal dimension, the layer’s output relies more on the information at specific frames and less on the information at the other frames among the five frames entering to the kernel, which is more emphasized through the max-pooling layer. In the case of the pre-trained I3D model used in our experiment, the average magnitude of the weights of the first convolutional layer is measured as 0.01, 0.01, 0.02, 0.03, and 0.10, which

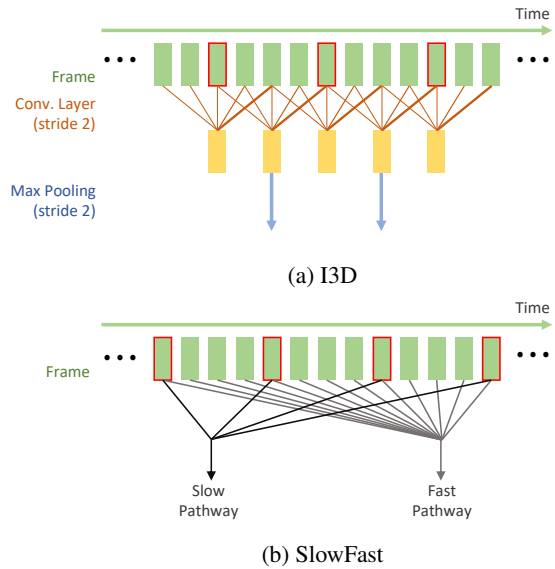


Figure 4: Illustration of structural vulnerability. The frames marked with red boxes correspond to the vulnerable frames showing high fooling rates in Figure 2.

means that the layer mostly relies on the information in the fifth frame among the five input frames (indicated by thicker lines in Figure 4a). By these two mechanisms (an effective stride of 4 and the asymmetric information extraction), the perturbation inserted at $i \in \{3, 7, 11, 15, 19, 23, 27, 31\}$ easily attacks the model (the frames marked with red boxes in Figure 4a), while the perturbation at the other frames does not.

SlowFast. SlowFast [4] is a two-stream model, which includes a fast pathway and a slow pathway. Since the fast pathway uses all of the 32 frames and the slow pathway takes only every fourth frame, only eight frames are simultaneously used by both pathways, which are marked with red boxes in Figure 4b. We find that the highly vulnerable frames ($i \in \{1, 5, 9, 13, 17, 21, 25, 29\}$) exactly match the frames used by both pathways. The other frames

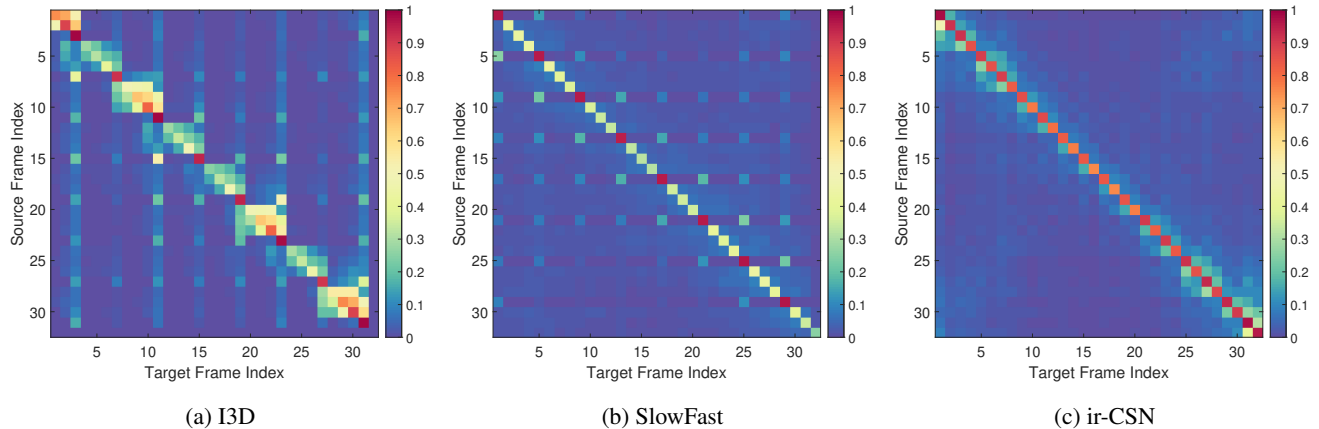


Figure 5: Transferability between different frames in each model. The value at (x, y) means the fooling rate for the x -th frame using the perturbation generated for the y -th frame. Red and blue colors indicate high and low values, respectively.

are processed only through the fast pathway and thus the perturbation on those frames is not so successful. Note that the fast pathway has the structure of I3D but the temporal stride is 1 unlike the original I3D, so the above-observed periodic pattern does not appear here.

ir-CSN. The ir-CSN [20] model used in our study is based on ResNet-152, which is deeper than the other two models using ResNet-50. As aforementioned, the ir-CSN model is relatively vulnerable across all frames; for example, when ϵ is 16, even the lowest fooling rate is 75.4%. The strides in the first convolutional layer and the first pooling layer are 1, thus all input frames are treated evenly in these layers. Therefore, the vulnerability is rather similar across all frames unlike I3D and SlowFast. It is also observed that the fooling rate increases around the two edge frames, which seems to be because zero paddings highlight the perturbation in the edge frames.

Transferability of perturbation. We investigate the transferability of perturbations between frames using I-FGSM, i.e., whether the perturbation generated for a frame can be also used for another frame location directly to attack the model. Figure 5 shows that there exist pairs of source and target frame locations showing particularly high transferability, and the transferability patterns differ depending on the action recognition model.

For I3D and SlowFast, high transferability is achieved between relatively more vulnerable frames (shown in Figure 2). The fooling rate by transferred perturbation is still higher than that by the uniform random noise attack. This suggests that perturbations for vulnerable frames have common features making the model operate wrongly.

Furthermore, in the cases of I3D and ir-CSN, there is a relatively high level of transferability between adjacent frames. However, this is not the case in SlowFast, because

	One frame attack				Wei [23]
	$\epsilon=2$	$\epsilon=4$	$\epsilon=8$	$\epsilon=16$	
I3D [22]	0.83	0.95	0.99	1.00	0.81
SlowFast [4]	0.73	0.90	0.96	0.98	0.72
ir-CSN [20]	0.60	0.79	0.91	0.97	0.68

Table 1: Fooling rates of the one frame attack and the compared attacked method [23] on the three action recognition models. The one frame attack is conducted with various values of ϵ .

the slow pathway does not take all adjacent frames but only every fourth frame.

4. Vulnerability against one frame attack

In Section 3, we could find the most vulnerable frames related to the structural vulnerability of an action recognition model. In this section, we perform the *one frame attack* under a white-box scenario, which applies the I-FGSM algorithm to only the most vulnerable frames, i.e., the 31st, 29th, and first frames for I3D, SlowFast, and ir-CSN, respectively. We evaluate the performance of this attack with respect to two criteria: 1) the fooling rate and 2) the degree of inconspicuousness of perturbation. It is demonstrated that by exploiting the structural vulnerability, the one frame attack can fool the models with high fooling rates and high invisibility in comparison to the existing attack methods.

4.1. Fooling rate

Table 1 summarizes the fooling rates of the one frame attack for each case. The attack achieves high fooling rates for all target models. Especially, when ϵ is equal to or larger

than 8, the fooling rates exceed 90%. The fooling rates are over 60% even when ϵ is as small as 2.

For comparison, we implement the attack method in [23] to attack the deep action recognition models. The fooling rate of this method is lower than that of the one frame attack except the case of ir-CSN with $\epsilon = 2$. Considering that the one frame attack with up to $\epsilon = 16$ is fairly inconspicuous (see Section 4.2), the method in [23] does not effectively capture the vulnerability of the action recognition models. This is because it is designed to exploit the particular mechanism (temporal information propagation) of the LSTM-based models.

4.2. Inconspicuousness

We conduct a subjective test to examine the level of inconspicuousness of the one frame attack. We use 4 (the number of videos) \times 4 ($\epsilon \in \{2, 4, 8, 16\}$) \times 3 (the number of target models) = 48 perturbed videos. Fifteen participants are employed, which meets the required number of participants for subjective tests according to the recommendation ITU-R BT.500-13 [21]. Our test is based on the double-stimulus impairment scale (DSIS) method in [21]. In other words, the participants watch the original video and its perturbed version sequentially for three seconds each with a mid-gray image displayed between the two by following [21]. The order of the video pair is randomly switched. Then, the participants answer whether they notice the difference between the video pair. We also include pairs of original videos to obtain a ‘‘baseline’’ detection performance of the participants. The same procedure is repeated for the pairs of a perturbed frame and its original version where the exposure time of one image is set to two seconds. For comparison, we implement the flickering attack [14] aiming at a high level of inconspicuousness where flickering perturbation is added to all frames to attack a video clip, and the resulting videos are also evaluated.

Figure 6 shows examples of perturbed frames for I3D when we set $\epsilon \in \{2, 4, 8, 16\}$. We also present the detection rates of these frames (when viewed as images) and the videos containing these frames (when viewed as videos). The perturbation in the frames for high values of ϵ is easily found by the participants. As a result, we obtain higher detection rates when they are viewed as images than when they are viewed as videos.

Figure 7 shows the overall results of the subjective test. The frames perturbed by the one frame attack, when viewed as images, are relatively easily detected especially for large values of ϵ . The detection rate of the images increases as ϵ increases, which is natural. However, the videos containing the perturbed frames are hardly detected, showing detection rates even lower than the erroneous detection rate (‘‘baseline’’) for the pairs of the same original videos. In contrast, the flickering attack is easily detectable.

With these results, we can confirm the inconspicuousness of the one frame attack.

5. Vulnerability against universal one frame attack

In Section 4, the one frame attack finds the adversarial perturbation for each video clip and deteriorates the performance of the recognition models, showing the risk of the vulnerability. In this section, we examine the possibility of universal attack [12], which is to find a video-agnostic perturbation that can affect any video clip for a target action recognition model. In addition, we investigate the possibility of extending the one frame attack to a time-invariant universal attack, which assumes a real-time action recognition situation.

5.1. Attack method

The video-agnostic universal perturbation is obtained in a similar way to the method described by (1) and (2). However, instead of the gradient for each video clip, the average gradient for K videos is used in the sign function, i.e.,

$$G^{n+1} = \frac{1}{K} \sum_{k=1}^K \sum_{i \in I} \nabla_{X_k(i)} J(X_k^n, y_k) \quad (4)$$

where I is the set of target frame indices that are used to find a universal perturbation, $X_k(i)$ is the i -th frame of the k -th video clip, and y_k is the ground-truth label of the k -th video clip.

5.2. Implementation details

For finding universal perturbations, we consider two different frame sets I : the set of multiple highly vulnerable frames and the set of only the most vulnerable frame. We set $N = 100$ and $\epsilon \in \{32, 48, 64\}$. Note that a larger number of iterations and larger values of ϵ are required to find universal perturbations compared to the case of video-specific perturbations, as also mentioned in the previous studies [14, 23]. Besides, finding a universal perturbation requires a high computational complexity, since the gradients need to be calculated from all target videos at every iteration. Employing a larger number of videos to find a universal perturbation takes more time, but a higher fooling rate can be expected since the perturbation is found from more diverse videos. To examine this, we vary the number of videos (K) to find the universal perturbation, where $K \in \{100, 200, 500, 1000, 1500\}$.

To test the universal perturbation, we randomly choose additional 1000 videos from the Kinetics-400 dataset, which are different from the videos used for generating the universal perturbation.

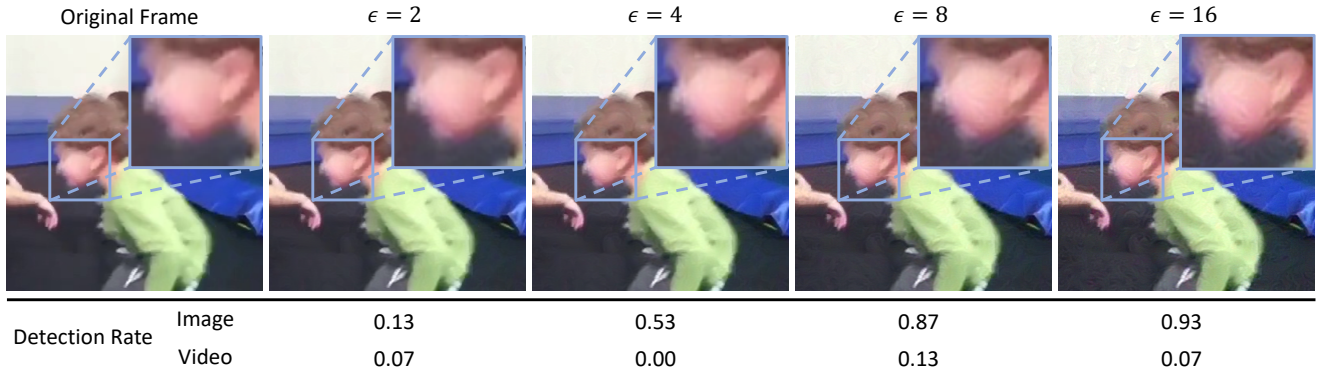


Figure 6: Visibility and the detection rate of the perturbations with $\epsilon = 2, 4, 8, 16$ for I3D.

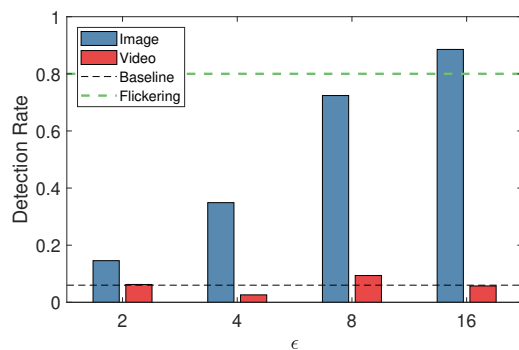


Figure 7: Results of the subjective test in terms of the detection rate. ‘Image’ and ‘Video’ correspond to the cases where the image frames perturbed by the one frame attack and the video clips containing the attacked frames are shown to the participants, respectively. ‘Baseline’ indicates the erroneous detection rate for pairs of original video clips. ‘Flickering’ means the detection rate for the perturbed videos by the flickering attack [14].

5.3. Results of universal attack

Attack on the most vulnerable frame. We first find a universal perturbation from the most vulnerable frames found in Section 3.3 (i.e., I in (4) has only one frame index) for all K video clips. Figure 8a shows fooling rates of this universal one frame attack with respect to the value of K . The results show that increasing the number of videos is usually beneficial to achieve a higher fooling rate. When 1500 videos are used, the attack with $\epsilon = 32$ achieves a fooling rate above 80% for every target action recognition model, indicating the universal one frame attack is feasible.

Transferability of universal perturbation. We also investigate the transferability of the universal perturbation found from the most vulnerable frames. To do this, for every target frame index, we add the same universal

perturbation to that frame and measure the fooling rate. Figure 8b shows the results for I3D and SlowFast. It is shown that the vulnerable frame indices found in Section 3.3 are also highly vulnerable against the transferred universal perturbation, i.e., the universal perturbation is highly transferrable among the vulnerable frames.

Attack on multiple vulnerable frames. Although the one frame attack can find a strong universal perturbation even when only the most vulnerable frame is exploited, more powerful universal perturbation can be found using multiple vulnerable frames. To evaluate this, we find a universal perturbation using all the vulnerable frames found in Section 3 (i.e., I in (4) has multiple frame indices). Figure 8c shows the results when the found perturbation is injected to the most vulnerable frames. When compared to the result of the universal attack found using only the most vulnerable frame (Figure 8a), this result shows that the universal perturbation found from multiple vulnerable frames is more powerful. For example, when 500 or 1000 videos are employed to find the universal perturbation, the fooling rates for both I3D and SlowFast exceed 90% in any values of ϵ in Figure 8c. In addition, sufficiently strong universal perturbations can be found even when a small number of videos are employed. For instance, the fooling rates for all cases when $\epsilon \in \{48, 64\}$ exceed 90% when only 200 videos are employed.

6. Vulnerability against time-invariant universal attack

We additionally consider a real-time recognition scenario, where video data is continuously generated and a video clip is repeatedly chosen for recognition from the generated video sequence using a sliding window [10]. Two challenges arise in this scenario. First, there may not be enough time to generate perturbations specific to the inputted video. Therefore, a universal perturbation computed

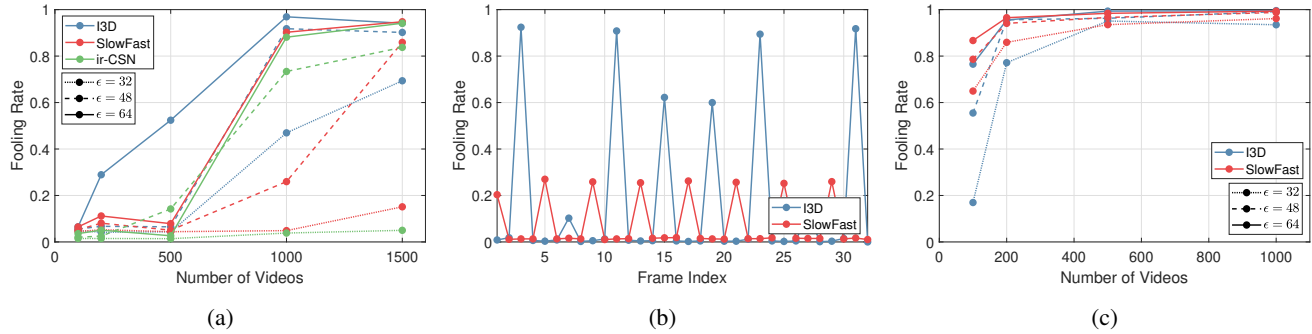


Figure 8: Results of the universal one frame attack. (a) Fooling rates with respect to the number of videos when the most vulnerable frame of each video clip is attacked. (b) Transferability of the universal perturbation found using the most vulnerable frame of each of 1500 videos to the other frames, where $\epsilon = 48$. (c) Fooling rates with respect to the number of videos when multiple vulnerable frames in each video clip are used to obtain the perturbation.

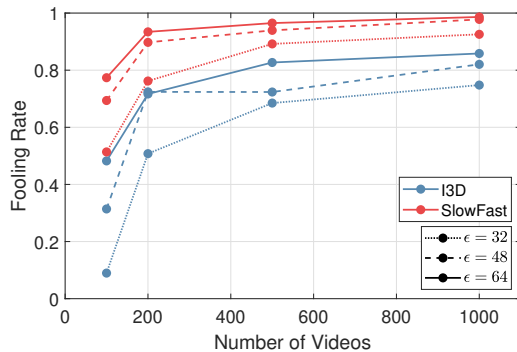


Figure 9: Fooling rates of the time-invariant universal attack with respect to the number of videos.

a priori needs to be used to ensure real-time operation. Second, there may exist an unknown temporal offset between the video clip chosen for recognition and the video clip that the attacker observes. An attack method that can deal with this scenario is referred to as a time-invariant attack [14]. In this section, we examine the feasibility of the universal one frame attack as a time-invariant attack.

As shown in Section 3.3, vulnerable frames appear periodically for I3D and SlowFast. By taking advantage of this, a time-invariant universal attack can be designed as follows. Let P denote the period of vulnerable frames and L the length of a video clip, which are 4 and 32 for both I3D and SlowFast, respectively. Then, we add a pre-computed universal one frame perturbation, which is generated by attacking on multiple vulnerable frames in Section 5.3, to P frames in every L frames in such a way that the frame index of each of the P frames corresponds to a distinct remainder when divided by P . For instance, the 1st, 10th, 19th, and 24th frames, whose remainders are 1, 2, 3, and 0 when divided by 4, respectively, are perturbed, which is repeated for

the next 32 frames. This ensures that one of the P perturbed frames always corresponds to a vulnerable frame index no matter which frame in the video sequence is chosen as a starting frame of the video clip for recognition. As a special case of this process, we can simply perturb the first P frames in every L frames, e.g., 1st to 4th, 33rd to 36th, etc.

Figure 9 shows the results of this attack with respect to the number of videos used to find the universal perturbation. When 1000 videos are used, the fooling rates exceed 70% in all cases. This confirms that the action recognition models are highly vulnerable even in the challenging real-time scenario against the time-invariant universal attack.

It is also worth noting that the attack presented here is more efficient than the existing methods in [10, 14]. They need to generate perturbations considering all possible cases of temporal offsets, which is not required in the case of the presented attack method.

7. Conclusion

We presented in-depth analysis of the structural vulnerability of deep action recognition models against adversarial attack. Based on the results of perturbing a single frame in a given video clip, we analyzed that the vulnerability is caused by the structural properties such as strides in convolutional layers and max pooling layers and uneven use of inputted frames. As a result, we demonstrated the possibility of the strong one frame attack that is found to be highly inconspicuous via a subjective experiment. Finally, we investigated the possibility of finding universal perturbations showing high fooling rates in various attacking scenarios.

Acknowledgement

This work was supported by the Artificial Intelligence Graduate School Program (Yonsei University, 2020-0-01361).

References

- [1] 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.
- [2] MMAction2 Contributors. Openmmlab’s next generation video understanding toolbox and benchmark. <https://github.com/open-mmlab/mmdetection>, 2020.
- [3] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.
- [4] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019.
- [5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Proceedings of the International Conference on Learning Representations*, 2015.
- [6] Samitha Herath, Mehrtash Harandi, and Fatih Porikli. Going deeper into action recognition: A survey. *Image and Vision Computing*, 60:4–21, 2017.
- [7] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, 2012.
- [8] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv:1705.06950*, 2017.
- [9] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Proceedings of the International Conference on Learning Representations Workshop*, 2017.
- [10] 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 *Proceedings of Network and Distributed System Security Symposium*, 2019.
- [11] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *Proceedings of the International Conference on Learning Representations*, 2016.
- [12] 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.
- [13] Alexandre Ninassi, Olivier Le Meur, Patrick Le Callet, and Dominique Barba. Considering temporal variations of spatial visual distortions in video quality assessment. *IEEE Journal of Selected Topics in Signal Processing*, 3(2):253–265, 2009.
- [14] Roi Pony, Itay Naeh, and Shie Mannor. Over-the-air adversarial flickering attacks against video recognition networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 515–524, 2021.
- [15] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 568–576, 2014.
- [16] Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. Is robustness the cost of accuracy? – A comprehensive study on the robustness of 18 deep image classification models. In *Proceedings of the European Conference on Computer Vision*, pages 631–648, 2018.
- [17] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.
- [18] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations*, 2013.
- [19] 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.
- [20] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5552–5561, 2019.
- [21] International Telecommunication Union. Recommendation ITU-R BT.500-13: Methodology for the subjective assessment of the quality of television pictures. 2012.
- [22] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018.
- [23] Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8973–8980, 2019.
- [24] Yin Zhao and Lu Yu. Evaluating video quality with temporal noise. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 708–712, 2010.