Jailbreaking Deep Models: Adversarial Attacks on ResNet-34 and Transferability Analysis

Anshi Shah, Kanishk Aggarwal, Shashank Dugad

New York University, Tandon School of Engineering
Project Codebase: github.com/anshi312/Jailbreaking-Deep-Models-ResNet-Adversarial-Attacks

Abstract

This project investigates the vulnerability of state-of-the-art image classification models to adversarial attacks. A pretrained ResNet-34 is targeted on ImageNet-1K and multiple adversarial strategies are explored under pixel-wise (L_∞) and patch-wise (L_0) constraints. The attacks significantly degrade the model's performance, and their transferability is further assessed using DenseNet-121, observing a substantial drop in accuracy even without model-specific tuning. The results demonstrate the susceptibility of production-grade models to subtle perturbations and underline the importance of robust defenses.

Introduction

Adversarial examples are imperceptibly perturbed inputs that cause deep neural networks to make incorrect predictions. In this project, a pretrained ResNet-34 is systematically attacked on ImageNet-1K using multiple techniques: Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), targeted PGD, and patch-based attacks. The effectiveness of each attack is evaluated in terms of top-1 and top-5 accuracy, and their transferability is tested on DenseNet-121.

Related Work

Adversarial attacks were first introduced by Szegedy et al. (2013), who demonstrated that deep neural networks are vulnerable to imperceptible perturbations, leading to incorrect predictions. This discovery initiated a significant line of research focused on developing both attack algorithms and robust defense mechanisms.

Goodfellow et al. (2015) proposed the Fast Gradient Sign Method (FGSM), a one-step L_∞-norm based attack that perturbs each pixel in the direction of the gradient sign. While simple and fast, FGSM is often not sufficient to fully degrade model performance under small perturbation budgets. To address this, Madry et al. (2018) introduced the Projected Gradient Descent (PGD) attack — a multi-step iterative version of FGSM that remains the benchmark for strong white-box adversarial attacks.

Kurakin et al. (2016) extended adversarial evaluation to the physical world, highlighting that adversarial examples are not only theoretical but practical threats to deployed systems. Subsequent work by Brown et al. (2017) introduced the concept of patch-based attacks, where only a small part of the image is perturbed. These attacks are particularly concerning due to their high success rates and real-world applicability (e.g., adversarial stickers).

Transferability — the ability of adversarial examples to fool models other than the one they were crafted on — was analyzed by Liu et al. (2017). Their findings confirmed that black-box attacks are viable due to shared vulnerabilities across model families. This insight directly motivates the importance of evaluating attacks on multiple architectures, which we do by testing our attacks against DenseNet-121 in addition to ResNet-34.

Our project builds on these core ideas by integrating FGSM, PGD, targeted attacks, and localized patch attacks, evaluating both effectiveness and generalizability across models.

Methodology

Dataset and Preprocessing

The experiments were conducted on a subset of 500 images from 100 different classes of the ImageNet-1K dataset. Each image was resized and normalized using the standard ImageNet normalization parameters: mean = [0.485, 0.456, 0.406] and standard deviation = [0.229, 0.224, 0.225]. The dataset was formatted to comply with PyTorch's `ImageFolder` interface, and label mappings were retrieved from a provided `labels_list.json` file.

ResNet-34 Model

ResNet-34 was selected as the primary target model for adversarial evaluation. It is a 34-layer deep convolutional neural network that utilizes residual blocks to mitigate the vanishing gradient problem. These blocks enable the network to learn identity mappings, improving training stability and performance. The pretrained model was obtained from `torchvision.models.resnet34` with weights `IMAGENET1K_V1`.

Step 1: Baseline Evaluation

The clean test dataset was evaluated on ResNet-34 to establish baseline performance. Predictions were extracted using the model's top-1 and top-5 softmax scores. Accuracy was computed by comparing predicted indices with ground truth labels. This served as the control metric against which adversarial accuracy degradation was measured.

Step 2: FGSM Attack

The Fast Gradient Sign Method (FGSM) was implemented with $\epsilon = 0.02$ under the L_∞ norm constraint. A single gradient step was computed using the cross-entropy loss, and the input was perturbed in the direction of the sign of the gradient. The perturbation was added to the original image and clipped to ensure pixel values remained in valid ranges. The attack targeted the full image. Resulting adversarial images were saved as "Adversarial Test Set 1."

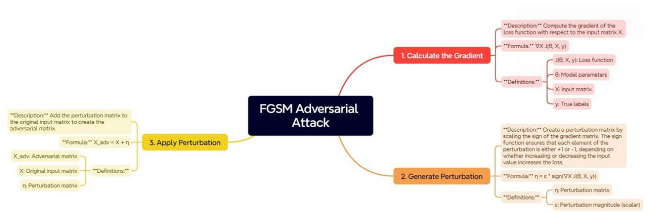
FGSM Adversarial Attack
1. Calculate the Gradient
Description: Compute the gradient of the loss function with respect to the input matrix X.
Formula: ∇X J(θ, X, y)
Definitions:
J(θ, X, y): Loss function
θ: Model parameters
X: Input matrix
y: True labels
2. Generate Perturbation
Description: Create a perturbation matrix by scaling the sign of the gradient matrix. The sign function ensures that each element of the perturbation is either +1 or -1, depending on whether increasing or decreasing the input value increases the loss.
Formula: η = ε * sign(∇X J(θ, X, y))
Definitions:
η: Perturbation matrix
ε: Perturbation magnitude (scalar)
3. Apply Perturbation
Description: Add the perturbation matrix to the original input matrix to create the adversarial matrix.
Formula: X_adv = X + η
Definitions:
X_adv: Adversarial matrix
X: Original input matrix
η: Perturbation matrix

Figure 1: FGSM attack process

As shown in Figure 1, the input is perturbed by computing the loss gradient w.r.t. the input, generating a signed perturbation scaled by ϵ, and applying it to form the adversarial image.

Step 3: Targeted PGD Attack

Projected Gradient Descent (PGD) was applied as an iterative, stronger variant of FGSM. A targeted attack was implemented by selecting a fixed incorrect class label (e.g., class 0) and minimizing the model's confidence in the true label over 40 steps of gradient descent. Each step used a learning rate of $\alpha = 0.0025$ and projected the perturbation back into the $\epsilon = 0.02$ L_∞ ball around the original input. Perturbations were applied across the entire image. The perturbed set was saved as "Adversarial Test Set 2."

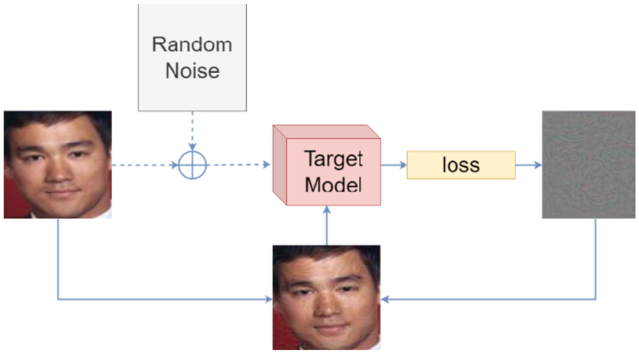

Figure 2: PGD attack loop

As shown in Figure 2, starting with random noise, the input is iteratively perturbed and passed through the target model. Gradients are computed from the loss, and the perturbation is projected to stay within the ℓ_∞ constraint.

Step 4: Patch-based Attack

To simulate localized attacks, a targeted PGD was restricted to a 32×32 patch randomly placed within the image. The perturbation budget was increased to $\epsilon = 0.5$, and the attack was conducted for 40 steps with $\alpha = 0.05$. All pixels outside the patch remained untouched. This form of attack tests the robustness of models against constrained, spatially localized manipulations. Resulting examples were saved as "Adversarial Test Set 3."

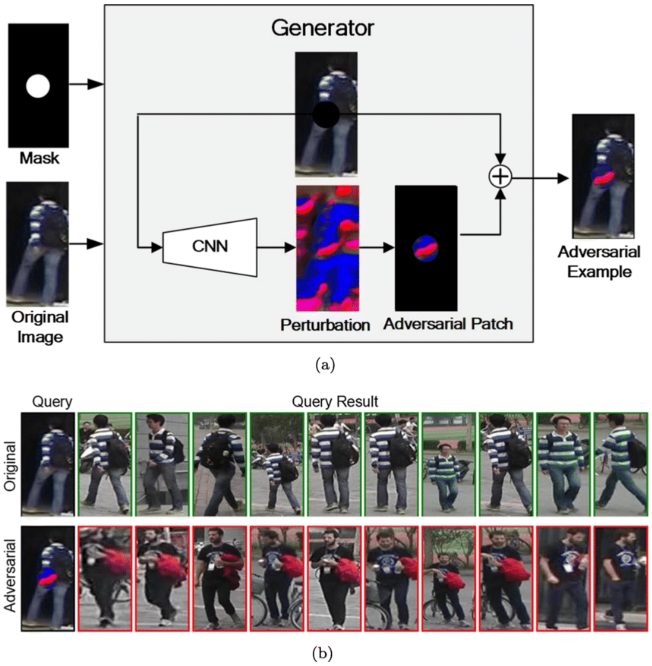

Figure 3: Illustration of a patch-based adversarial attack. (a) A mask and a learned perturbation are applied to a specific region of the image to generate an adversarial patch. (b) The patch causes significant errors in downstream image retrieval, despite only modifying a localized area.

As illustrated in Figure 3, the adversarial patch modifies only a small spatial region yet results in misclassification.

Step 5: Transferability to DenseNet-121

To evaluate transferability, all datasets (original and adversarial) were tested on a second pretrained model — DenseNet-121. Unlike ResNet, DenseNet utilizes dense connections between layers, where each layer receives feature maps from all preceding layers. The pretrained model was loaded via `torchvision.models.densenet121`. Evaluations used the same top-1 and top-5 accuracy metrics. This analysis revealed how well attacks on ResNet-34 transferred to structurally different architectures.

Summary of Constraints and Parameters

Property	FGSM	Targeted PGD	Patch PGD
Norm (ℓ)	L_∞	L_∞	L_∞
ϵ	0.02	0.02	0.5
Steps	1	40	40
α	–	0.0025	0.05
Target Area	Full image	Full image	32×32 patch

Table 1: Comparison of attack configurations across FGSM, PGD, and patch-based methods.

Results

ResNet-34 Performance

Table 2 summarizes the top-1 and top-5 classification accuracies of ResNet-34 on the original dataset and three adversarial test sets. As expected, all attacks caused significant degradation in model performance.

Dataset	Top-1 Acc (%)	Top-5 Acc (%)
Original Test Set	76.00	94.20
Adversarial Test Set 1 (FGSM)	43.20	63.20
Adversarial Test Set 2 (Targeted PGD)	1.00	5.80
Adversarial Test Set 3 (Patch PGD)	23.20	43.60

Table 2: Classification accuracy of ResNet-34 across original and adversarial datasets.

Targeted PGD achieved the most drastic accuracy reduction, reducing top-1 accuracy to just 1%. Patch-based PGD also significantly impacted the model despite modifying only a small image region. FGSM, while less aggressive, still resulted in over 30% top-1 accuracy drop, confirming its effectiveness as a lightweight attack.

Transferability to DenseNet-121

Dataset	Top-1 Acc (%)	Top-5 Acc (%)
Original Test Set	74.80	93.60
Adversarial Test Set 1 (FGSM)	5.20	8.00
Adversarial Test Set 2 (Targeted PGD)	4.00	7.60
Adversarial Test Set 3 (Patch PGD)	4.40	7.40

Table 3: Classification accuracy of DenseNet-121 on original and transferred adversarial datasets.

To evaluate the transferability of the attacks, all datasets were also tested on DenseNet-121, a structurally different convolutional model. Table 3 presents the corresponding accuracies.

The adversarial examples transferred effectively to DenseNet-121, with all three attacks reducing top-1 accuracy to below 6%. This indicates a high degree of vulnerability even for models not directly targeted by the attacks. Notably, the patch-based attack maintained high transferability, despite perturbing only a small portion of the input.

Observations and Trends

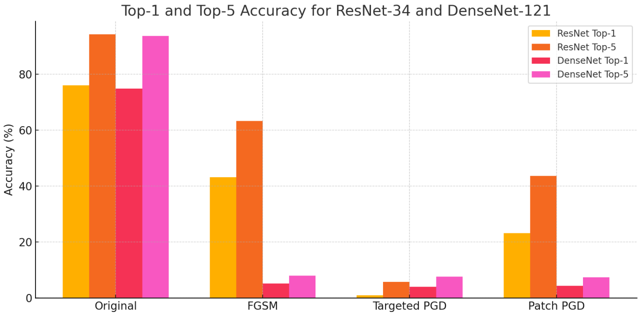

Figure 4: Accuracy Trends

Across both models, Targeted PGD consistently yielded the lowest accuracies, affirming its strength in targeted misclassification. FGSM, although single-step, was still highly effective on both ResNet-34 and DenseNet-121. The success of patch attacks highlights the danger of localized perturbations, especially in real-world applications. Overall, the results emphasize that adversarial robustness requires model-independent solutions, as attack generalization remains a serious concern.

Conclusion and Outcomes

This project demonstrates the vulnerability of deep convolutional neural networks, such as ResNet-34 and DenseNet-121, to adversarial attacks crafted under L_∞ and localized patch constraints. Through systematic evaluation of FGSM, targeted PGD, and patch-based PGD attacks, we were able to significantly degrade model performance on ImageNet-1K test subsets. Notably, targeted PGD reduced ResNet-34 top-1 accuracy from 76% to just 1%, and patch-based attacks achieved similar degradation despite modifying only a 32×32 region. The transferability study showed that these adversarial examples generalize across architectures, reducing DenseNet-121 accuracy to below 6% in all cases. These findings highlight the urgent need for model-agnostic robustness techniques, especially in real-world, black-box deployment scenarios.

The outcomes align with theoretical expectations and prior literature while reinforcing how both simple (FGSM) and iterative (PGD) attacks are effective under minimal perturbation budgets. This validates the practical relevance of adversarial robustness as a critical research area in modern deep learning.

Challenges Faced

1. Aligning the `labels_list.json` file with the PyTorch `ImageFolder` class indices required careful remapping to avoid evaluation mismatches. Without this correction, predicted labels would not align with ground truth, invalidating the accuracy metrics.

2. In Task 3, achieving the required 70% accuracy drop was non-trivial under the $\epsilon = 0.02$ constraint. Basic PGD failed to degrade accuracy sufficiently until a targeted version was implemented, combined with tuning of the step size and iteration count.

3. Patch-based attack (Task 4) introduced spatial constraints, which required masking logic to restrict perturbations to a region 32×32. A larger ϵ value and careful gradient control were necessary to maintain both effectiveness and visual similarity.

Future Work

This project opens up several directions for further exploration. First, defenses such as adversarial training, input gradient regularization, and feature denoising could be evaluated to measure their effectiveness against the implemented attacks. Extending the attack suite to include L_2 and L_0-norm based perturbations, such as DeepFool or Carlini-Wagner (C&W), would also provide deeper insights into model robustness.

Another avenue involves testing adversarial transferability across a broader range of architectures, including vision transformers (ViTs) or hybrid CNN-transformer models, to better understand structural vulnerabilities. From a deployment standpoint, integrating real-time adversarial detection modules into inference pipelines would provide practical robustness.

Finally, incorporating human perceptual evaluation—by quantifying visual similarity between original and adversarial images—could add qualitative validation to complement numerical results.

References

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Kurakin, A.; Goodfellow, I.; and Bengio, S. 2016. Adversarial Examples in the Physical World. In *Proceedings of the Workshop on Artificial Intelligence Safety*, ICLR.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Brown, T.; Mane, D.; Roy, A.; Abadi, M.; and Gilmer, J. 2017. Adversarial Patch. arXiv preprint arXiv:1712.09665.

Liu, Y.; Chen, X.; Liu, C.; and Song, D. 2017. Delving into Transferable Adversarial Examples and Black-Box Attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

OpenAI. 2024. ChatGPT (April 2024 Release). Large Language Model used for formatting, and refinement in this report.

Sabokrou, M.; Khalooei, M.; Fathy, M.; and Adeli, E. 2019. Adversarial examples using FGSM: Generation and visualization. In *Proceedings of the IEEE Conference on Computer Vision*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Zhao, G.; and et al. 2020. Adversarial Patch Generation via GANs. In *Pattern Recognition Letters*.