Evaluating Adversarial Patch Attack and Patch Attack Transferability on CIFAR-10 Classifiers

William Chown
Electrical & Computer Engineering
Duke University
william.chown@duke.edu

Malcolm Smith Fraser
Social Science Research Institute
Duke University
malcolm.smith.fraser@duke.edu

Wei Wu
Electrical & Computer Engineering
Duke University
ww148@duke.edu

Abstract

Many deep learning systems are vulnerable to adversarial attacks. In the paper we investigate the potential of an adversarial patch attack on images from the CIFAR-10 dataset. We find that a patch attack is successful in two modes of use. First, it can fool a model into predicting a specific targeted class. And second, it can simply fool the model into predicting an erroneous class. In addition to the success of these modes, we demonstrate that a patch attack can be transferred to different models while retaining much of its effectiveness.

1 Background

Adversarial attacks have been demonstrated to be widely successful against deep learning systems. Examples of adversarial attacks include Fast Gradient Sign Method [1] and Projected Gradient Descent [2]. While effective, it is notable that these two examples (and others) make small and potentially imperceptible changes that depend on the images they are modifying. As such, they run a risk of being "scene-dependent" - that is, scene changes such as changes in background or changes in lighting can impact performance.

In contrast, this paper explores the adversarial patch attack, as previously examined in Brown et al. [3] A key component to the patch attack is that by not restricting the attack to be (humanly) imperceptible, a scene-independent patch can be generated that is effective across various lightings and backgrounds. Further, it is transferable; that is, a patch trained on one type of classifier can be effective when applied to different types of classifiers. Notably, a patch attack has a great deal of potential as a widespread attack. Once patches are created, they could, for example, be spread across the Internet and used by many other attackers.

Thus, all an attacker needs to construct this type of attack is a sample dataset and a target class. However, the attack is not perfect. In addition to being far more perceptible than attacks such as FGSM and PGD, an efficient method of detecting an adversarial patch attack was developed earlier this year that significantly improves provable robustness against a patch attack as well as clean performance [4].

ECE 661 (2021 Fall) final report.

2 Methods

The CIFAR-10 dataset is one of the most popular datasets for machine learning research. It consists of 60,000 32x32 color images, evenly distributed across ten distinct classes: airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. Due to its widespread use, it serves as a good benchmark for the success and transferability of an adversarial patch attack. Because our models were pre-trained PyTorch models, we first transformed the dataset images by normalizing and transforming them to tensors.

The target model was a pre-trained ResNet34 [5]. A pre-trained ResNet18 [5] model was used to evaluate the patch attack's transferability on a similar model architecture, and a pre-trained Vgg16 [5] was used to test the transferability for a more different architecture.

Patch pixel size was 2x2, 4x4, 8x8, or 16x16. Patch values were normalized to fall within the range of values in the original CIFAR-10 data, and the patch location was generated at random. Patches were trained as a new parameter in the pre-trained model over five epochs, and initialized with zeros. The training procedure was as follows:

Algorithm 1 General Workflow for Training Adversarial Patch Attack

```
Initialize Patch with SGD optimizer, target class
for each training epoch do
    Place patch on image
    Generate predictions
    Compute cross entropy loss of model w.r.t. patch target class
    Back-propagation
    Update optimizer
end for
```

The model was evaluated with four different metrics. Top1 and Top5 accuracies measure how well the patch fooled the model into predicting a target class as the top or within the top 5 predictions. Similarly, Top1 and Top5 error measure how frequently the model failed to predict the true class as the top or within the top 5 predictions.

3 Experiment results

The detailed numerical results of the adversarial patch attacks are presented in Appendix C in tables 1, 2, 3, 4, and 5.

3.1 Targeted Attacks

For targeted attacks, we evaluated the Top 1 and Top 5 misled accuracies for patched images where a higher accuracy of the misled target class means a more successful patch attack. An example of a targeted attack on a "ship" image is shown in 1. Results of Top-1 and Top-5 misled accuracies are plotted in 2.

3.1.1 Adversarial Patch Size

In this case, we explored how the adversarial patch size can impact the performances of targeted attacks.

As we can see here, under the same model, as the patch size increases, the targeted misled accuracy also has an increasing trend with the only exception for Vgg-16 Top 5 accuracy. Essentially, the result indicates that a larger patch size generally results in a higher misled accuracy, meaning that the performance of a targeted attack is also getting better.

3.1.2 Attack Transferability

In this case, we explored how well the targeted attack performance was transferred from the baseline ResNet-34 model to other CNN structures. Here we selected ResNet-18 and Vgg-16 models as the comparisons where we tested the model using the same patch trained with ResNet-34. Although both

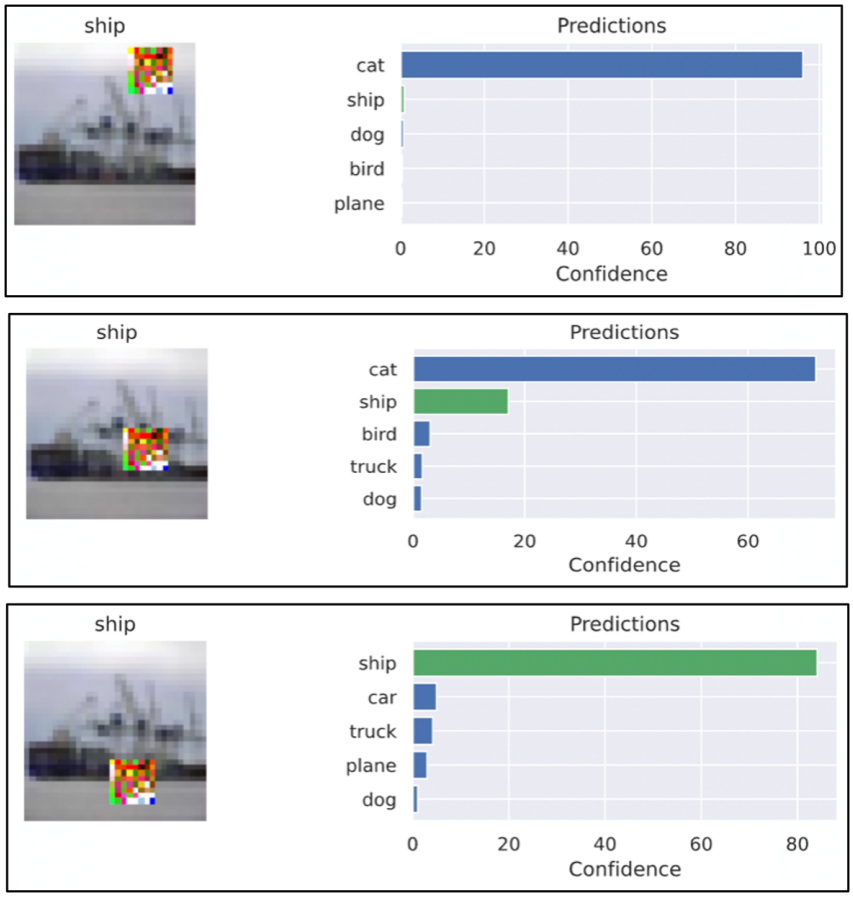

Figure 1: Comparison of Top 5 predictions using "Cat" patch to attack ResNet-34, ResNet-18, and Vgg-16 on a "Ship" image

models had different structures compared to ResNet-34, since ResNet-18 and ResNet-34 are still within the same CNN family, there is more similarity within these two models than within Vgg-16 and ResNet-34.

As we can see here, with the same patch size, in general ResNet-18 has higher Top 1 and Top 5 accuracies than those of Vgg-16. A higher misled accuracy here means that the patch attack has a better targeted attack performance after being transferred to another model, meaning that the attack performance is better transferred to ResNet-18 than Vgg-16.

Therefore, the more similarity between the baseline model and the transferred model, the more transferable the targeted attack could achieve.

3.2 Untargeted Attacks

For untargeted attacks, we are evaluating the Top 1 and Top 5 errors generated by the model testing on patched images where a higher error stands for a more successful patch attack. Results of Top-1 and Top-5 misled accuracies are plotted in 3.

3.2.1 Adversarial Patch Size

In this case, we explored how the adversarial patch size can impact the performances of targeted attacks.

As we can see here, under the same model, as the patch size increases, the model error increases as well. Essentially, the result indicates that a larger patch size generally results in a larger error by the

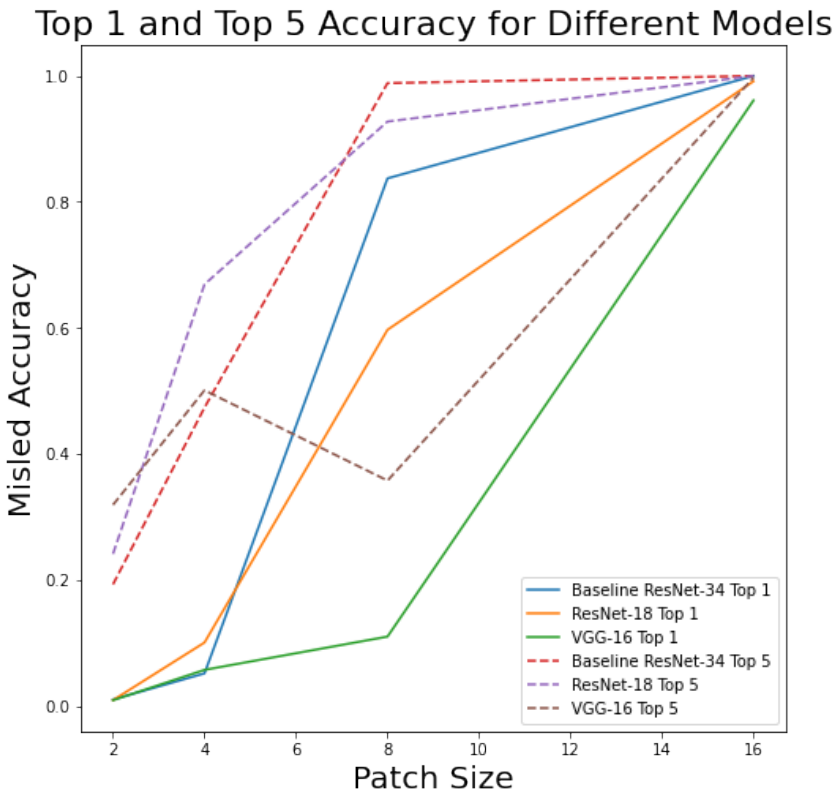

Figure 2: Top 1 and Top 5 Misled accuracies for ResNet-34, ResNet-18, and Vgg-16 vs. Patch Size

model, meaning that the performance of an untargeted attack also scales with patch size. This makes sense: the larger, the patch, the more significant the changes to the image.

3.2.2 Attack Transferability

Similarly to the targeted attack, we again explored how well the attack performance was transferred from the baseline ResNet-34 model to other CNN structures.

As we can see here, with the same patch size, in general ResNet-18 has higher Top 1 and Top 5 errors than those of Vgg-16. A higher error generated by the transferred model means that the patch attack has a better untargeted attack performance after being transferred, meaning that the attack performance is better transferred to ResNet-18 than Vgg-16.

Therefore, the more similarity between the baseline model and the transferred model, the more transferable the untargeted attack could achieve. Notably, the attack transferred to the Vgg-16 model almost as effectively as to the ResNet-18 model.

3.3 Targeted Attacks vs. Untargeted Attacks

During the experiments, we also noticed that the effects of patch sizes and attack transferability differs between targeted attacks and untargeted attacks.

In particular, we have observed that in 3, the drops for ResNet-18 and Vgg-16 models for both Top 1 and Top 5 errors are not very significant, whereas the Top-1 and Top-5 misled accuracies for Vgg-16 models are much lower than the accuracies for ResNet-18 and ResNet-34 until patch size 16×16. Also, the Top 5 accuracy for Vgg-16 model at patch size 8×8 is actually lower than that of 4×4, showing great variability from the predicted increasing curve. Both figures reflect that in general, untargeted attacks are better suited for transferred models compared to targeted patch attacks

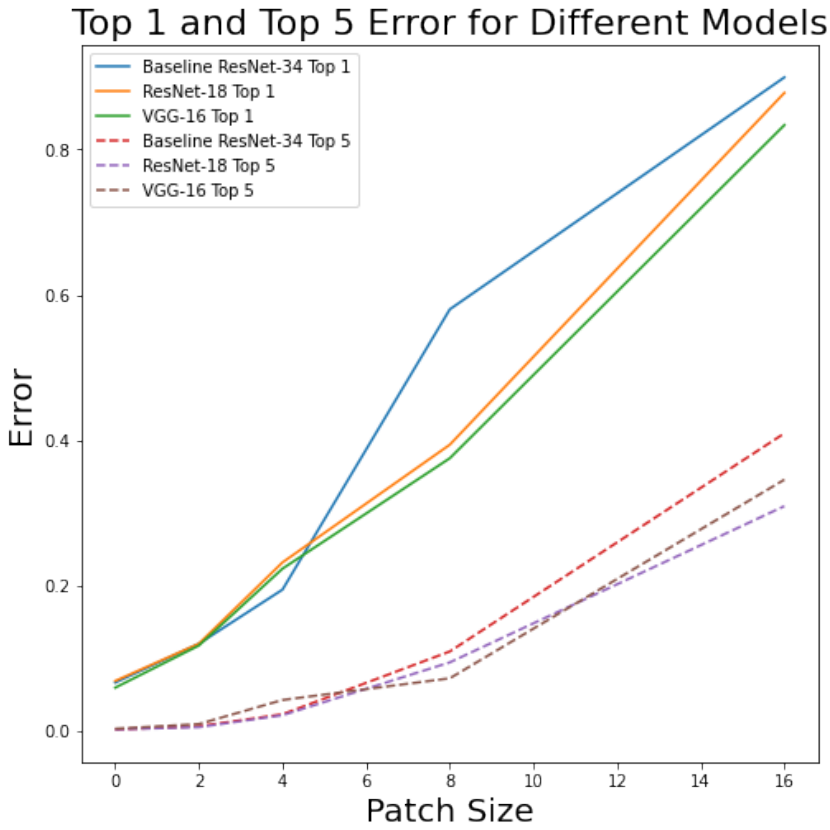

Figure 3: Top 1 and Top 5 Errors for ResNet-34, ResNet-18, and Vgg-16 vs. Patch Size

because untargeted attacks on transferred models typically have lower drops and better consistency on performance.

4 Future Works

There are several areas where we feel further research on patch attacks is merited. First, with patch placement. We randomly placed each patch, but we expect that patch placement would play an important role in the success of a patch attack on, say, facial recognition software - if a patch is placed on someone's arm, it will not fool a facial recognition scanner. Second, with alternative methods of generating a patch for a targeted attack. It would be interesting to see how a downsampled image of the target class would compare to our trained patch implementation. Third, we feel there is still a need for more research on defending against patch attacks. One idea we have is to train a model using CutMix regularization, where the "mix" part would come from a target example.

5 Conclusions

Adversarial patch attacks do not require access to a model, or even exact knowledge of the model's type. As we demonstrated, patch attacks can have a significant impact on classification error not only for the model the patch is trained on, but also for distinctly different models as well. Targeted attacks were most effective on the baseline ResNet34 model for which the patches were trained, but also proved effective when transfered to the ResNet18 model. However, the effect of a targeted transfer attack on Vgg16 was relatively small. In contrast, while untargeted attacks were again most effective on the baseline ResNet34 model, untargeted transfer attacks proved almost as effective on both the ResNet18 and Vgg16 models.

Notably, pictures of simple adversarial patches can fool state-of-the-art facial recognition system, which exposes the underlying vulnerability of security systems using deep learning techniques [6]. Due to their transferability effectiveness (especially in untargeted attacks), ease of use, and relatively straightforward implementation, adversarial patch attacks are an important type of attack to train deep learning systems against, and more research should be done in this area.

The use of deep learning systems is increasingly widespread in the world, in diverse fields from natural language processing to autonomous vehicles to fraud detection to healthcare. As more of the infrastructure we take for granted becomes reliant on deep learning, the danger of an adversarial attack that impacts people's lives becomes more serious. Real-world examples of adversarial attacks include text-based poisoning with the chatterbot Tay, an audio-based evasion attack on Mozilla DeepSpeech, and image-based evasion attacks that render objects such as stop signs "invisible" to a classifier [7].

In light of the proliferation of adversarial attacks, it is important to understand how they work and what types of models they can attack, especially regarding their transferability to similar and more different models. A weakness of adversarial attacks that rely upon access to the model they are attacking is that in the real world such access may not be feasible or even possible.

References

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

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

[3] Tom B. Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv:1712.09665*, 2018.

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

[5] Huy Phan. huyvnphan/pytorch_cifar10, January 2021.

[6] Mikhail Pautov, Grigorii Melnikov, Edgar Kaziakhmedov, Klim Kireev, and Aleksandr Petiushko. On adversarial patches: Real-world attack on arcface-100 face recognition system. *2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, Oct 2019.

[7] H. Lin and B. Biggio. Adversarial machine learning: Attacks from laboratories to the real world. *Computer*, 54(05):56–60, may 2021.

Note that the Reference section does not count towards the 6 pages of content that are allowed.

Table 1: Top1 / Top5 baseline error

Model	Top1	Top5
ResNet34	6.67%	0.25%
ResNet18	6.93%	0.26%
Vgg16	6.00%	0.36%

Table 2: Top1 error vs. patch size

Model	Transfer	2x2 Patch	4x4 Patch	8x8 Patch	16x16 Patch
ResNet34	No	12.04%	12.04%	58.01%	89.90%
ResNet18	Yes	12.05%	23.22%	39.40%	87.77%
Vgg16	Yes	11.79%	22.36%	37.54%	83.34%

Table 3: Top5 error vs. patch size

Model	Transfer	2x2 Patch	4x4 Patch	8x8 Patch	16x16 Patch
ResNet34	No	0.70%	2.35%	10.98%	40.17%
ResNet18	Yes	0.57%	2.21%	9.48%	30.96%
Vgg16	Yes	1.00%	4.31%	7.29%	34.60%

A Timeline and task allocation

Please provide your working timeline of the project in this section. Please also specify how the task is allocated among the two members.

November:

- November:
  - Find and evaluate pretrained CIFAR10 model (ResNet34, ResNet18, Vgg16) baselines: Malcolm
  - Implement patch generation from 4 classes in CIFAR10 (car,plane, cat, bird): Malcolm
  - Implement patch attacks on the ResNet34 model, ResNet18, and Vgg16: Malcolm
- December:
  - Finalize experiment settings and refine experiment results: Malcolm and Wei
  - Generate results figures: Malcolm and Wei
  - Draft Report: William
  - Draft Poster: Wei, William
  - Build poster: William, Wei, Malcolm
  - Complete report: William, Wei, Malcolm
  - Evaluate the effect of a transfer patch attack on the ResNet18 and Vgg16 models: Malcolm

B Additional experiment results

Table 4: Top1 accuracy vs. patch size

Model	Transfer	2x2 Patch	4x4 Patch	8x8 Patch	16x16 Patch
ResNet34	No	0.99%	5.14%	83.73%	99.99%
ResNet18	Yes	0.89%	10.04%	59.69%	99.15%
Vgg16	Yes	0.88%	5.70%	10.98%	96.11%

Table 5: Top5 accuracy vs. patch size

Model	Transfer	2x2 Patch	4x4 Patch	8x8 Patch	16x16 Patch
ResNet34	No	19.27%	47.34%	98.85%	100%
ResNet18	Yes	24.11%	66.90%	92.72%	100%
Vgg16	Yes	31.89%	50.08%	35.75%	99.72%