CERTIFIED ROBUSTNESS AGAINST PHYSICALLY-REALIZABLE PATCH ATTACKS VIA RANDOMIZED CROPPING

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper studies a certifiable defense against adversarial patch attacks on image classification. Our approach classifies random crops from the original image independently and the original image is classified as the vote over these crops. This process minimizes changes to the training process, as only the crop classification model needs to be trained, and can be trained in a standard manner without explicit adversarial training. Leveraging the fact that a patch attack can only influence some pixels of the image, we derive certified robustness bounds on the resulting classification. Our method is particularly effective when realistic physical transformations are applied to the adversarial patch, such as affine transformations. Such transformations occur naturally when an adversarial patch is physically introduced to a scene. Our method improves upon the current state of the art in defending against patch attacks on CIFAR10 and ImageNet, both in terms of certified accuracy and inference time.

1 INTRODUCTION

Despite their incredible success in many computer vision tasks, deep neural networks are known to be sensitive to adversarial attacks; small perturbations to an input image can lead to large changes in the output. A wide range of defenses against adversarial attacks have been conducted in image classification, where the goal of the attacker is simply to change the predicted label(s) of an image (Kurakin et al., 2016a; Szegedy et al., 2013; Madry et al., 2017). But these defenses have typically considered a relatively unrealistic threat model that does not easily extend to the physical setting. In particular, these works have mainly considered the so-called ℓ_p-norm threat model, where an attacker is allowed to perturb the intensity at all pixels of the input image by a small amount. In contrast, *adversarial patch* attacks are considered as physically-realizable alternatives, modeling scenaria where a small object is placed in the scene so as to alter or suppress classification results (Brown et al., 2017). Here, the attack is spatially compact, but can change the pixel value to any value within an allowable range.

This paper develops a practical and provably robust defense against patch attacks. Inspired by the randomized smoothing defense (Cohen et al., 2019; Levine & Feizi, 2019) for the ℓ_p-norm threat model, our approach classifies randomly sampled sub-regions or crops of an image *independently* and outputs the majority vote across these crops as the class prediction of the input image. This approach has numerous benefits. First, given the size of adversarial patches, we can compute the probability of a sampled crop overlapping with the attacked region (patch), and use this probability to determine if the classification outcome of an image can be guaranteed (certified) to not be changed by any adversarial patch. Second, this approach is highly practical, as the crop classifier can be trained using standard architectures such as VGG (Simonyan & Zisserman, 2014) or ResNet (He et al., 2016) without the need for adversarial training. Indeed, random cropping is already a common data augmentation strategy for training machine learning models, and thus the method can be trained via standard techniques. This is different from most existing work on certifiable defenses against patch attacks (Levine & Feizi, 2020; Xiang et al., 2020; Chiang et al., 2020) which either need extra computation for certification during training. Third, the proposed approach separates the training procedure from the patch threat model, and makes our method more robust against numerous variations of patch attacks, for example, patch transformations, such as rotation in x-y plane

	CIFAR10 2.4% patch		ImageNet 2.0% patch	
	certification acc. (clean acc.)	time in ms	certification acc. (clean acc.)	time in ms
Proposed method	**47.5** (**88.4**)	**0.7**	**12.2** (**55.7**)	**21.8**
De-rand. smoothing	17.5 (83.9)	17.5	3.2 (43.1)	703.2
PatchGuard: De-rand. smoothing	18.2 (84.5)	18.2	3.5 (43.6)	734.5
PatchGuard: Bagnets	27.1 (82.6)	**0.7**	9.6 (54.4)	25.7

Table 1: Worst-case certified accuracy (%), clean accuracy(%), and certification time of the proposed method, De-randomized smoothing (Levine & Feizi, 2020), and PatchGuard (Xiang et al., 2020) with De-randomized smoothing and Bagnets as base structure. For each method, we list the *worst* certified accuracy under affine transformation of the patch. *Note that this is different from results in the original paper where patch transformations are not considered.*

and aspect ratio changes. This is in sharp contrast to prior work that either have fixed strategies to exclude parts of the image or extract features from fixed parts of the image (Levine & Feizi, 2020; Xiang et al., 2020).

We summarize our main results on CIFAR10 and ImageNet in Table 1 in comparison with the current state of the art certifiable defense against patch attack (Xiang et al., 2020). We report certified accuracy, which is the ratio of images with which classification outcome equals ground truth labels and is guaranteed to not change under patch attack. Our method is better in both speed and certified accuracy compared to De-randomized smoothing (Levine & Feizi, 2020) and PatchGuard (Xiang et al., 2020) under patch attack with possible affine transformation of the patch. In addition, our method outperforms these past approaches on ImageNet (though not on CIFAR10) in the setting where the patch aligns with coordinates of the image axes and cannot undergo these transformations as in Table 2, which was the setting considered in this past work.

We have made several contributions in this paper: first, we propose a defense against patch attack for image classification with certified robustness; second, the proposed method is fast in computing image certification and robust against patch transformation; third, the proposed method can be applied to any image classification models with only minimal changes to the training process.

2 Background and related work

Adversarial attacks Adversarial attacks on image classification have been known for some time, with original work coming out of the field of robust optimization (Ben-Tal et al., 2009). Test-time attacks on ML models in general were studied in (Dalvi et al., 2004; Biggio et al., 2013), though the area gained momentum considerably when these methods were applied to deep learning systems to demonstrate that deep classifiers could be easily fooled by imperceptible changes to images (Szegedy et al., 2013; Goodfellow et al., 2014). In the following years many defenses against such attacks were proposed (Tramèr et al., 2017; Papernot et al., 2016), though most heuristic approaches were later found to be ineffective (Athalye et al., 2018). Amongst the defense strategies that have stood the test of time are: 1) adversarial training (Goodfellow et al., 2014; Kurakin et al., 2016b; Madry et al., 2017), now commonly carried out using a projected gradient descent based approach for synthesizing adversarial attacks and then incorporating them into the training; and 2) provably robust training (Wong & Kolter, 2017; Raghunathan et al., 2018). Our approach is more related to the randomized smoothing-based methods (Cohen et al., 2019; Levine & Feizi, 2019) in the latter direction, in which random points around the original input are sampled and classified, and the predicted class of the original input is declared as the aggregation of these outputs.

The majority of the attacks mentioned above focus on attacks with bounded ℓ_∞-norm, where attacks are permitted to modify any pixel in the image by (at most) some fixed amount (and are usually further permitted to design a new adversarial perturbation for each new input image, though so-called "universal" ℓ_∞ attacks have been studied as well (Moosavi-Dezfooli et al., 2017)). If we

consider real-life attackers, however, this level of freedom on the attacker's side seems uncommon: attackers would need to have access to the deep learning system at the pixel level, which if true, also means that the system has already been compromised, making it unclear what benefit such an attack would have. In contrast, *physical* attacks such that can be implemented in the real world have focused primarily on the so-called patch attacks (Brown et al., 2017; Eykholt et al., 2018), where a particular 'pattern' is designed that can fool a deep learning system. We consider patch attack as the threat model to defend against for it being physically realizable.

Defense against patch attacks There are also two threads of research in defending against patch attack: 1) empirical defense strategies which show stronger empirical robustness but no analytical guarantees (McCoyd et al., 2020; Naseer et al., 2019; Hayes, 2018); most of these methods rely on the fact that a patch attack has to produce strong local features that are very different from its neighborhood to change classification outcome, and by analyzing local features, one can detect patch location or smooth out such feature values to reduce effectiveness of the patch, and 2) certified defense which provides analytical lower bound of classification accuracy under patch attack.

Certifiable defense against patch attacks Relevant to our work is that which provides provable lower bounds of classification accuracy. In Chiang et al. (2020), interval bound propagation was used to compute certification, but the method is not applicable to commonly used image classification networks such as ResNet50 or VGG16 because the depth of these structures made the bound trivial and cannot provide certification. Another line of research utilizes the fact that an adversarial patch can only have local influence on the image, and to reduce such influence, Levine & Feizi (2020) uses ablation to exhaustively block all possible patch locations while Zhang et al. (2020) uses Bagnets (Brendel & Bethge, 2019) with limited receptive field. Both methods then aggregate logits from ablated versions of images or local pixels to the final image classification. Xiang et al. (2020) adds additional masking which goes through logits of all classes and selectively mask out logits that are unusually high. However all three methods require special image classifier structures – Levine & Feizi (2020) require three additional channels to express ablation and Zhang et al. (2020) is limited to Bagnets, and Xiang et al. (2020) uses either of the former structures. Our proposed method instead can use *any image classification structure* and does not require any additional steps to compute certification during model training except cropping images. This makes our method much more practical and easier to deploy onto existing systems.

3 Provable patch defense with randomized cropping

A physically realizable adversarial patch attack can be found by solving the optimization problem:

$$\max_{\delta \in \Delta} \mathbf{E}_{(x,y)\sim D, t\sim T} \left[\ell(h_\theta(A(x, \delta, t)), y) \right] \tag{1}$$

where $h_\theta : \mathcal{X} \to \mathcal{Y}$ denotes some (presumably deep-network-based) hypothesis function; θ denotes parameters of the model, $x \in \mathcal{X}$ denotes the original input to the network and $\delta \in \mathcal{X}$ the perturbation to the input; $\Delta(x) \subseteq \mathcal{X}$ denotes the set of allowable perturbations, which may depend on the input; $y \in \mathcal{Y}$ denotes the true label; $\ell : \mathcal{Y} \times \mathcal{Y} \to R$ denotes a loss function that measures the performance of the image classifier; and $\mathcal{T} : \mathcal{X} \to \mathcal{X}$ is the set of transformations that the perturbation δ might go through. Δ denotes a simple allowable set of values for the patch, which typically would just be constrained to lie in valid RGB space. Patches *overwrite* a portion of the image with the patch perturbation itself, at a given location with a given set of transformations, such as scaling, rotation, and other transformations in T. We refer to this combination as the patch application function $A : \mathcal{X} \times \mathcal{X} \times \mathcal{T} \to \mathcal{X}$, where $A(x, \delta, t)$ denotes the application of patch δ to image x with transformation t. Attack budget considered for patch attack is the percentage of area that the patch covers over the whole image. Throughout this paper, we consider the patch being one connected area bounded by a rectangle with size p_i by p_j.

3.1 Randomized cropping defense

Although a patch attack can change pixel values to any arbitrary value, it can only influence the pixels within the patch itself. Therefore networks extracting features with compact receptor fields and aggregating such local features for final classification are more robust against patch attacks. An

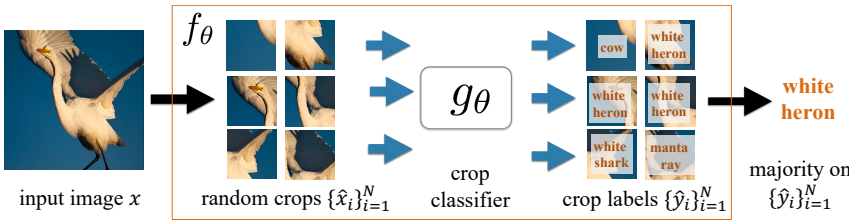

Figure 1: Forward pass of randomized crop defense

example of such methods is Bagnets (Brendel & Bethge, 2019), where 1x1 convolutional kernels are used to restrict the size of the receptor fields. Several existing techniques (Zhang et al., 2020; Xiang et al., 2020) utilize Bagnets for adversarial defense. However, the major drawback of Bagnets is that it extracts local features at fixed locations, therefore a patch may be placed in particular locations to influence more number of local features than other locations, thus reducing the worst-case robustness of Bagnets. In contrast to this architecture, we propose to extract local features at *random* locations by sampling random crops; this greatly reduces variations in the number of local features an adversarial patch can influence.

Based on above observations, we propose a randomized cropping approach as shown in Fig. 1: given a full-size image x as input, we first randomly select n crops, where crop size $k_i \times k_j$ is smaller than image size $m_i \times m_j$ in both dimensions. Each of the sampled crops ($\widehat{x_i}$) then goes through the crop-based classifier g_θ and its predicted class ($\widehat{y_i}$) is obtained. The final classification of x is the majority of $\{\widehat{y_i}\}_{i=1}^n$.

Training randomized cropping classifier It is clear from Fig. 1 that the only trainable part of the randomized cropping classifier is the crop classifier g_θ, which can be trained in standard ways with only data augmentation of randomly cropping the input x as in Algorithm 1. Other variables are either pre-determined hyperparameters (crop size k_i, k_j), or can be adjusted at test time (number of crops n). This is because we use only the *predicted class labels* $\widehat{y_i}$ of the crops instead of logits as in (Xiang et al., 2020; Levine & Feizi, 2020; Zhang et al., 2020).

Since the training procedure of randomized cropping classifier f_θ only requires training the crop classifier g_θ without any assumption on attack parameters (size, shape, and location) nor does it depend on hand-crafted rules or thresholds, our method is robust against different patch shapes. We show supportive experimental results in Section 4 for this claim.

Algorithm 1: Training crop classifier for T epochs given crop size $k_i \times k_j$, image classifier g_θ with loss function ℓ, and a dataset of M images

Input: Full size image data x, labels y
for t = *1* ... *T* **do**
  for i = *1* ... *M* **do**
    $\widehat{x_i}$ = a $k_i \times k_j$ crop of x_i at random location;
    $\theta = \theta - \nabla_\theta \ell(f_\theta(x_i), y_i)$ *//Update model weights with some optimizer, e.g. SGD*
  end
end

3.2 Certifiable Robustness Bound

To compute the probability that an image x is certified with randomized cropping defense, i.e., the classification outcome of x cannot be changed by patch attack of given size, we first compute the probability that one sampled crop contains at least one pixel from the adversarial patch, and use this probability to compute the probability that x is certifiably robust.

Probability of a sampled crop to have overlaps with the adversarial patch can be computed as the total number of crops overlapping with the patch (denoted by n_{adv}) divided by the number of all possible crops (n_{all}). Although a physically-realizable adversarial patch can be at any location in the image, to guarantee robustness, we consider the worst-case scenario where the patch is at the

	CIFAR10			ImageNet		
Patch Size	ours	DRS	PG+DRS / Bagnets	ours	DRS smoothing	PG+DRS / Bagnets
0.4%	65.7	68.9	**69.2** / 53.2	24.7	22.3	**24.8** / 23.1
1.0%	60.2	62.7	**65.3** / 41.2	**20.1**	17.7	19.9 / 18.6
2.0%	55.2	60.9	**61.1** / 37.2	**16.4**	14.0	16.0 / 13.3
2.4%	52.3	57.1	**58.1** / 31.7	**15.3**	13.1	14.7 / 11.2
3.0%	37.8	42.1	**43.5** / 25.1	**14.2**	11.2	13.01 / 8.9

Table 2: Certified accuracy (%) over CIFAR10 and ImageNet of our method, De-randomized smoothing (DRS) (Levine & Feizi, 2020), and PatchGuard (PG)(Xiang et al., 2020)

location that it can influence maximum number of crops (usually at the center of the image). When the crop is at the center of the image,

$$n_{adv} = \min(p_i + k_i - 1, m_i - k_i + 1) \times \min(p_j + k_j - 1, m_j - k_j + 1), \tag{2}$$

$$n_{all} = (m_i - k_i + 1) \times (m_j - k_j + 1), \text{ and} \tag{3}$$

$$p_a = \frac{n_{adv}}{n_{all}} \tag{4}$$

Then, assuming we sample n crops out of the original input x and run them independently through the crop classifier g_θ, this results in $\{\widehat{y}_i\}_{i=1}^n$ crop-based predicted classes. Suppose in the n crop-based predicted classes $\{\widehat{y}_i\}_{i=1}^n$, n_1 of them are the same as the majority class c_1, and n_2 of them are the same as the second-majority class c_2. If an adversarial patch changes the majority class, the minimal number of crop-based predicted classes that it has to change will be

$$n_{2to1} = \frac{\lfloor n_1 - n_2 \rfloor}{2} + 1 \tag{5}$$

to make c_2 the majority class.

In order to change predicted classes of n_{2to1} crops, these n_{2to1} crops must all overlap with the adversarial patch. On the other hand, if there are fewer than n_{2to1} crops overlapping with the adversarial patch, then the image is certified to be robust with probability

$$p_c = \sum_{i \in \{0, n_{2to1}\}} C_i^N * p_a^i * (1 - p_a)^{(n-i)}. \tag{6}$$

If p_c is close to 1.0, the input image is certifiably robust under patch attack. Throughout this paper, we consider *certified accuracy*, meaning only data with predicted class c_1 being the ground truth class will be considered for certification.

The above computation of probability of certification in Eq. 6 again shows that p_c is monotonically increasing with n_{2to1}. So to maximize number of certified robust images, the randomized cropping classifier should maximize n_{2to1}, which is equivalent to maximizing classification accuracy of g_θ. Hence our training procedure is decoupled from variations of patches.

4 EXPERIMENTAL RESULTS

Dataset, model, and device We conduct experiments on two benchmark image classification datasets: CIFAR10 and ImageNet. For both datasets we evaluate over the whole evaluation set and report certified accuracy as percentage of images that are classified correctly and can be certified with probability higher than 0.95. For CIFAR, we use ResNet9 without the final pooling layer as the crop model, and for ImageNet we use ResNet34 as the crop model and resize and pad the image to 224x224. Experiments and timing are done on single Nvidia 2080 Ti GPU, and 16-core Intel i7-5960X CPU.

Positional encoding Because both CIFAR10 and ImageNet images have a certain region of interest which is usually in the center of the image, some parts of the images contain more information than others. For example, an image with label airplane usually has the airplane in the center, while the corners and edges have the background of sky or airport. Therefore, crops sampled at different position of the image contains different information. To represent such information, we add learnable positional encoding (Vaswani et al., 2017) to the first layer of our classifier.

4.1 WITHOUT PATCH TRANSFORMATION

We first present our main results in Table 1 and Table 2: we chose crop size of 10x10 (10% total area) for CIFAR and 80x80 (13% total area) for ImageNet, and for these experiments we assume patch shape is square with no rotation. For each input image in CIFAR-10, we randomly sample 128 crops and for each image in ImageNet we sample 256 crops. Position encoding used here is learnable position encoding with values up to 10% of normalized input value range. We mainly compare our results with two prior arts: De-randomized smoothing (DRS) (Levine & Feizi, 2020) and PatchGuard (PG) (Xiang et al., 2020). For all PatchGuard results there are two sets of numbers separated by slash: the first set of numbers use De-randomized smoothing(Levine & Feizi, 2020) as base structure and the second set of numbers use Bagnets (Brendel & Bethge, 2019).

Table 1 shows clean model accuracy and inference time per image compared with the above mentioned prior work. In terms of inference time our method is faster under all cases. This is because although multiple forward passes are needed, the number of crop samples is smaller than equivalent sub-regions in Bagnets and much smaller than number of all band-smoothed images as in De-randomized smoothing. PatchGuard with Bagnets is slower than our method in ImageNet for the additional reason of having to go through all logits of all possible classes sequentially. In addition, both De-randomized smoothing and PatchGuard with De-randomized smoothing the base structure are *an order of magnitude* slower than our method because De-randomized smoothing passes the whole image through the classifier, which is much more expensive then passing crops (10% or 13% of the whole image) through the classifier. Although using Bagnets as base structure makes PatchGuard much faster than using De-randomized smoothing, it pays the price of certified accuracy with only 31% for CIFAR10 and 13.3% for ImageNet. On the other hand, on CIFAR10 our randomized cropping defense is as fast as PatchGuard with Bagnets, but achieves much better (52%) certified accuracy. Moreover, on ImageNet our proposed method is the fastest and with highest certified accuracy as well as clean accuracy.

Table 2 shows a more detailed comparison of our randomized cropping method with De-randomized smoothing and PatchGuard on patch size ranging from 0.4% to 3%. Methods with highest certified accuracy are highlighted in bold. We can see that although our method is sub-optimal for CIFAR10, for ImageNet we have the highest certified accuracy except for patch size 0.4%

Combining the certified accuracy on ImageNet in Table 2 and the short inference time in Table 1, our experiments show that the proposed randomized cropping defense is *practical in the aspects of fast certification and high certified accuracy* for the dataset that is closer to real-life pictures.

4.2 WITH PATCH TRANSFORMATION

As described in Section 3, a physically-realizable patch can be subject to certain transformations, such as scaling and rotation, before overwriting pixels in an image. These transformations can be used to mimic relative camera angle and distance to the scene in the real physical world. While it is reasonable to restrict the maximal area that an adversarial patch can influence in an image (since it corresponds to how big the physical patch is), one cannot assume such area would always align with coordinate axes of the image – even if the physical patch itself is square, when the scene is captured with different camera angles, the patch on the image will be rotated (rotation in x-y plane) or stretched (rotation in depth). Therefore it is important that the patch defense can also provide guarantees when the patch is rotated or with different aspect ratio, given the same patch size.

In Table 3 we compare the proposed randomized cropping defense with De-randomized smoothing (Levine & Feizi, 2020) and PatchGuard (Xiang et al., 2020) when a square patch is rotated 45 degrees and with aspect ratio (AR) 1:1, 6:1, 2.7:1, 1:2.7, 1:6, respectively. We highlight the highest certified accuracy under worst transformation for both datasets. Aspect ratio 1:1 without rotation is

	CIFAR10 2.4% patch			ImageNet 2% patch		
Transformation	ours	DRS	PG+DRS / Bagnets	ours	DRS smoothing	PG+DRS / Bagnets
AR 1:1	52.3	57.1	58.1 / 31.7	16.4	14.0	16.0 / 13.3
AR 2.7:1	50.7	65.8	67.2 / 30.4	15.8	15.2	17.6 / 12.4
AR 6:1	47.5	71.1	74.5 / 27.1	12.2	17.9	19.0/9.6
AR 1:2.7	50.7	40.6	42.4 / 30.4	15.8	11.3	11.9 / 12.4
AR 1:6	47.5	17.5	18.2 / 27.1	12.2	3.2	3.5 / 9.6
Rotate 45°	48.0	50.3	52.1 / 28.1	12.4	12.1	12.8 / 10.1
Worst case	**47.5**	17.5	18.2/27.1	**12.2**	3.2	3.5/9.6

Table 3: Certified accuracy (%) under patch rotation and different aspect ratio

the same as Table 2 but listed here as reference. We chose 45 degrees because it is the worst case for a square patch, as the same number of pixels within the patch but occupying the largest rectangle area that aligns with coordinate axes of the image. Therefore rotation of 45 degrees provides a lower bound of robustness under patch rotation. For all transformations of the patch, we consider the same patch size.

We can clearly see that in Table 3, with the same patch size, transformation brings down the certified accuracy of all three competing methods, however, the proposed randomized cropping defense has the highest certified accuracy under worst-case patch transformation as listed in the last row of Table 3. This is because of our *random sampling* strategy that we have neither fixed locations of sub-regions as in Bagnets, nor fixed smoothing strategy as in De-randomized smoothing.

On the other hand, the fixed column-smoothing strategy (ablating columns of images) in De-randomized smoothing (Levine & Feizi, 2020) which has highest certifiable accuracy for square patches aligning coordinate axes of the image suffers particularly because as the patch gets shorter and longer (aspect ratio 1:3 and 1:6), the number of columns needed to ablate all the patch gets higher, and left fewer pixels of the image for classification. Also rotating the patch 45 degrees is equivalent to increasing the number of columns the patch occupies by 1.4 times, and hence significantly reduces certified accuracy for column-smoothing strategy. While there are other smoothing techniques in De-randomized smoothing (block smoothing), similar effects can be expected as some aspect ratio or rotation degree will cause significant degradation of certified accuracy. Same level of degradation is also seen in PatchGuard with De-randomized smoothing base structure (numbers in PatchGuard column before slash symbol).

Clean and certified accuracy with or without positional encoding, with regards to different crop sizes are shown in Figure 2 – we use square crops and square patches aligning with coordinate axes of the image. Comparing clean accuracy with and without positional encoding, we can see that although clean accuracy still increases as crop size increases when crop classifier includes positional encoding, but not as much as without positional encoding. Such results show positional encoding does provide extra information for crops sampled from different locations. On the other hand, comparing clean accuracy with certified accuracy, it is clear that certified accuracy actually gets lower when crop size crosses some threshold. This is because with larger crops, probability of the patch overlaps the sampled crop increases.

Finally we show certified accuracy and inference time with different number of sampled crops. Certified accuracy and inference time of using all possible crops are added as reference. With more crops sampled, the certified accuracy increases but however inference time also increases almost linearly. Therefore we chose relatively small number of crop samples.

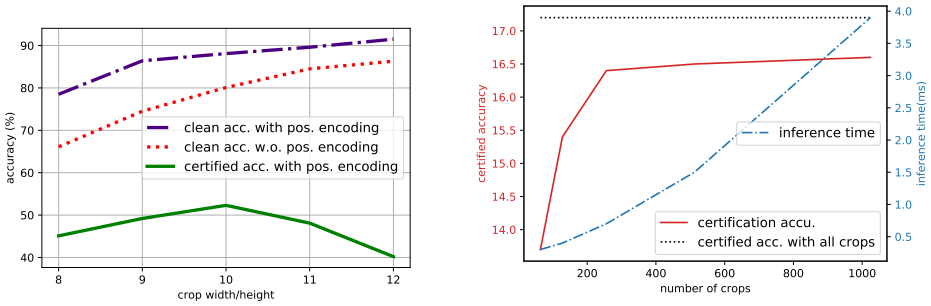

Figure 2: Clean and certified accuracy with different crop sizes. Left: CIFAR10 with 2.4% patch. Right: ImageNet with 2.0% patch

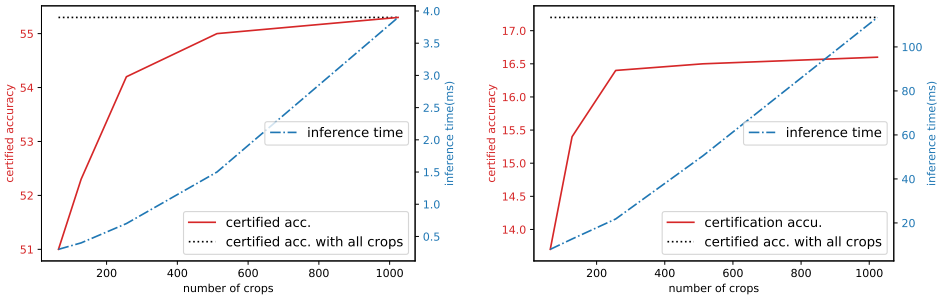

Figure 3: Certified accuracy and inference time(ms) with different number of sampled crops. Left: CIFAR10 with 2.4% patch. Right: ImageNet with 2.0% patch

5 Conclusion

This paper proposes a new architecture for defense against physically-realizable patch attacks. The proposed approach decomposes an image into a random assortment of crops, each of which is processed by a classifier, and the majority across the crops is used as the classification outcome for the input image. This approach provides a significant advance in the form of improved certified accuracy, while maintaining a high clean accuracy when compared to prior art. We have shown that the proposed approach can be easily incorporated into standard training and test pipelines with minimal change to the underlying codebase. We believe this approach that relies on simple and effective architectural change provides a compelling alternative to traditional defense strategies and, in this sense, advances robust machine learning.

References

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018.

Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*, volume 28. Princeton University Press, 2009.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 387–402. Springer, 2013.

Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

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

Ping-Yeh Chiang, Renkun Ni, Ahmed Abdelkader, Chen Zhu, Christoph Studor, and Tom Goldstein. Certified defenses for adversarial patches, 2020.

Jeremy M Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing, 2019.

Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, Deepak Verma, et al. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 99–108. ACM, 2004.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2018. doi: 10.1109/cvpr.2018.00175.

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

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

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*, pp. 770–778, 2016.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016a.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale, 2016b.

Alexander Levine and Soheil Feizi. Wasserstein smoothing: Certified robustness against wasserstein adversarial attacks, 2019.

Alexander Levine and Soheil Feizi. (de)randomized smoothing for certifiable defense against patch attacks, 2020.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Michael McCoyd, Won Park, Steven Chen, Neil Shah, Ryan Roggenkemper, Minjune Hwang, Jason Xinyu Liu, and David Wagner. Minority reports defense: Defending against adversarial patches. *arXiv preprint arXiv:2004.13799*, 2020.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. doi: 10.1109/cvpr.2017.17.

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)*, pp. 1300–1307. IEEE, 2019.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 582–597. IEEE, 2016.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples, 2018.

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

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.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2017.

Eric Wong and J Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. *arXiv preprint arXiv:1711.00851*, 2017.

Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. Patchguard: Provable defense against adversarial patches using masks on small receptive fields. *arXiv preprint arXiv:2005.10884*, 2020.

Zhanyuan Zhang, Benson Yuan, Michael McCoyd, and David Wagner. Clipped bagnet: Defending against sticker attacks with clipped bag-of-features. 2020.