Multi-Orientation Adversarial Patch

Nitay Yakoby Kalanit Segal Noy Gabay Shay Sitri

Colab Notebook at [5]

1 Introduction

Deep learning systems are highly capable, yet they remain vulnerable to manipulations known as adversarial examples. These adversarial examples consist of specially crafted inputs designed to deceive the system into making classification errors, all without noticeable changes to human observers. Such adversarial attacks often involve subtle, nearly imperceptible adjustments to each pixel, utilizing various optimization strategies like L-BFGS, the Fast Gradient Sign Method (FGSM), and Projected Gradient Descent (PGD). Among the methods, the adversarial patch attack emerges as a distinct approach, where a purposefully designed image, when placed within a scene or on an object, leads the model to misclassify its observations. The patch works by exploiting the model's vulnerabilities, effectively overriding its normal decision-making process with false data. This report introduces a new approach, aiming to conduct a targeted adversarial patch attack. A targeted attack specifically aims to manipulate a model's output to misclassify an input as a predetermined incorrect class, rather than merely causing a random classification error. In our work, the adversarial patch is not only designed to trick the model under standard conditions but also engineered to alter its target class when rotated by 90 degrees. This means that the same patch can be used to trigger different specific misclassifications depending on its orientation, significantly enhancing the versatility and threat level of the attack. In this report, we will present the two methodologies employed to create these adversarial patches. Furthermore, we will extend our exploration beyond the initial 90-degree rotation, investigating the effects of varying the patch's orientation at additional degrees—such as 30, 60, and a complete 360-degree rotation. These expansions aim to thoroughly understand and demonstrate the full potential and versatility of adversarial patch attacks in real-world applications. Finally, we conducted experiments to assess the effectiveness of our suggested approaches in creating and deploying patches, by testing the performance of attacks with patches of various sizes, shapes and at different rotation angles.

2 Related Work

An adversarial patch attack, introduced by [1], involves creating universal and robust patches that can mislead models in real-world scenarios. The patch is initially generated from random noise, with transformations like scaling and rotation applied to fit specific locations on an image. It is then trained via gradient descent to optimize its effectiveness, ensuring the patch causes the model to misclassify the image into a targeted class. The method's adaptability to various conditions, including black-box models, underscores its robustness and offline applicability.

In contrast to traditional adversarial patches with fixed shapes, recent research [2] highlights the importance of patch shape. The Deformable Patch Representation (DPR), using geometric properties like triangles, enables a differentiable mapping between contour modeling and masking. This method, along with the Deformable Adversarial Patch (DAPatch) algorithm, optimizes both shape and texture, significantly improving the attack's success across different datasets and architectures.

Beyond exploring shape and size, our research investigates the role of rotation in targeted adversarial patch attacks. As demonstrated by [4], spatial transformations, particularly rotation, can induce targeted misclassifications in deep learning models. While their work explored rotation as a backdoor trigger, our approach extends this concept by designing a patch that adapts to multiple orientations, leading to different misclassifications at various rotation angles. This enhances the adaptability and stealth of adversarial attacks, making them more effective in real-world applications.

3 Methods

This study investigates the generation of adversarial patches using two distinct approaches. Both methods aim to train a patch capable of effectively misleading the target model into misclassifying an image as a predefined class across various orientations. These adversarial patches are typically small portions of an image, but with strong adversarial perturbations that can significantly alter the model's predictions. Initially, a basic patch is designed as a targeted attack for a specific class X. However, when this patch is rotated by 90 degrees counter clockwise, the nature of the attack transforms, now targeting a different class Y. This approach was expanded further to demonstrate that each additional rotation can introduce new targeted attacks. For instance, two rotations can create a scenario where the patch simultaneously conducts three distinct attacks, complicating the model's ability to correctly classify the image.

3.1 Loss Calculation

The core element shared by both methods is the calculation of the loss function. For each training image, the model predicts the class of the patched image, which is then compared to the designated target class. The Cross-Entropy loss is computed for each rotation of the patch within the image. The cumulative loss across all rotations is defined as:

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{N} \sum_{j=1}^{M} \text{CE}(f(x_i, p, \theta_j), y_j) \tag{1}$$

Where $\mathcal{L}_{\text{total}}$ is the total loss accumulated over all training samples and rotations, N is the number of training samples in a batch, M is the number of rotation angles applied to the adversarial patch, x_i represents the i-th input image, p is the adversarial patch being optimized, θ_j is the rotation angle applied to the patch at step j, $f(x_i, p, \theta_j)$ is the model's output when applying the rotated patch to the image x_i, y_j is the target label associated with the rotation θ_j, and $\text{CE}(\cdot, \cdot)$ is the cross-entropy loss function, measuring the discrepancy between the model's prediction and the target class.

3.2 Method 1: PGD with Optimizer and Zero Initialization

This method utilizes a Stochastic Gradient Descent (SGD) optimizer with momentum to iteratively update the patch parameters. A key advantage of this approach lies in its capacity for fine-grained optimization. The optimizer facilitates controlled and precise adjustments to the patch, enabling it to effectively deceive the model across a range of rotations. Figure 1 illustrates examples of square patches generated using this method. In addition to the classic square patches, we also generated circular patches and conducted experiments with them, based on findings from [2], which demonstrate that the shape and size of patches play a crucial role in the effectiveness of these attacks. Figure 2 displays examples of circular patches created using method 1.

- **Patch Initialization:** The patch is initialized as a zero-valued tensor of size $(3, h, w)$, providing a neutral starting point for the optimization process. h, w represents the patch size.
- **Patch Updates:** The optimizer calculates gradients based on the cumulative loss across all rotations and subsequently updates the patch parameters accordingly.

3.3 Method 2: PGD with Random Initialization and FGSM-Based Update

This method utilizes the Fast Gradient Sign Method (FGSM) for adversarial training [3], offering a computationally efficient and straightforward alternative. The patch is initialized randomly, and updates are computed directly by taking the sign of the gradient of the loss function with respect to the patch. This approach avoids the need for an iterative optimization process, reducing computational overhead while maintaining effectiveness.

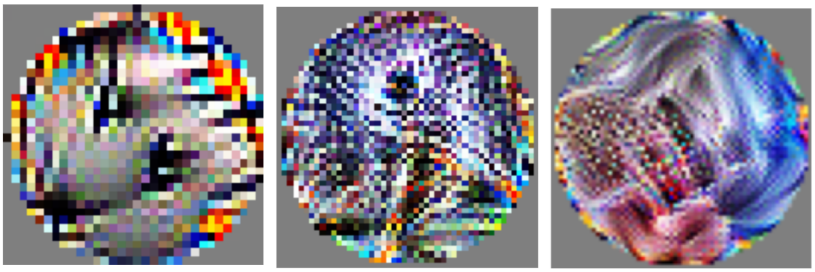
The magnitude of the updates is controlled by the parameter ϵ, which determines the maximum perturbation allowed in each step. By carefully choosing ϵ, the method ensures that the perturbations are strong enough to mislead the model effectively while avoiding excessive distortions to the original image. A relatively high ϵ value is used in this study to enable the creation of more pronounced and impactful patterns, as stronger perturbations are critical for generating patches that remain effective across various rotations and orientations of the image. This

(a) Iguana, Tiger
Patch size 32x32

(b) Iguana, Schooner
Patch size 48x48

(c) Tiger, Schooner
Patch size 64x64

Figure 1: Examples of 3 different patches in different sizes, created using PGD with optimizer and Zero initialization.



(a) Forklift, PlasticBag
Patch size 32x32

(b) Limpkin, GasMask
Patch size 48x48

(c) GasMask, PlasticBag
Patch size 64x64

Figure 2: Examples of 3 different circle patches in different sizes, created using PGD with optimizer and Zero initialization.

balance is crucial for ensuring robustness while maintaining the overall integrity of the visual content. Figure 3 shows examples of patches created using this method.

- **Patch Initialization:** The patch is initialized with random values within the range $[0, 1]$. Patch updates are constrained to maintain the patch values within the range $[-\epsilon, \epsilon]$.
- **Patch Updates:** Instead of using an explicit optimizer, updates are performed based on the sign of the gradient.

(a) Iguana, Tiger
Patch size 32x32

(b) Iguana, Schooner
Patch size 48x48

(c) Tiger, Schooner
Patch size 64x64

Figure 3: Examples of 3 different patches in different sizes, created using PGD with random initialization and FGSM-based update.

4 Experiments

Our experiments evaluated the effectiveness of rotation-based adversarial patch attacks using a subset of the Tiny ImageNet dataset. Specifically, we utilized the 200-class subset, where each class contains 500 training images. Image data was normalized with means of 0.485, 0.456, and 0.406 and standard deviations of 0.229, 0.224, and 0.225 for the Red, Green, and Blue channels, respectively. The dataset was split into training (90%) and validation (10%) sets, resulting in 9,500 training images and 500 validation images.

We assessed the attack effectiveness against a pre-trained ResNet34 model, using Top-1 and Top-5 accuracy as our primary metrics. These metrics measure the frequency with which the true label appears within the model's top 1 or top 5 predictions, providing insight into the precision of the model's classifications under attack. To ensure the robustness and stability of the adversarial patch evaluation, the patch is placed at four random locations on each image during testing. This approach accounts for variability in patch placement and provides a more reliable measure of the patch's effectiveness in fooling the model.

Our experiments explored a comprehensive set of configurations, varying patch size (32x32, 48x48, and 64x64), patch shape (square and circular), the number of target classes (controlled by the number of rotations), and the specific rotation angles. Specific configurations for each experiment are detailed in the following subsections. Our analysis is divided into two main sections, one for square patches and one for circular patches, each detailing the experimental setup and results obtained using two distinct patch generation methodologies.

For the patch generation experiment, we selected five random classes for both square and circular patches from the predefined subset. For the 12-rotation experiment, we chose 12 different random classes. In the 12-rotation experiment, the same classes were used for both square and circular patches.

Each experiment was conducted using one of two training methodologies. In Method 1, we initialized the patch as a learnable parameter and optimized it using Stochastic Gradient Descent (SGD) with a learning rate of 0.1 and momentum of 0.8. In Method 2, the patch was initialized randomly and updated iteratively using gradient-based sign updates with a step size of $\alpha = 0.1$ and an ε-clamping constraint of 3.

Additionally, different numbers of epochs were used for each experiment:

- For the baseline experiment: 5 epochs
- For experiments with 2 rotations: 10 epochs
- For experiments with more than 2 rotations: 20 epochs

4.1 Squared Patch

Table 1 presents the average Top1 and Top5 accuracies for attacks involving permutations of specific classes at various rotation counts and patch sizes, evaluated using a square patch configuration. For configurations with fewer than 12 rotations, only the classes common iguana, tiger, Petri dish, schooner, and grey whale were used. Combinations for single rotations included pairs such as common iguana and tiger, while setups with two rotations featured triplets like common iguana, tiger, and Petri dish. For the complex scenario of 12 rotations, a diverse array of classes was utilized, including minibus, slug, European fire salamander, reflex camera, tabby, Siberian husky, komondor, dowitcher, radio, wallaby, packet, and reel.

Generally, we observed that larger patch sizes resulted in higher Top1 and Top5 accuracies for both methods, suggesting that size plays a crucial role in the effectiveness of these attacks. Notably, Method 2 consistently outperformed Method 1 across all configurations. The results also demonstrated that the basic rotational attacks, involving single rotations at 30° and 90°, as well as configurations with two and three rotations, surpassed the performance of the baseline (no rotations). Specifically, among all configurations examined, the one with a single rotation at 90 degrees achieved the best performance, particularly when applied to the largest patch size of 64x64. This indicates that the patches, despite encompassing diverse targeted attacks, successfully deceived the model. However, as the number of rotations increased, the accuracy of the attacks significantly declined, particularly evident in the scenario with 12 rotations where both Top1 and Top5 accuracies experienced substantial decreases.

4.2 Circled Patch

Table 2 presents the average Top1 and Top5 accuracies for attacks involving permutations of specific classes at various rotation counts and patch sizes, evaluated using a circle patch configuration. For configurations with fewer than 12 rotations, only the classes forklift, monarch, limpkin, gas mask, and plastic bag were used. Combinations for single rotations included pairs such as forklift and monarch, while setups with 4 rotations featured foursomes

Rotation	Method	Patch Size					
		32x32		48x48		64x64	
		Top1	Top5	Top1	Top5	Top1	Top5
No Rotations (Baseline)	Method 1	14.66%	36.71%	48.20%	77.58%	74.25%	94.59%
	Method 2	16.54%	42.80%	69.80%	90.88%	96.77%	99.81%
1 Rotation 0°, 30°	Method 1	19.66%	45.56%	32.65%	48.26%	53.13%	79.89%
	Method 2	42.77%	75.99%	51.93%	55.43%	89.31%	98.19%
1 Rotation 0°, 90°	Method 1	54.92%	74.70%	48.56%	57.91%	84.94%	95.06%
	Method 2	51.01%	83.64%	51.95%	61.07%	92.91%	98.83%
2 Rotations 0°, 30°, 90°	Method 1	34.73%	62.01%	38.86%	61.32%	30.25%	47.41%
	Method 2	55.67%	79.77%	43.56%	64.99%	33.55%	47.49%
3 Rotations 0°, 30°, 60°, 90°	Method 1	25.31%	46.11%	29.08%	53.70%	39.73%	64.11%
	Method 2	30.23%	51.80%	36.80%	55.70%	51.54%	69.63%
3 Rotations 0°, 90°, 180°, 270°	Method 1	18.19%	34.35%	20.79%	40.00%	30.95%	54.7%
	Method 2	24.0%1	43.88%	31.76%	50.61%	42.88%	63.72%
12 Rotations 12 Steps of 30°	Method 1	0.01%	0.31%	0.02%	0.23%	1.15%	6.59%
	Method 2	11.49%	22.28%	9.68%	21.81%	22.60%	38.64%

Table 1: Average Top1 and Top5 accuracies for various attacks using a squared path, detailed by rotation angles and patch sizes.

like forklift, monarch, limpkin, and gas mask. For the complex scenario of 12 rotations, we used the same 12 classes from the experiment of the squared patch.

Consistently across the experiments, we found that larger circular patches led to higher Top1 and Top5 accuracies for both methods, underscoring the significant impact of patch size on the success of adversarial attacks. Method 2 outperformed Method 1 in every tested scenario, affirming its superior efficacy in manipulating model outputs. Notably, the rotational attacks, particularly those with one rotation of 30°, demonstrated improved performance over the no-rotation baseline. Among all tested configurations, the configuration involving one rotation achieved the most notable results, especially with the 64x64 patch size, suggesting optimal disruption at this level of complexity. Despite this, there was a clear trend where the effectiveness of the attacks decreased as the number of rotations increased, culminating in a sharp decline in both Top1 and Top5 accuracies in the 12-rotation scenario. This pattern indicates that while initial increases in complexity can enhance the deception, excessively complex manipulations may become counterproductive.

5 Discussion

Our primary objective was to assess whether rotational patches can successfully deceive classification models as part of an adversarial attack. Our findings affirmatively support this hypothesis, evidenced by the consistent decline in model accuracy across different patch configurations and rotational complexities. We expanded our study beyond square patches to include circular patches, which yielded comparably effective results, thereby underscoring the robustness of our approach across different patch geometries. Furthermore, we explored the impact of increasing the number of rotations on the efficacy of the attacks. Our results indicated a decrease in performance with more rotations, a trend that aligns logically with the increased complexity of the attacks. Each additional rotation introduces another targeted class within the patch, effectively increasing the adversarial challenge. This complexity, while initially beneficial in deceiving the model, begins to yield diminishing returns. This pattern suggests that the approach of rotating patches works effectively only up to a certain limit; beyond this point, the additional complexity may introduce too many adversarial cues that make the attacks ineffective. This threshold effect highlights the need for a balanced strategy in the design of adversarial patches, where the degree of rotation and

Rotation	Method	Patch Size					
		32x32		48x48		64x64	
		Top1	Top5	Top1	Top5	Top1	Top5
No Rotations (Baseline)	Method 1	7.39%	20.45%	24.45%	53.23%	43.91%	75.48%
	Method 2	24.65%	46.38%	87.96%	98.25%	99.35%	99.98%
1 Rotation 0°, 30°	Method 1	24.27%	48.13%	32.19%	48.45%	54.39%	77.91%
	Method 2	50.5%	69.39%	55.14%	64.46%	86.03%	96.86%
3 Rotations 0°, 30°, 60°, 90°	Method 1	26.45%	41.60%	23.42%	47.08%	34.53%	58.12%
	Method 2	40.89%	52.17%	43.91%	64.96%	62.65%	78.19%
12 Rotations 12 Steps of 30°	Method 1	0.05%	0.55%	0.01%	0.23%	1.64%	7.57%
	Method 2	1.07%	4.31%	6.87%	16.40%	14.45%	33.35%

Table 2: Average Top1 and Top5 accuracies for various attacks using a circled path, detailed by rotation angles and patch sizes.

complexity must be carefully calibrated to optimize efficacy without overtly alerting the classification system.

Future research could greatly benefit from several avenues of exploration. First, exploring a wider range of patch shapes and sizes is crucial to better understand their impact on deceiving classification models. This could include investigating various geometric configurations such as ellipses, triangles, or custom irregular shapes, alongside experiments varying patch sizes to pinpoint optimal dimensions for effective adversarial attacks. Second, developing machine learning algorithms to automatically optimize patch parameters, such as rotation angles and placement, presents a promising avenue for automating the design of more effective adversarial patches. Finally, and importantly, our current evaluation of patch robustness has been limited to a ResNet34 model. Future work should investigate the transferability of these rotational patches by evaluating their effectiveness against a broader range of model architectures, including, but not limited to, other ResNet variants, VGG, and EfficientNet, to determine the generalizability of our findings and identify potential model-specific vulnerabilities. This will provide a more comprehensive understanding of the robustness of our approach and its potential impact on real-world applications.

References

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

[2] Zhaoyu Chen, Bo Li, Shuang Wu, Jianghe Xu, Shouhong Ding, and Wenqiang Zhang. Shape matters: deformable patch attack. In *European conference on computer vision*, pages 529–548. Springer, 2022.

[3] Yujie Liu, Shuai Mao, Xiang Mei, Tao Yang, and Xuran Zhao. Sensitivity of adversarial perturbation in fast gradient sign method. In *2019 IEEE symposium series on computational intelligence (SSCI)*, pages 433–436. IEEE, 2019.

[4] Tong Wu, Tianhao Wang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Just rotate it: Deploying backdoor attacks via rotation transformation. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*, pages 91–102, 2022.

[5] Nitay Yakoby, Kalanit Segal, Noy Gabay, and Shay Sitri. Multi-orientation adversarial patch. https://colab.research.google.com/drive/1q7P3rQt5di3NT-V2iR0xGsEKHqw7LXm6?usp=sharing, 2025.