PatchZero: Defending against Adversarial Patch Attacks by Detecting and Zeroing the Patch

Ke Xu*   Yao Xiao*   Zhaoheng Zheng   Kaijie Cai

Ram Nevatia

{kxu47918, yxiao915, zhaoheng.zheng, kaijieca, nevatia}@usc.edu

Abstract

Adversarial patch attacks mislead neural networks by injecting adversarial pixels within a local region. Patch attacks can be highly effective in a variety of tasks and physically realizable via attachment (e.g. a sticker) to the real-world objects. Despite the diversity in attack patterns, adversarial patches tend to be highly textured and different in appearance from natural images. We exploit this property and present PatchZero, a general defense pipeline against white-box adversarial patches without retraining the downstream classifier or detector. Specifically, our defense detects adversaries at the pixel-level and "zeros out" the patch region by repainting with mean pixel values. We further design a two-stage adversarial training scheme to defend against the stronger adaptive attacks. PatchZero achieves SOTA defense performance on the image classification (ImageNet, RESISC45), object detection (PASCAL VOC), and video classification (UCF101) tasks with little degradation in benign performance. In addition, PatchZero transfers to different patch shapes and attack types.

1. Introduction

Early adversarial image attacks [13, 5, 34] perturb pixels over the entire image; while these attacks are highly effective, they are hard to realize in the physical environment. This has led to study of adversarial patch attacks that inject adversarial pixels within specified local regions, from the early Adversarial Patch [4], LaVAN [24], Masked Carlini-Wagner [5], and Masked PGD [35] attacks to the more recent DPatch [30], Robust DPatch [26], and Masked AutoPGD [9] attacks. Patch attacks are physically realizable as they can be printed and placed into the scene. Real-world safety-critical computer vision systems, such as autonomous driving and security surveillance, are vulnerable to adversarial patches in the targeted scenes.

To tackle adversarial patch attacks in different domains,

---

*Equal contributions

(a) PARACHUTE   (b) CASSETTE PLAYER

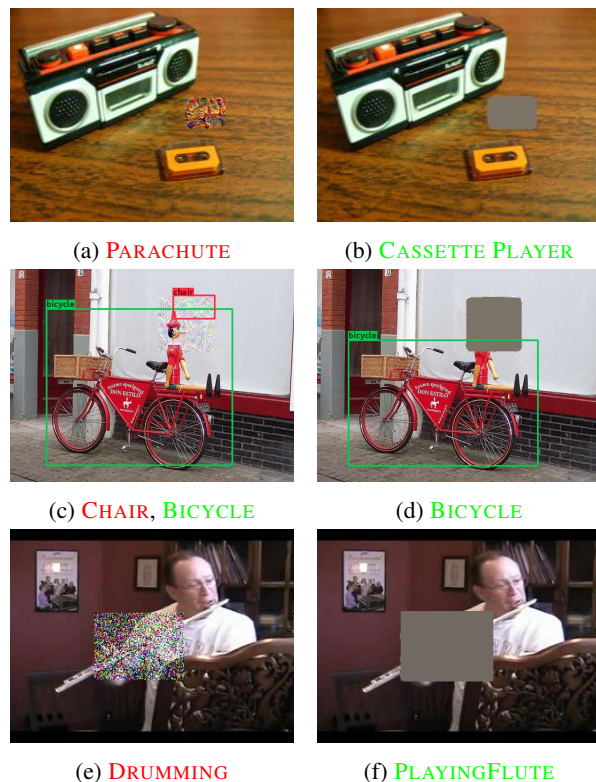(c) CHAIR, BICYCLE   (d) BICYCLE

(e) DRUMMING   (f) PLAYINGFLUTE

Figure 1: **PatchZero defense for adversarial patch attacks**. PatchZero takes an adversarial image (left) as input and outputs a processed image (right) with adversarial pixels effectively removed. Our approach can be applied to the image classification (top), object detection (middle), and video classification (bottom) tasks without any retraining or modification of the downstream classifiers or detectors. Green and red denote correct and incorrect predictions.

many patch defense methods have been proposed. Most of the defenses focus on image classification [18, 36, 15, 27, 8, 49, 45, 48, 46], while object detection defenses [51, 7, 47] and video classification defenses [1, 31, 33] are relatively underexplored. Moreover, most of the defenses cannot be

easily adapted to a different task and very few of them consider adaptive attacks. Some [45, 46] also require prior knowledge, such as adversarial patch size, to be effective. In this study, we focus on adversarial patch attacks under the white-box setting, as they are stronger [2, 43] than the black-box counterparts. We aim to design a general defense pipeline that can be easily applied to different classification and detection tasks under adaptive attacks, and does not require any prior attack knowledge.

Our defense is based on the observation that although adversarial patches are localized, they can mislead predictions for objects far away in the image by exercising abnormally large influence on the spatial context due to their highly textured patterns. As shown in the first column of Figure 1, adversarial patches often have quite distinct texture and color distributions from those found in natural images. This observation leads to our idea to identify adversarial pixels with pixel-level patch detector and replace those pixels with mean pixel values (zero values after image normalization) to reduce or even eliminate their influences. Empirically, this process effectively "zeros out" the adversary and restores most of the accuracy for the downstream tasks. Therefore, we name our method "PatchZero".

In an adaptive, white-box attack setting, the patch detector itself may be vulnerable to the attack. Since our patch detector outputs a binary mask which is non-differentiable during the backpropagation, we approximate the gradient of the binary mask using the Backward Pass Differential Approximation (BPDA) [3] technique. We propose a two-stage adversarial training scheme to efficiently train PatchZero under BPDA attacks. The patch detector is first trained with DO attack examples and then reinforced with BPDA joint attack examples in successive stages. It is natural for the adversary and defender to train their models iteratively; however, it may be expected that this will always lead to a win for the adversary as the attacker gets to make the last call. One of our key contributions is in demonstrating that after some iterations of alternate training, the defense model becomes robust and able to detect the adaptive patches effectively.

We evaluate PatchZero under the Masked PGD, Masked AutoPGD, and Masked CW attacks since they give a good coverage of the white-box patch attacks and can be easily applied to different tasks. PatchZero achieves state-of-the-art performances on all three tasks compared with the previous work, with little degradation in benign performance. Under the stronger BPDA adaptive attacks, the advantage margin of our defense method is even larger. Note that a recent paper [29] uses a similar defense approach for patch attacks on the object detection task. We are unable to compare with it directly as it uses different datasets and attack conditions.

To summarize, our contributions are threefold:

1. We present PatchZero, a general defense pipeline against white-box patch attacks that can be easily adapted to the tasks of image classification, object detection, and video classification without retraining of the downstream classifier or detector.

2. We introduce a two-stage training scheme that reinforces PatchZero's robustness under the stronger adaptive attacks and accelerates training in the early stages.

3. We evaluate our defense on multiple datasets and demonstrate generalization to different patch shapes and attack types.

2. Related Work

Patch Attacks: Perturbation attacks manipulate the whole image to mislead neural networks. Patch Attacks, on the other hand, only modify a restricted region of the image. Brown et al. [4] first introduce the Adversarial Patch attack that generates a universal and physically realizable patch to mislead the image classification models. LaVAN [24] is proposed at the same time but focuses on the digital patch. After the introduction of the full-image Carlini-Wagner (CW) [5], PGD [35] and AutoPGD [9] attacks, Masked CW, Masked PGD and Masked AutoPGD are three extensions to patch attacks by restricting the attack region.

We would like to mention some task-specific attacks. In the object detection domain, Liu et al. [30] design DPatch against popular object detectors. Lee et al. [26] investigate failure cases of DPatch and later introduce the Robust DPatch. Furthermore, Saha et al. [40] introduce a blindness attack against the classifier inside an object detector, while Rao et al. [38] propose localization-optimized attacks. In the video classification domain, the only patch attack we can find besides Masked PGD and Masked AutoPGD attacks is the MultAV attack by Lo et al. [32]. MultAV is very similar to Masked PGD, but uses multiplication instead of summation when applying the perturbation.

We select Masked PGD and Masked AutoPGD for our experiments, since they can be easily applied across different tasks. Empirically, we also find them to be stronger than the task-specific attacks.

Patch Defenses for Image Classification: Digital Watermark(DW) [19] and Local Gradient Smoothing (LGS) [36] are among the early patch defenses. Both are later proved to be ineffective by Chiang et al. [8], who proposes the first certified defense call Interval Bound Propagation (IBP). IBP limits the values of activation maps to guarantee a robustness lower bound. More recently, Xiang et al. [45] put forward PatchGuard, a network with small receptive field and outlier masking. It requires non-trivial modification of the backbone classifier. The same authors later propose another defense called PatchCleanser [46] that can

be applied to any classifier. PatchCleanser uses an ensemble and exhaustive masking technique to identify the patch region. Both PatchGuard and PatchCleanser require the prior knowledge of the attack patch size to compute the optimal mask size and their certified robustness do not hold well for large patches. In comparison, our approach can defend against patch attacks of any patch sizes and shapes without any prior knowledge.

Patch Defenses for Object Detection: Similar to the image classification defense, object detection patch defense also receives a lot of attention recently. Liang et al. [28] uses Grad-Cam to detect and filter out the unusual area of the image. However, Grad-Cam can only provide a coarse map and is subject to miss detection and false positives. Zhou et al. [52] combine Grad-Cam gradient map and discrete entropy to locate the adversarial pixels, but the detection results are still coarse and limited. DetectorGuard [47] uses small receptive field CNN to output a robust objectness map that indicates the probability of objects being present at different locations. If the map results are different from the basic predictions, they will raise an alert for the adversary. However, this work only identifies but does not defend against patch attacks. SAC [29] uses an approach of detecting and removing adversarial patches for the object detection task that is similar to ours. SAC uses identity mapping for binary mask gradient estimation, which we believe is weaker than our Sigmoid BPDA gradient estimation. It was proposed almost the same time as our approach and the code was not released, so we are not able to provide a direct comparison.

Patch Defenses for Video Classification: Adversarial patch defense is a relatively under explored research direction in video classification. Anand et al. [1] propose Inpainting with Laplacian Prior (ILP) to detect and inpaint adversarial pixels in the Laplacian space. However, the method only works for optical-flow based video classifiers. Lo et al. propose to replace each Batch Normalization (BN) [22] layer of a regular video classifier with three BNs. The network needs to be retrained adversarially to learn a "switch mechanism" to connect to the correct BN module. The same authors later propose OUDefend [33] module as an embedding feature denoiser to be inserted between the layers of a video classifier. Both methods from Lo et al. require modification and retraining of the downstream classifiers, while our approach can be plugged into any classifier.

3. Defense Against Adversarial Patch Attacks

In this section, we first introduce some related background in Section 3.1. Then we explain the PatchZero defense in Section 3.2. Lastly, we elaborate on the two-stage training scheme for robustness against the stronger adaptive patch attacks in Section 3.3.

3.1. Background

Projected Gradient Descent (PGD) and AutoPGD: Introduced by Madry et al. [34], PGD attack is one of the strongest perturbation attacks proven to be effective against image classification models. Given an input image X, its ground-truth label Y^*, model weights θ and the loss function ℓ, PGD attack is generated by maximizing the loss function in an iterative manner:

$$X_{adv}^{(t+1)} = C_\epsilon\{X_{adv}^{(t)} + \alpha Sign(\nabla_X \ell(X_{adv}^{(t)}, Y^*, \theta)\}. \quad (1)$$

Note that the clipping function C is utilized to prevent the per-pixel modification from going beyond the threshold ϵ. In addition, random initialization and restarts are adopted to further strengthen the attack. AutoPGD [9] is later proposed as a PGD with auto step size tuning and a refined objective function. It is shown to be more effective than PGD under the same attack budget.

Masked PGD and Masked AutoPGD: Although the original PGD attack is designed for the full-image perturbation attacks, it can be easily converted into a patch attack. As shown in Eq. 2, only pixels inside the patch region $[x, y, h, w]$ will be modified by the PGD:

$$X_{adv}^{(t+1)}[patch] = \\ C_\epsilon\{X_{adv}^{(t)} + \alpha Sign(\nabla_X \ell(X_{adv}^{(t)}, Y^*, \theta)\}[patch]. \quad (2)$$

Here *patch* refers to the region defined as $[x : x + h, y : y + w]$ with the given $[x, y, h, w]$. Masked PGD can attack object detectors and video classifiers by deriving the gradients from corresponding loss functions. AutoPGD can be converted to its patch attack counterpart Masked AutoPGD in a similar manner.

Adversarial Training: Adversarial training [13, 34] has proven to be effective against various adversarial attacks. The key idea is to generate adversarial examples and inject them into the mini-batches during training. Generally, the effectiveness of adversarial training depends on the strength of adversarial examples. In practice, several researchers [42, 23, 41] have studied PGD attack and achieved significant robustness through adversarial training. To tackle adversarial patch attacks, we propose a two-stage training scheme that adversarially trains our models in two stages with samples produced by Masked PGD and Masked AutoPGD. The details are provided in Section 3.3.

3.2. PatchZero Network

The full pipeline of PatchZero is shown in Figure 2. Our method consists of two steps. In the first step, the input image $X \in \mathbb{R}^{H \times W \times C}$ is processed by the patch detector $d : \mathbb{R}^{H \times W \times C} \to [0, 1]^{H \times W}$, which yields a probability
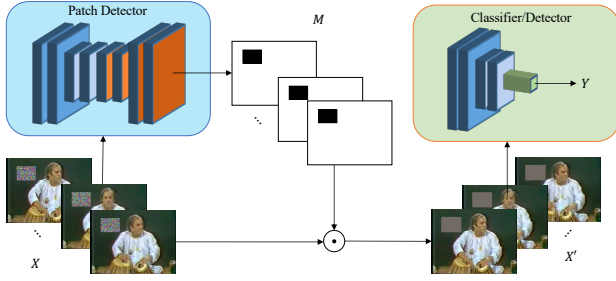
Figure 2: **Defense Pipeline of PatchZero.** The patch detector takes one or multiple attack images X and predicts pixel-wise adversarial binary mask M (black for adversarial pixels and white for benign pixels). We "zero out" the patch by multiplying X with M and fill the patch region with the mean pixel values. The preprocessed image X' is passed to the downstream model for final predictions.

map describing the possibilities for each pixel not being manipulated. We then binarize through a threshold ϵ_p the probability map to a binary mask $M \in \{0, 1\}^{H \times W}$, where patch pixels are denoted zeros. For the second step, we remove the identified patch region via element-wise multiplication between X and M. The masked region is then filled with mean pixel value $\overline{X}$ computed from the dataset to generate:

$$X' = X \odot M + \overline{X} \odot \neg M. \qquad (3)$$

After the zero-out step, the downstream model f takes the sanitized image X' and makes the final prediction Y.

We generate adversarial patches of random locations and sizes and the corresponding ground truth binary mask for each image. We construct the training set for d by equally mixing attack images and benign images. During the patch detector training, we follow the loss function of PSPNet [50], which consists of a main cross entropy loss and two auxiliary loss terms. During inference, the patch detector can detect and "zero out" the adversarial pixels most of the time, but occasionally misses some pixels at the border. To this end, we use morphological dilation to slightly enlarge the predicted mask by a few pixels.

3.3. Adaptive Attack and Two-stage Training

When generating adversarial patches, there are two attack strategies. In the Downstream-only (DO) attack, only gradients from the downstream classifier f are considered. However, the patch detector d itself is vulnerable to adversarial attacks, especially under the white-box setting, where attackers have full knowledge of the pipeline. In the stronger adaptive attack, both the gradients from the downstream classifier f and from the patch detector d are considered. As defined in Eq. 3, the zero-out step includes the non-differentiable binarization operation. The pixel-level gradi-

ent $\nabla_X \ell(X_{adv}^{(t)}, Y^*, \theta)$ cannot be computed directly through back-propagation.

BPDA Adaptive Attack: Proposed by Athalye et al. [3], BPDA is an approximation strategy to bypass non-differentiable layers inside a network and achieve effective adaptive attacks. Given a non-differentiable operation h, BPDA finds a differentiable approximation h' that satisfies $h(x) \approx h'(x)$. The original operation h is used in the forward pass but replaced by the approximation h' in the backward pass. To apply BPDA to PatchZero, we approximate the binarization operation by the Sigmoid function, since the binarization is essentially a Step function. With this approximation, we can utilize gradients from both the patch detector and the victim classifier or detector to generate attacks.

Two-stage Adversarial Training: The BPDA adaptive attack introduces some difficulty to the training process of the patch detector. In the early stages, the patch detector is immature and creates random gradients. Since the adaptive attack passes the gradients from the downstream classifier through the patch detector, the resulting gradients will be misleading. To resolve this issue, we propose a two-stage training scheme as described below:

- **Training Stage 1:** We first generate adversarial patches using the DO attack, which only consider the gradients from the downstream classifier or detector (Figure 2 green box). We train the patch detector d with a mixture of benign and adversarial images.

- **Training Stage 2:** When the patch detector starts to converge on the DO attack images, we switch to the 2nd stage of training. We generate adversarial patches using the BPDA adaptive attack, which considers gradients from both parts (blue and green box of Figure 2) of the pipeline. We generate online adversarial attacks at every training step with updated model weights. This practice creates an attacker-defender race and further fortifies the effectiveness of the patch detector.

The two-stage training mechanism greatly accelerates the training process and improves the robustness of PatchZero under the stronger BPDA adaptive attacks.

4. Experiments

We adopt the PSPNet [50] with the ResNet-50 [20] backbone as the patch detector of PatchZero. We initialize the PSPNet with weights pre-trained on the ImageNet[10] and follow the loss function for image segmentation. We train our PSPNet patch detector through the two-stage adversarial training introduced in Section 3.3. Regarding the binarization threshold, we set ϵ_p as 0.5. We developed the PatchZero and the two-stage training scheme in PyTorch [37] and use the Adversarial Robustness Toolbox

Defense	Benign	MPGD	MAPGD	MCW
Undefended	81.62%	14.35%	9.40%	49.57%
GT Mask	81.60%	81.42%	81.34%	81.37%
PG [45]	60.40%	49.41%	48.91%	56.95%
PC [46]	80.54%	64.30%	63.57%	73.12%
PZ (DO)	**81.47%**	**75.60%**	**76.80%**	**74.24%**
PZ (BPDA)	81.48%	55.46%	70.02%	-

Table 1: **Benign and robust accuracy on the ImageNet classification dataset.** PG, PC, and PZ stand for Patch-Guard, PatchCleanser, and PatchZero respectively.

(ART)[1] for generating attacks. For the two-stage training, we use a learning rate of 0.0001, the Adam [25] optimizer, and a batch size of 64 for image classification, 16 for object detection, and 36 for video classification.

4.1. Image Classification

Implementation Details: We conduct our image classification experiments on two datasets. We use the validation split of the ImageNet [10] dataset with 50,000 images and 1000 classes. We also evaluate on the RESISC-45 [6] remote sensing dataset that contains 31,500 images and 45 scene classes. Compared with the ImageNet, RESISC-45 has a larger image size (256x256) and provides a remote sensing perspective. On the ImageNet, we use ResNet50-v2 as the backbone image classifier for all the defense methods. On the RESISC-45, we use DenseNet121 [21] as the image classifier. We use the top1 accuracy for evaluation.

Attacks: For the Masked PGD (MPGD) attack, we use a perturbation strength of 1.0, a step size of 0.01 and 100 iterations. For the Masked AutoPGD (MAPGD) attack, we use a perturbation strength of 0.3, a step size of 0.1 and 100 iterations. For the Masked Carlini-Wagner (MCW) attack, we use a perturbation confidence of 0.5, a learning rate of 0.1 and 100 iterations. Following the same settings as the previous works, we use 2% rectangular patches for ImageNet and 9% square patches for RESISC-45. The patch sizes are w.r.t. the image area and patch locations are random.

Baseline Defenses:

- **PatchGuard**: PatchGuard [45] is a certified defense with small receptive field and outlier masking. We empirically evaluate the robustness under the same attack settings. Note that the method requires prior knowledge of the attack patch size to estimate the defense mask window size.

- **PatchCleanser**: PatchCleanser [46] is another certified defense against adversarial patches via two rounds

[1] https://github.com/Trusted-AI/adversarial-robustness-toolbox

Defense	Benign	MPGD	MAPGD
Undefended	92.9%	3.0%	1.7%
GT Mask	92.9%	87.8%	87.2%
JPEG Comp [11]	91.0%	4.1%	1.7%
Adv Training [14]	83.9%	71.8%	67.2%
PZ (DO)	**92.9%**	**87.5%**	**85.0%**
PZ (BPDA)	92.9%	81.2%	76.4%

Table 2: **Benign and robust accuracy on the RESISC-45 classification dataset.**

of exhaustive masking and ensemble. We empirically evaluate the robustness under the same attack settings. This defense method also requires prior knowledge of attack patch size.

- **JPEG Compression**: Guo et al. [16] propose to defend against adversarial attacks through image transformations, including JPEG compression. Here we use JPEG compression as a preprocessor defense.

- **Adversarial Training**: For each downstream model f, we follow a typical adversarial training scheme [34] and train the downstream classifier with a mixture of clean and adversarial images.

Defense Results: We first present the undefended baseline and GT Mask baseline which assume perfect adversarial patch detection. As shown in Table 1, the GT baseline recovers most of the robustness accuracy compared with no attack, showing the potential of our approach. The two certified defense baselines, PatchGuard and PatchCleanser, both require prior knowledge of attack patch size and the robustness declines as the patch size increases. Our method's performance is not strongly dependent on the patch size (except for occlusion effects) but we tested with 2% patch size for fair comparison.

PatchZero outperforms PatchGuard by 26% and Patch-Cleanser by 13% on both the MPGD and MAPGD attacks. PatchZero has similar performance as PatchCleanser and both outperform PatchGuard by 17% on the MCW attack. Compared with the GT Mask results, PatchZero has almost no drop in accuracy in all attacks except for MCW under the DO attack, but larger gaps under the stronger BPDA adaptive attack. Note that neither PatchGuard nor PatchCleanser can be easily adapted for adaptive attack.

We also evaluated PatchZero on the RESISC-45 dataset to test robustness under higher image resolution and much larger patch sizes (9% of image size), as shown in Table 2. We compare with JPEG compression and adversarial training baselines. JPEG compression performs poorly; adversarial training shows much better defense but PatchZero performs better by a large margin, even under the stronger

Defense	Benign			MPGD		
	AP	AP50	AP75	AP	AP50	AP75
Undefended	49.20%	76.4%	52.6%	6.5%	10.9%	6.7%
GT Mask	49.2%	76.4%	52.6%	43.0 %	68.8%	44.4%
JPEG Comp [11]	47.7%	75.0%	51.4%	30.0%	48.1%	32.3%
Adv Training [14]	47.7%	75.1%	51.7%	16.8%	31.9%	15.2%
PZ (DO)	**48.4%**	**75.3%**	**51.8%**	**41.5%**	**66.1%**	**43.8%**
PZ (BPDA)	48.4%	75.3%	51.8%	35.1%	60.0 %	35.5 %

Table 3: **Benign and robust AP on the PASCAL VOC object detection dataset.**

Defense	Benign	MPGD		MAPGD	
		5%	10%	5%	10%
Undefended	94.55%	8.42%	3.96%	18.81%	0.00%
GT Mask	94.55%	91.58%	93.07%	91.58%	93.07%
Video Comp [17]	**94.55%**	21.29%	6.44%	12.87%	0.99%
PZ (BPDA)	**94.55%**	**81.68%**	**82.67%**	**73.27%**	**76.24%**

Table 4: **Benign and robust accuracy on the UCF101 video classification dataset.**

BPDA adaptive attack. Also, adversarial training reduces benign accuracy substantially (by 9%), while PatchZero maintains the benign accuracy of the undefended model.

For both datasets, we can see that the BPDA accuracy drops from GT accuracy, more seriously in ImageNet than in RESISC-45, likely due to much larger variety in the former. Nonetheless, substantial improvements are achieved over undefended model and available alternatives. Further improvements in the patch detection performance will be a consideration in our future research.

4.2. Object Detection

Implementation Details: For the object detection task, we evaluate on the PASCAL VOC [12] dataset, which has 20 object categories. Following the same setting as previous works [44], our models are trained on VOC 2007 plus VOC 2012 and tested on VOC 2007. We use the Faster-RCNN [39] with ResNet-50 [20] as the downstream detector. For evaluation, we use the standard Average Precision (AP), AP50, and AP75 metrics.
Attacks: We defend against Masked PGD attack with a perturbation strength of 0.3, step size of 0.1, and 100 iterations. Patch sizes are 120×120 and patch locations are random.
Baseline Defense: We adopt Adversarial Training, JPEG Compression as the baseline defense methods, since we are unable to find any other patch defense baselines and the two image classification baselines do not obviously transfer to the detection task.
Defense Results: Table 3 shows evaluation results against

the Masked PGD attack on PASCAL VOC. Similarly, the GT Mask baseline assumes perfect patch detection and recovers most of the accuracy compared with no attack. PatchZero achieves an AP of 41.5%, around 8% lower than the benign performance, while JPEG compression and adversarial training only get 30.0% and 16.8% AP, respectively. The DO attack results of PatchZero are very close to the GT Mask results. PatchZero also outperforms the other baselines on the benign images. The BPDA results are lower compared with our DO results, but still 5% higher than JPEG Compression and 18% higher than Adversarial Training, even though they use the much weaker DO attack.

4.3. Video Classification

Implementation Details: We conduct our video classification experiments on the UCF101, an action recognition dataset that has 13,320 short trimmed videos from 101 action categories. Since adversarial defense is computationally expensive on the video domain, we randomly select 202 video from the test dataset. We adopt the MARS [21] model as the downstream classifier. We use the top1 and top5 classification accuracy as the evaluation metrics.
Attacks: For video classification, we consider the Masked PGD and Masked AutoPGD attacks, with perturbation strength of 1.0, step size of 0.2, and 20 iterations. All attacks use BPDA and have patch sizes of 5% and 10%. The patch locations are fixed for all frames of the same video but random for each video.
Baseline Defenses: Due to the lack of reliable adversarial

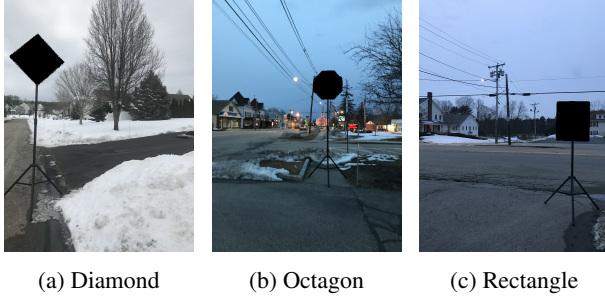(a) Diamond   (b) Octagon   (c) Rectangle

Figure 3: **Transfer across patch shapes on the DAPRI-COT object detection dataset.**

patch defense methods in video classification, we pick the H.264 video compression [17] as the baseline defense.

Defense Results: We compare the defense performance of different defense methods in Table 4. All methods use the MARS model as the downstream classifier. The GT baseline assumes perfect patch detection and recovers most of the robustness accuracy compared with no attack. For the benign videos, neither PatchZero nor the video compression degrade accuracy compared with the undefended MARS classifier. For the attack scenarios, Masked AutoPGD attack is stronger than the Masked PGD attack and attacks with larger patch sizes (10%) are stronger. PatchZero significantly outperforms the video compression baseline under all attack combinations. The margin is even larger for the stronger Masked AutoPGD attacks and larger patch sizes. For example, for the 10% Masked AutoPGD attacks, our method outperforms the Video Compression baseline by a margin of 75.25% on the top1 accuracy. Compared with the GT Mask, PatchZero still has some performance gap, but already outperforms the other baseline by a large margin.

Lo et al. proposed "3-BN" [22] and "OUDefend" [33] modules as defense for multiple video attacks. The authors do not provide implementation for either method, so we cannot thoroughly compare with them. In their only patch attack experiment, they use a much weaker, Downstream-only Masked PGD attack with patch size of 1.2%, perturbation strength of 1.0, and 5 iterations. The "3-BN" achieves a 63.8% accuracy and OUDefend achieves a 42.00% accuracy. In comparison, we use the adaptive version of Masked PGD attack and with stronger attack parameters: patch sizes of 5% and 10%, perturbation strength of 1.0, and 20 iterations. PatchZero achieves 81.68% top1 accuracy, almost 20% higher. Also, both methods require modification and adversarial training of the downstream video classifier. Neither of them can be easily applied across tasks.

4.4. Discussion

Effectiveness of the Patch Detector: To figure out how the patch detector performs in identifying the corrupted pix-

	Recall	Prec	Acc	F1
DO	99.8%	99.1%	99.9%	99.5%
BPDA	98.8%	99.1%	99.8%	99.0%

Table 5: **Adversarial pixel segmentation performance of the patch detector on RESISC-45.**

	MPGD	MAPGD	MCW
MPGD	**81.07%**	80.18%	71.67%
MAPGD	**81.07%**	**81.13%**	66.97%
MCW	80.71%	80.72%	**77.41%**

Table 6: **Generalization to different attacks on ImageNet.** Each row and column represents a model trained with and tested on a specific type of attack.

Model	Time	GPU	Param
ResNet-50	8 mins	1.96 GB	25.5M
PC [46]	758 mins	7.32 GB	25.5M
PZ	12 mins	3.33 GB	72.2M

Table 7: **Memory cost and speed.** Inference time is on the entire ImageNet validation dataset with one Nvidia 2080Ti.

els, we conduct quantitative evaluations on the RESISC-45 dataset. Attacks are generated by Masked AutoPGD under the DO and BPDA attack modes. We report the precision, recall, accuracy, and F1 of the adversarial pixel segmentation task on attack images. According to Table 5, our patch detector can effectively identify manipulated pixels under both attack modes, although there is a 1% drop in Recall from DO attack to the stronger BPDA attack. Empirically, we observe that these 1% uncovered pixels, especially at the patch border, can have some influence on the overall accuracy. We also evaluated our patch detector on the benign images. The false positive detection rate is 5.05e-06.

Transfer across Patch Shapes: The patch detector in PatchZero operates at the pixel-level, so it can generalize well to different patch shapes. We evaluate a version of PatchZero, trained using only square patches, on the Dynamic APRICOT dataset[2], which contains a mixture of diamond, octagon, and rectangle shape patches. We use Faster-RCNN as the downstream detector and the Masked PGD attack with 100 iterations. As shown in Figure 3, PatchZero can detect and remove adversarial pixels accurately across the three different shapes. Quantitatively, the mAP of the undefended baseline drops from 27.33% to 0%; in contrast, PatchZero has the same benign accuracy of 27.33%

---

[2]https://armory.readthedocs.io/en/latest/scenarios/#dapricot-object-detection-updated-july-2021
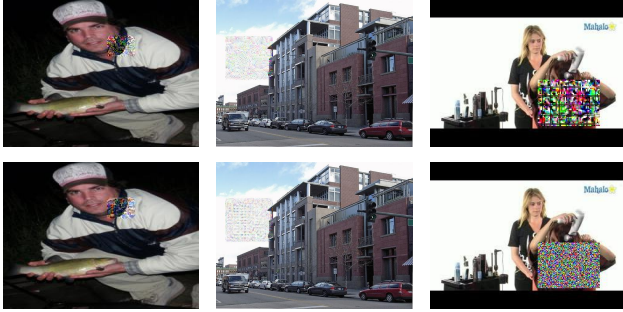
Figure 4: **Attack visualization.** We compare the MAPGD attack patterns under the DO (top) and BPDA adaptive attacks (bottom) in the image classification (left), object detection (middle), and video classification tasks (right).

and maintains 20.67% mAP after attack.

Transfer across Attack Types: We performed experiments of training PatchZero on one of the three attacks (MPGD, MAPGD, and MCW) and evaluating the defense performance on all three; results are shown in Table 6. The attacks are DO only on the Imagenet classification task (the numbers are a bit different than those reported in Table 1 of the submitted paper as those results were the BPDA trained models). The results show that, in general, PatchZero trained on one type of attack defends against another attack quite effectively, with only small drops compared to training on the seen attacks. This is particularly the case for the model trained with the MCW attack.

It is noteworthy that MCW attack is the weakest of the three (it drops the undefended accuracy less) but is harder to defend against and provides the best generalization. A likely explanation for this behavior is that the MPGD and MAPGD patches have high pixel values and produce more distinct patterns from natural images than the MCW patches. A detector that can detect subtle patches generalizes to more distinct patches.

Computation Overhead: We analyze the memory cost and inference speed of PatchZero in Table 7. Both PatchZero and PatchCleanser use ResNet50 as backbone. Although Patchzero has more model parameters, it has a faster (60x) inference speed and lower (2x) GPU memory.

DO vs. BPDA Attack Patterns: We compare the DO (top) and BPDA adaptive attack patterns (bottom) in Figure 4. The left column shows adversarial patches in the image classification task. The BPDA attack pattern seems to be more colorful and granular than the DO attack pattern. For the object detection task (middle), BPDA attack patterns are more structured, rather than a seemingly random appearance. For the video classification task (right), the DO patch shows some "grid-like" pattern, while the BPDA patch is denser and more granular.

In all cases, the two types of attack patterns are very dif-

(a) Dog, Car   (b) TaiChi, YoYo   (c) BlowDryHair, Haircut

Figure 5: **Three common failure cases of PatchZero.** (a) miss patch detection, (b) leaking adversarial pixels, (c) occlusion. Green denotes ground truth labels and red denotes incorrect predictions. Figure best viewed when zoomed in.

ferent. The BPDA attack patterns start with the DO patterns and gradually evolve into the granular patterns. The changing appearance requires the patch detector to be updated at each iteration. It is not at all obvious that the process should converge but, thanks to the two-stage training, the trained patch detector becomes robust against the BPDA patches.

Failure Cases and Limitations: We present three common failure cases of PatchZero in Figure 5. As shown in part (a), missed patch detection can lead to defense failure, since repainting cannot be effectively applied without a correct patch detection. In the example shown, the adversarial patch has a similar texture as the background, leading to a missed detection. Leaking adversarial pixels in (b) is another failure case though morphological operations applied to the binary mask prediction can reduce the effects. Final failure cases in part (c) arises due to significant occlusion caused by random patch location falling on top of the main object in the scene, regardless of correct patch detection.

5. Conclusions

In this paper, we proposed PatchZero, a general defense pipeline against white-box patch attacks. PatchZero first detects the adversarial pixels and then "zeros out" the patch region by repainting with mean pixel values. We further propose a two-stage training scheme to defend against the stronger adaptive attacks. Extensive experiments demonstrate the state-of-the-art robustness of PatchZero across the tasks of image classification, object detection, and video classification, with little degradation in benign performance. PatchZero transfers well to different patch shapes and attack types.

Acknowledgment

References

[1] Adithya Prem Anand, H Gokul, Harish Srinivasan, Pranav Vijay, and Vineeth Vijayaraghavan. Adversarial patch defense for optical flow networks in video action recognition. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1289–1296. IEEE, 2020.

[2] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, pages 484–501. Springer, 2020.

[3] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, 2018.

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

[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] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

[7] Ping-yeh Chiang, Michael J Curry, Ahmed Abdelkader, Aounon Kumar, John Dickerson, and Tom Goldstein. Detection as regression: Certified object detection by median smoothing. *arXiv preprint arXiv:2007.03730*, 2020.

[8] Ping-yeh Chiang, Renkun Ni, Ahmed Abdelkader, Chen Zhu, Christoph Studer, and Tom Goldstein. Certified defenses for adversarial patches. *arXiv preprint arXiv:2003.06693*, 2020.

[9] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[11] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. A study of the effect of jpg compression on adversarial images, 2016.

[12] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

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

[14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[15] Sven Gowal, Krishnamurthy Dj Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4842–4851, 2019.

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

[17] Till Halbach. The h. 264 video compression standard. In *Proceedings of 6th Nordic Signal Processing Symposium*. Citeseer, 2003.

[18] Jamie Hayes. On visible adversarial perturbations & digital watermarking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1597–1604, 2018.

[19] Jamie Hayes. On visible adversarial perturbations & digital watermarking. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1678–16787, 2018.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[21] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[23] Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018.

[24] Danny Karmon, Daniel Zoran, and Yoav Goldberg. Lavan: Localized and visible adversarial noise. In *International Conference on Machine Learning*, pages 2507–2515. PMLR, 2018.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Mark Lee and Zico Kolter. On physical adversarial patches for object detection. *arXiv preprint arXiv:1906.11897*, 2019.

[27] Alexander Levine and Soheil Feizi. (de) randomized smoothing for certifiable defense against patch attacks. *arXiv preprint arXiv:2002.10733*, 2020.

[28] Bin Liang, Jiachun Li, and Jianjun Huang. We can always catch you: Detecting adversarial patched objects with or without signature, 2021.

[29] Jiang Liu, Alexander Levine, Chun Pong Lau, Rama Chellappa, and Soheil Feizi. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14973–14982, 2022.

[30] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299*, 2018.

[31] Shao-Yuan Lo and Vishal M Patel. Defending against multiple and unforeseen adversarial videos. *arXiv preprint arXiv:2009.05244*, 2020.

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

[33] Shao-Yuan Lo, Jeya Maria Jose Valanarasu, and Vishal M Patel. Overcomplete representations against adversarial videos. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 1939–1943. IEEE, 2021.

[34] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[35] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019.

[36] Muzammal Naseer, Salman Khan, and Fatih Porikli. Local gradients smoothing: Defense against localized adversarial attacks. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1300–1307. IEEE, 2019.

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

[38] Sukrut Rao, David Stutz, and Bernt Schiele. Adversarial training against location-optimized adversarial patches. In *European Conference on Computer Vision*, pages 429–448. Springer, 2020.

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[40] Aniruddha Saha, Akshayvarun Subramanya, Koninika Patil, and Hamed Pirsiavash. Role of spatial context in adversarial robustness for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 784–785, 2020.

[41] Ali Shafahi, Mahyar Najibi, Zheng Xu, John Dickerson, Larry S Davis, and Tom Goldstein. Universal adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5636–5643, 2020.

[42] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766*, 2017.

[43] Yixiang Wang, Jiqiang Liu, Xiaolin Chang, Jianhua Wang, and Ricardo J Rodríguez. Di-aa: An interpretable white-box attack for fooling deep neural networks. *arXiv preprint arXiv:2110.07305*, 2021.

[44] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[45] Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. Patchguard: A provably robust defense against adversarial patches via small receptive fields and masking. In *30th {USENIX} Security Symposium ({USENIX} Security 21)*, 2021.

[46] Chong Xiang, Saeed Mahloujifar, and Prateek Mittal. Patchcleanser: Certifiably robust defense against adversarial patches for any image classifier. *arXiv preprint arXiv:2108.09135*, 2021.

[47] Chong Xiang and Prateek Mittal. Detectorguard: Provably securing object detectors against localized patch hiding attacks. *arXiv preprint arXiv:2102.02956*, 2021.

[48] Chong Xiang and Prateek Mittal. Patchguard++: Efficient provable attack detection against adversarial patches. *arXiv preprint arXiv:2104.12609*, 2021.

[49] Zhanyuan Zhang, Benson Yuan, Michael McCoyd, and David Wagner. Clipped bagnet: Defending against sticker attacks with clipped bag-of-features. In *2020 IEEE Security and Privacy Workshops (SPW)*, pages 55–61. IEEE, 2020.

[50] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[51] Guangzhi Zhou, Hongchao Gao, Peng Chen, Jin Liu, Jiao Dai, Jizhong Han, and Ruixuan Li. Information distribution based defense against physical attacks on object detection. In *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2020.

[52] Guangzhi Zhou, Hongchao Gao, Peng Chen, Jin Liu, Jiao Dai, Jizhong Han, and Ruixuan Li. Information distribution based defense against physical attacks on object detection. In *2020 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–6, 2020.