ECE661 Final Project - Adversarial Patch Attacks

Kashaf Ali, Elisa Chen, Ya-Yun (Kate) Huang

Abstract

This study delves into the realm of adversarial attacks with specific focus on training adversarial patches using the CIFAR-10 dataset. The primary objective is to understand the impact of different versions of these patches on various Deep Neural Network architectures. Specifically, our investigation aims to uncover three pivotal facets: first, the susceptibility of neural networks to adversarial patch attacks; second, the sensitivity of models to varying patch sizes; and third, the transferability of these adversarial patches across a spectrum of neural network models. We found larger patches relative to input size exhibited higher success rates in deceiving neural networks, with notable transferability observed in untargeted attacks across DenseNet-121 and VGG-16. However, targeted attacks showed relatively limited transferability, highlighting challenges in carrying over adversarial patterns to different architectures.

1 Introduction

Recent developments in machine learning and deep learning have paved the way for advancements in various applications including image recognition, autonomous vehicles and intelligent virtual assistants. However, as we embrace the transformative power of this technology, we also become increasingly susceptible to potential adversarial attacks. Adversarial attacks are deliberate and malicious attempts to deceive or manipulate machine learning and deep learning models by exploiting their vulnerabilities. While these attacks can be of various kinds, most of them involve making small and imperceptible changes to input data that can lead the model to make incorrect predictions. The consequences of these successful adversarial attacks can be profound, ranging from security breaches in image recognition systems [1] to compromised safety in autonomous vehicles [2].

Adversarial patches are a sophisticated form of adversarial attacks that involve placing a carefully crafted patch on an image to fool the model in making incorrect classifications. In this case, the pixel values within the patch area are also trained strategically to maximize the likelihood of misclassification under various randomly applied transformations such as rotations, changes in lighting or scale of the patch. The patch is trained to be robust against these different transformations and this makes the adversarial attack more potent because the model will consistently misclassify an image containing the patch regardless of the changes made to it. As the effectiveness of an adversarial patch increases, so does the level of concern it poses for applications reliant on image-based machine learning models.

In our project, we utilized the CIFAR-10 dataset to train adversarial patches, aiming to examine both the impact of various patch types and the degree to which these trained patches could transfer across different Deep Neural Network architectures. More explicitly, we seek to understand 1) how susceptible are neural networks to adversarial patch attacks and 2) how sensitive is the model to the size of the patch, and 3) the transferability of the adversarial patch to other models. Understanding the impact and the limitations of adversarial patch attacks is crucial for ensuring the robustness and safety of machine learning systems.

2 Literature Review

Common adversarial attacks modify each pixel by only a small amount invisible to the human eye to mislead the neural network to output a different target class. These modifications can be found with a number of optimization strategies including L-BFGS, Fast Gradient Sign Method, DeepFool, Projected Gradient Descent, and Logit-space Projected Gradient Ascent [3]. Other attack methods seek to modify only a small number of pixels in the image (Jacobian-based saliency map), or a small patch at a fixed location of the image. While prior methods have mostly focused on attacking with

either small or imperceptible changes to the input, our work focuses on generating adversarial patches that can be placed on an image to cause the classifier to output a targeted class.

Brown et al. pioneered the concept of adversarial patches in image classification, aiming to create universal, robust and localized patches that could be applied to images through masking rather than pixel additions. Their research demonstrated that despite their relatively small size, these patches could significantly influence classifiers, causing them to disregard other elements in the image to predict the targeted class [3]. Subsequently, Lee and Kolter's research further expanded on this concept by showcasing that a strategically designed patch could be positioned anywhere within an image to effectively obscure not just nearby objects but also those situated at a distance from the patch itself [5]. Moreover, Bai et al.'s innovative approach, which accounted for the spatial placement of patches by leveraging the perceptual sensitivity of victim models during adversarial patch training, also made a valuable contribution to the existing literature [6].

Furthermore, various other studies have also delved into the practical applications of adversarial patches. Thys, Ranst and Goedemé's work presents an approach to generate adversarial patches to targets with a lot of intra-class variety that can be used to successfully yet maliciously hide people from surveillance systems [7]. Sharif et al. instead employ a more creative approach where they use eyeglasses patches to evade recognition or mimic another individual within a facial biometric system [8]. In line with the existing literature on adversarial patches, our research investigates the implications of adversarial patch attacks and explores the potential risks they pose on neural networks.

3 Methodology

In this project, we trained adversarial patches with CIFAR-10 dataset, explored the outcomes of different sizes and shapes of patches and the transferability of the trained patch on different Deep Neural Network architectures. Overall, we aim to evaluate if we can train an attack patch that is not restricted to imperceptible changes that can be applied universally on different models. Our approach can be broken into the following parts:

- Implement the Adversarial Patch generation process for a CIFAR-10 classifier based on the Adversarial Patch paper by Brown et al. The adversarial patches are trained on pre-trained ResNet-18 models and would be served as white-box attacks. There will be two types of attack: untargeted and targeted attack.
- Explore the outcomes of different sizes of adversarial patches on CIFAR-10.
- Utilizing the patches trained with ResNet-18, test the transferability of different-size patches on VGG-16 and DenseNet. This step is considered as a black-box attack.

The adversarial patch training process starts by randomly initializing our patch with a square shape with a size that is smaller than the size of CIFAR-10 images (32 x 32 pixels). We also define a patch application operator A(patch, image, location,rotation) to apply a patch with rotation to a different location of each trained image during the training process. The untargeted and targeted attack takes a different approach to update the patch. For the untargeted attack, the patch is updated through stochastic gradient ascent in order to maximize the cross entropy loss. Whereas, for the targeted attack, each image's label is set to the target class y, and the loss is calculated with respect to a target label y. The patch is updated through stochastic gradient descent to minimize the loss as we would like the loss for predicting the class as the target class to be minimized. During the training for both targeted and untargeted approach, only the patch is being updated and the weights of the model remain the same. The objective function of both approach takes the following form:

- Untargeted attack:

$$\hat{p} = \arg\max_{p} E_{x\sim X, r\sim R, l\sim L}[CE(\hat{y}|A(p, x, l, t), y]$$

  where p refers to the patch, CE refers to Cross Entropy Loss, and A is the patch application operator.

- Targeted attack:

$$\hat{p} = \arg\min_{p} E_{x\sim X, r\sim R, l\sim L}[CE(\hat{y}|A(p,x,l,t),y]$$

where all images are assigned to the target class y.

Then, we evaluate and compare the effect of various patch sizes: 3x3, 5x5, 7x7, 16x16, to understand how patch sizes impact the effectiveness of the adversarial patches. To quantify the attack effectiveness, we define our attack success rate (ASR) as follows:

- Untargeted attack:

$$\frac{\text{Number of incorrectly predicted images}}{\text{Number of all images}}$$

- Targeted attack:

$$\frac{\text{Number of Images that are not in the target class, but are predicted as the target class}}{\text{Number of Images that are not in the target class}}$$

We also compared the untargeted ASR to targeted ASR to test which type of patch offers us the best model. To test the transferability of our adversarial patches, we applied different trained targeted and untargeted patches of different sizes to a DenseNet model and VGG-16 model to evaluate the effectiveness of the adversarial patches as a black-box attack.

4 Experiment Results

We used ResNet-20 to train our untargeted patches and obtained results shown in Table 1. Due to the modest dimensions of our training images and patches, discernible patterns were not readily apparent in the patches. However, a consistent color palette emerged across all patches, with shades of purple, pink, green, and blue appearing in all the patches. To establish a baseline for our ASR, we fine-tuned all models using the CIFAR-10 dataset and calculated the baseline error rate, representing the validation error when no patches were applied (see red line in Figure 1). Notably, all patches demonstrated a substantial improvement over the baseline error rate, underscoring their efficacy in compromising the neural network with a certain degree of success.

Model	Patch 3x3	Patch 5x5	Patch 7x7	Patch 16x16
Whitebox-ResNet-20	0.342	0.563	0.791	0.904
Blackbox-DenseNet	0.398	0.579	0.394	0.767
Blackbox-VGG16	0.305	0.493	0.521	0.814

Table 1: Untargeted Validation ASR

Furthermore, our experiment results revealed a positive correlation between patch size and ASR. This aligns with our intuitive understanding of larger patches covering more extensive areas of the image, making it easier to deceive the neural network by incorrectly predicting a class. Interestingly, in all instances, the validation ASR equaled or surpassed the training ASR. This phenomenon may be attributed to the model's robustness to perturbations during training, leading to more accurate predictions. Perhaps, during validation, the model encounters a more diverse set of examples, potentially making it more susceptible to the adversarial patches.

Using the patches trained on ResNet-20, we also evaluated the ASR on alternative architectures, namely DenseNet-121 and VGG-16, to measure the generalization capability of the adversarial patches. The findings, presented in figure 2 for DenseNet and VGG, respectively, indicate a notable degree of transferability of the patches to these distinct models. With the exception of 7 x 7 patch, the validation ASRs for DenseNet and VGG were found to be comparable to those achieved in ResNet suggesting successful transferability of patches to other architectures. One plausible explanation for the

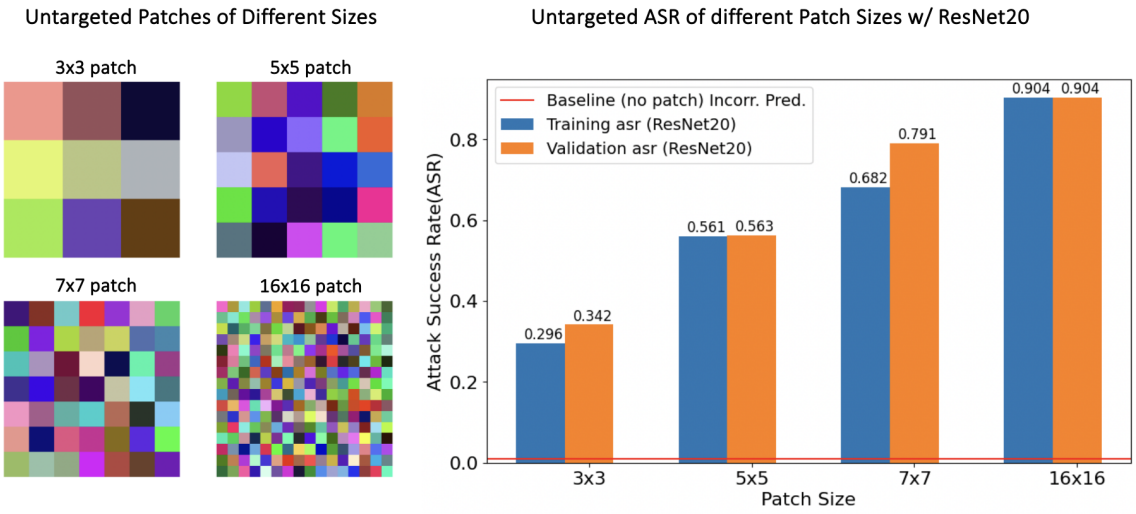

Figure 1: Untargeted Validation ASR

comparatively diminished ASR of the 7 x 7 patch size in both DenseNet and VGG could be attributed to potential interference with specific network structures. The 7 x 7 patch size might overlap with critical components of the network, such as convolutional filters or pooling operations, in a way that disrupts the model's internal representations. This interference could result in less effective adversarial attacks.

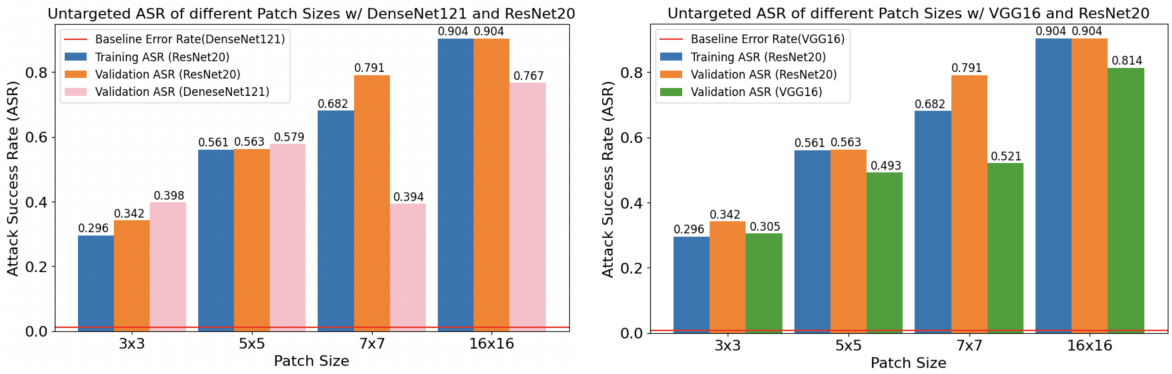

Figure 2: Untargeted Validation ASR with DenseNet121 and VGG16

Model	Patch 3x3	Patch 5x5	Patch 7x7	Patch 16x16
Whitebox-ResNet-20	0.149	0.489	0.764	0.958
Blackbox-DenseNet	0.164	0.292	0.361	0.311
Blackbox-VGG16	0.070	0.128	0.134	0.057

Table 2: Targeted Validation ASR for Target Class: Bird

Model	Patch 3x3	Patch 5x5	Patch 7x7	Patch 16x16
Whitebox-ResNet-20	0.008	0.015	0.124	0.975
Blackbox-DenseNet	0.019	0.022	0.074	0.051
Blackbox-VGG16	0.040	0.033	0.064	0.063

Table 3: Targeted Validation ASR for Target Class: Horse

Similar to untargeted ASR, we used ResNet-20 to train our targeted patches. Given the substantial number of classes, encompassing a total of 10 categories, we opt for a more focused analysis by only using results from target class Bird (representing a favorable outcome) and target class Horse (representing an unfavorable outcome) to illustrate our findings. A comprehensive compilation of all results is provided in the Supplementary Materials section for curious readers.

Just as we observed with untargeted attacks, a positive correlation is generally discerned between patch size and ASR, although the strength of this correlation is not as pronounced as encountered in untargeted attack scenarios. Notably, a relatively linear increase in ASR can be noticed in table 2 when we increase the patch size for target class 'Bird'. However, in the case of the target class 'Horse', the validation ASR remains consistently low for patch sizes 3x3, 5x5, and 7x7 as illustrated in table 3. This suggests potential challenges in the model's ability to effectively capture meaningful patterns associated with the 'Horse' class. Perhaps, the representation of certain classes like horse can possess more complex patterns or variations that are hard to separate from the rest of the classes. As an example, distinguishing classes such as 'dog', 'cat', and 'deer' become more intricate in comparison to horses as they share similar anatomical features in contrast to birds. When comparing the patch patterns of the 'Bird' class and the 'Horse' class in figure 3, we notice that the 'Bird' class contains a lot of shades of blue whereas the 'Horse' class contains a mix of yellow, blue and red shades. Perhaps, these might be less effective patterns to adversarially attack the neural network.

When evaluating the ASR on DenseNet and VGG for transferability properties, we observed significant disparities in the generalization capabilities of targeted patches as opposed to untargeted patches. For target class 'Bird', we obtain ASR values surpassing the baseline for all patch sizes, albeit with diminishing gains as the patch size increases. Interestingly, the 'Bird' patch exhibits a higher ASR on DenseNet compared to VGG, potentially attributable to the more analogous architectural structures shared between DenseNet and ResNet. Conversely, the 'Horse' patch demonstrates even more inferior transferability, with validation ASR consistently below 10% for both models across varying patch sizes, indicative of suboptimal transferability properties. Comparatively, in relation to untargeted attacks, targeted attacks yield lower ASR and transferability capabilities.

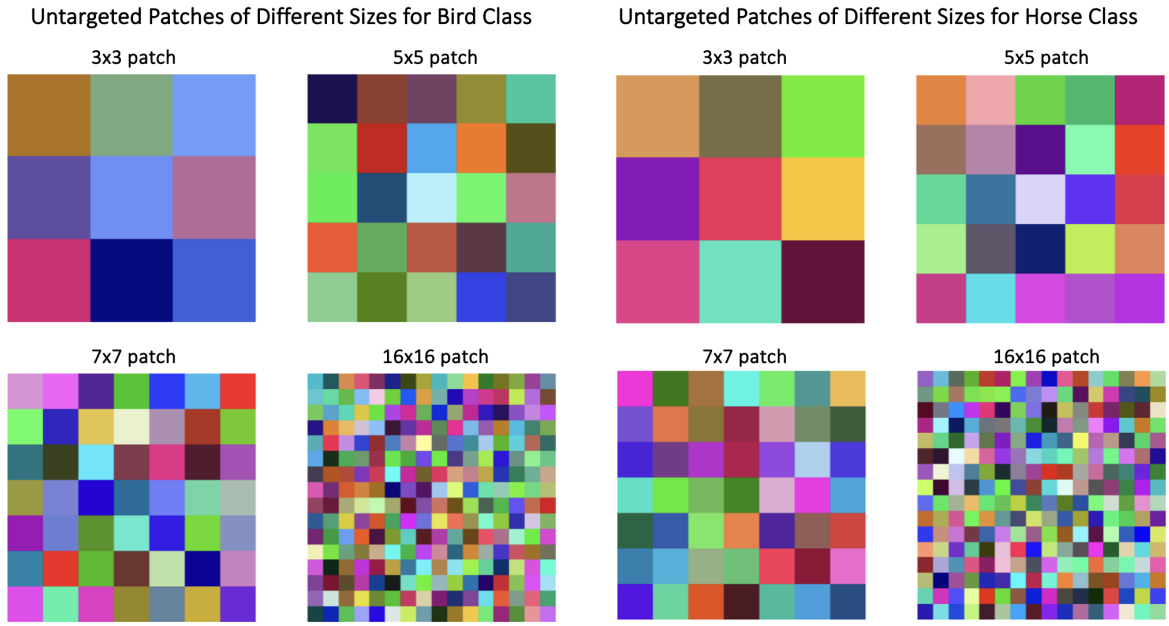

Figure 3: Untargeted Patches of Different Sizes for Bird and Horse Classes

5 Conclusion

In this study, we utilized the CIFAR-10 dataset to train adversarial patches, aiming to examine both the impact of various patch types and the degree to which these trained patches could transfer across

different Deep Neural Network architectures. Our experimentations with untargeted and targeted attacks were able to surpass baseline error rates highlighting their efficacy. Generally, we observed a positive correlation between patch size and attack success rate demonstrating that larger patches relative to the size of the input image have a better ability to deceive the neural network. Evaluations across DenseNet-121 and VGG-16 showcased notable transferability of adversarial patches for untargeted attacks. However, we observed relatively inferior transferability properties with targeted attacks using 'Bird' patch and 'Horse' patch as examples emphasizing nuanced challenges with targeted attacks in carrying over adversarial patterns to alternative architectures.

6 References

[1] Hwang, Ren-Hung, et al. "Adversarial Patch Attacks on Deep-Learning-Based Face Recognition Systems Using Generative Adversarial Networks." Sensors 23.2 (2023): 853.

[2] Yi, Ru, and Jicheng Chen. "Kalman Filter-Based Adversarial Patch Attack Defense for Autonomous Driving Multi-Target Tracking." 2023 IEEE International Conference on Industrial Technology (ICIT). IEEE, 2023.

[3] Brown, Tom B., et al. "Adversarial patch." arXiv preprint arXiv:1712.09665 (2017).

[4] Athalye, Anish, et al. "Synthesizing robust adversarial examples." International conference on machine learning. PMLR, 2018.

[5] Lee, Mark, and J. Zico Kolter. "On Physical Adversarial Patches for Object Detection." arXiv:1906.11897v1 [cs.CV], 20 June 2019.

[6] Bai, Tao, Jinqi Luo, Jun Zhao. "Inconspicuous Adversarial Patches for Fooling Image Recognition Systems on Mobile Devices." arXiv:2106.15202v2 [cs.CV], 21 Nov 2021.

[7] Thys, Simen, Wiebe Van Ranst, and Toon Goedemé. "Fooling Automated Surveillance Cameras: Adversarial Patches to Attack Person Detection." 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019, pp. 49-55. DOI: 10.1109/CVPRW.2019.00012.

[8] Sharif, Mahmood, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. "Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition." Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016.

7 Supplementary Material

We repeated our targeted attack experiments for 3x3, 5x5, 7x7, and 16x16 patches for 10 different classes and 3 different models.

Please refer to our results repository here for a comprehensive list of all the experimentation outputs. Below is a mapping between the class names and indices of CIFAR-10 images:

Index	Class Name
0	Plane
1	Car
2	Bird
3	Cat
4	Deer
5	Dog
6	Frog
7	Horse
8	Ship
9	Truck