# Exploring White Box Attack Design and Its Transferability to Black Box Settings

**Aaron Bengochea, Timothy Cao**
**Repository: https://github.com/timothycao/adversarial-attacks**

## Abstract

We present a systematic study of adversarial attacks on pre-trained ImageNet classifiers, focusing on ResNet-34 and DenseNet-121 models. First, we establish baseline top-1 and top-5 accuracies of ResNet-34 on a curated 500-image subset drawn from 100 ImageNet-1K classes. Next, we implement the Fast Gradient Sign Method (FGSM) with an $L_\infty$ perturbation budget of epsilon=0.02 to generate "Adversarial Test Set FGSM," achieving a relative accuracy drop exceeding 50%. Building on this, we explore both our multi-step (PGD) and momentum-based (MI-FGSM) variants to produce "Adversarial Test Set PGD" and "Adversarial Test Set MI-FGSM" ensuring consistent, comparable testing under the same budget and further degrading performance by at least 70%. Following these controlled comparisons, we then adapt our PGD and MI-FGSM attacks to a localized $32 \times 32$ patch with increased epsilon creating "Adversarial Test Set PGD Patch" and "Adversarial Test Set MI-FGSM Patch" to demonstrate vulnerability under sparse, high-magnitude perturbations. In the final analysis, we assess how our diverse PGD-patch and MI-FGSM-patch configurations transfer to a DenseNet-121 model, revealing key patterns in cross-model transfer attack robustness. Our findings highlight both the brittleness of deep classifiers to subtle and localized attacks and key trade-offs in designing effective adversarial methods. Our results indicate that the most effective transfer attack was achieved using MI-FGSM, which produced a 12% drop in top-1 accuracy on DenseNet-121 relative to the baseline, demonstrating that MI-FGSM with a modest perturbation budget of 0.02 can induce material transferable perturbations on ImageNet.

## Overview

To give readers a clear roadmap, we now summarize the five core tasks, each yielding a distinct adversarial testset along with the seminal papers that inspired our methods:

1. **Task 1 – Baseline Evaluation**
   We measure the clean-image performance of a pre-trained ResNet-34 on our 500-image ImageNet subset, reporting top-1 and top-5 accuracies as the reference point for all subsequent attacks.
2. **Task 2 – FGSM Attack**
   We apply the Fast Gradient Sign Method (FGSM)[1] with an $L_\infty$ budget of $\epsilon = 0.02$, to generate Adversarial Test Set FGSM. This single-step attack establishes our first controlled degradation of model accuracy.
3. **Task 3 – PGD & MI-FGSM Iterative Attacks**
   Under the same $\epsilon = 0.02$ constraint, we implement:
   - **Projected Gradient Descent (PGD)**[2], as our multi-step variant named Adversarial Test Set PGD.
   - **Momentum Iterative FGSM (MI-FGSM)**[3], as our momentum-based variant named Adversarial Test Set MI-FGSM.

   By including both, we ensure a consistent, comprable comparison of iterative strategies in the required Task 3 multistep attacks.
4. **Task 4 – PGD & MI-FGSM Patch Attacks**
   We constrain all perturbations to a random $32 \times 32$ patch of each image with an increased budget $\epsilon = 0.3$, yielding two datasets: PGD-Patch and MI-FGSM-Patch. To systematically evaluate attack scaling, we vary the step size $\alpha \in \{0.03, 0.05, 0.10, 0.20, 0.30, 0.40, 0.50, 1.00\}$ and iteration counts $T \in \{10, 30, 50, 100\}$, producing a wide variety of patch configurations. This grid of hyperparameters enables us to quantify how localized perturbation magnitude and iteration depth influence adversarial potency under sparse, targeted attacks.
5. **Task 5 – Transferability of Patch Attacks**
   Finally, We assess the transferability of our PGD-Patch and MI-FGSM-Patch adversarial sets on a pretrained DenseNet-121 (baseline Top-1: 71.8%, Top-5: 93.6%), sweeping the same $\alpha$ and iteration grid from Task 4 to quantify Top-1/Top-5 drops and identify the most transferable patch configurations.

## Task 1: Baseline Evaluation

In Task 1, we establish the clean-image performance of our reference model. We load a ResNet-34 pretrained on ImageNet, set to evaluation mode, and run inference on a held-out subset of 500 images drawn from 100 of ImageNet's 1000 classes. Each image is preprocessed as follows:

1. Normalize using mean $[0.485, 0.456, 0.406]$ and std $[0.229, 0.224, 0.225]$.

We use a batch size of 32 and compute both top-1 and top-5 accuracy. Table 1 reports the resulting baseline metrics.

Table 1: Baseline ResNet-34 performance on clean ImageNet subset (500 images).

| Metric | Accuracy (%) |
|---|---|
| Top-1 Accuracy | 76.00 |
| Top-5 Accuracy | 94.20 |

This baseline establishes reference points against which all adversarial attacks (Tasks 2–4) are compared.

## Task 2: FGSM Attack

In Task 2, we generate our first adversarial dataset by applying the Fast Gradient Sign Method (FGSM)[1] with an $L_\infty$ perturbation budget of $\epsilon = 0.02$ in normalized pixel space. The attack perturbs each input in the direction of the sign of the gradient and then clamps the min and max results to the valid normalized range of $[-2.12, 2.64]$, using an attack budget of $\epsilon = 0.02$.

We apply this function to each image in our 500-image test subset, producing **Adversarial Test Set FGSM**. We then re-evaluate the pretrained ResNet-34 on the perturbed images (batch size 32), computing both top-1 and top-5 on the original subset of the 500 images used during our baseline evaluations. Table 2 summarizes the results.

Table 2: ResNet-34 performance on Clean vs. FGSM-perturbed sets.

| Metric | Accuracy (%) | Drop (%) |
|---|---|---|
| Top-1 Accuracy | 7.00 | 69.00 |
| Top-5 Accuracy | 37.40 | 56.80 |

These results demonstrate that even a subtle FGSM perturbation, invisible to the human eye, can severely degrade model performance, highlighting the vulnerability of pretrained ImageNet classifiers to simple single step adversarial attacks. So, much so that the model can no longer consistently classify the true label within its top 5 classification choices distribution.

## Task 3: PGD & MI-FGSM Iterative Attacks

To further probe the resilience of ResNet-34 under a consistent adversarial budget, we generate two new separate adversarial datasets, **Adversarial Test Set PGD** and **Adversarial Test Set MI-FGSM**, using identical hyperparameters:

$$\epsilon = 0.02, \quad \alpha = 0.005, \quad iterations = 10.$$

### Adversarial Test Set PGD

We take inspiration from the original paper when performing Projected Gradient Descent (PGD)[2]:

1. **Start from the clean image.** Use the original, unmodified image as the baseline.
2. **Repeat the following ten times:**
   (a) **Measure how to fool the model.** Compute the gradient of the loss with respect to the current image to identify the direction that most increases the model's error.
   (b) **Take a small step.** Move each pixel by a step size $\alpha$ in the sign of this gradient to increase the loss.
   (c) **Project back into the $\epsilon$ attack budget.** Ensure that no pixel has changed by more than $\epsilon$ from its original value, keeping perturbations imperceptibly small.
   (d) **Clamp to valid range.** Clip all pixel values to lie within the normalized minimum and maximum bounds.
3. **Output the adversarial image.** After ten iterations of these steps, the final perturbed image is returned as the adversarial example.

We can observe that applying an iterative PGD attack gives us a significant improvement relative to the non-iterative FGSM attack method. Table 3 summerizes the results of the PGD attack relative to the baseline evaluation. It is important to note from our observations that the attacks are powerful enough to trick the model into having absolute confidence while miss-classifying both in the top1 and top5 with high consistency.

Table 3: ResNet-34 performance on Clean vs. PGD-perturbed sets.

| Metric | Accuracy (%) | Drop (%) |
|---|---|---|
| Top-1 Accuracy | 0.00 | 76.00 |
| Top-5 Accuracy | 11.80 | 82.40 |

### Adversarial Test Set MI-FGSM

We take inspiration from the original paper when performing Momentum Iterative FGSM (MI-FGSM) [3]:

1. **Start from the clean image.** Use the original, unmodified image as the baseline.
2. **Repeat the following ten times:**
   (a) **Measure how to fool the model.** Compute the gradient of the loss with respect to the current image to identify the direction that most increases the model's error.
   (b) **Accumulate momentum.** Normalize this gradient and add it to a running momentum buffer to stabilize perturbation direction.
   (c) **Take a small step.** Move each pixel by a step size $\alpha$ in the sign of the momentum-averaged gradient to increase the loss.
   (d) **Project back into the $\epsilon$ attack budget.** Ensure that no pixel has changed by more than $\epsilon$ from its original value, keeping perturbations imperceptibly small.
   (e) **Clamp to valid range.** Clip all pixel values to lie within the normalized minimum and maximum bounds.
3. **Output the adversarial image.** After ten iterations of these steps, the final perturbed image is returned as the adversarial example.

We can observe that MI-FGSM produces performance comparable to PGD while leveraging momentum to guide

perturbations. Table 6 summarizes the results of the MI-FGSM attack relative to the baseline evaluation. Our observations indicate MI-FGSM is similarly effective, often causing the model to assign high confidence to incorrect labels in both top-1 and top-5 categories. These findings suggest that a systematic investigation of alternative iterative attack schemes alongside careful tuning of hyperparameters such as step size, momentum, and extended iteration schedules could yield deeper insights into adversarial robustness.

Table 4: ResNet-34 performance on Clean vs. MI-FGSM-perturbed sets.

| **Metric** | **Accuracy (%)** | **Drop (%)** |
|---|---|---|
| Top-1 Accuracy | 0.20 | 75.80 |
| Top-5 Accuracy | 13.60 | 80.60 |

## Task 4: PGD & MI-FGSM Patch Attacks

We adapt our iterative attacks to a sparse patch setting by restricting all perturbations to a randomly located $32 \times 32$ patch on each image. First, we begin with the hyperparameters choices ($\epsilon = 0.3$, $\alpha = 0.1$, 10 iterations) for both methods, yielding two new adversarial datasets: **Adversarial Test Set PGD Patch** and **Adversarial Test Set MI-FGSM Patch**. These experiments probe model vulnerability when only a small, localized region is perturbed. We then perform a series of hyperparameter tests in an attempt to increase the attack's performance.

### Adversarial Test Set PGD Patch

We perform PGD[2] as before, but at each iteration restrict updates to a random $32 \times 32$ pixel block.

Table 5: ResNet-34 performance on Clean vs. PGD Patch sets.

| **Metric** | **Accuracy (%)** | **Drop (%)** |
|---|---|---|
| Top-1 Accuracy | 40.20 | 35.80 |
| Top-5 Accuracy | 77.00 | 17.20 |

### Adversarial Test Set MI-FGSM Patch

We similarly adapt MI-FGSM[3] to the patch setting, accumulating momentum gradients but restricting each $\alpha$-step to the same random $32 \times 32$ region.

Table 6: ResNet-34 performance on Clean vs. MI-FGSM Patch sets.

| **Metric** | **Accuracy (%)** | **Drop (%)** |
|---|---|---|
| Top-1 Accuracy | 56.20 | 19.80 |
| Top-5 Accuracy | 85.20 | 9.00 |

We expect the attack performance of patch attacks to be significantly less effective than the full image attacks, our observations align with our overall hypothesis.

Next, we evaluate PGD-patch and MI-FGSM-patch attacks under varied parameters to identify configurations that maximize adversarial strength and transferability.

Table 7: Performance of patch attacks on ResNet-34 under varied $\alpha$ and iteration counts with a fixed attack budget of $\epsilon$=0.3.

| | | | MI-FGSM Patch | | PGD Patch | |
|---|---|---|---|---|---|---|
| $\epsilon$ | $\alpha$ | Iterations | Top-1 | Top-5 | Top-1 | Top-5 |
| 0.3 | 0.03 | 10 | 61.0 | 89.4 | 50.6 | 84.8 |
| 0.3 | 0.05 | 10 | 56.8 | 88.2 | 45.8 | 81.2 |
| 0.3 | 0.10 | 10 | 56.2 | 85.2 | 40.2 | 77.0 |
| 0.3 | 0.03 | 30 | 57.2 | 86.4 | 31.6 | 73.2 |
| 0.3 | 0.05 | 30 | 51.0 | 84.2 | 31.0 | 71.0 |
| 0.3 | 0.10 | 30 | 45.8 | 80.0 | 29.4 | 69.4 |
| 0.3 | 0.03 | 50 | 55.4 | 85.6 | 26.0 | 69.4 |
| 0.3 | 0.05 | 50 | 50.6 | 82.2 | 28.6 | 70.4 |
| 0.3 | 0.10 | 50 | 42.8 | 78.0 | 24.6 | 66.2 |
| 0.3 | 0.03 | 100 | 46.2 | 82.2 | 24.6 | 66.6 |
| 0.3 | 0.05 | 100 | 44.4 | 79.6 | 22.6 | 66.0 |
| 0.3 | 0.10 | 100 | 37.0 | 75.2 | 21.0 | 62.4 |
| 0.3 | 0.20 | 100 | 31.4 | 72.2 | 19.6 | 63.6 |
| 0.3 | 0.30 | 100 | 33.0 | 72.6 | 22.2 | 64.6 |
| 0.3 | 0.40 | 100 | 29.4 | 71.6 | 22.6 | 66.4 |
| 0.3 | 0.50 | 100 | 30.6 | 73.2 | 22.4 | 65.0 |
| 0.3 | 1.00 | 100 | 28.8 | 68.2 | 26.2 | 66.0 |

Across both MI-FGSM and PGD patch attacks, we observe that larger step sizes $\alpha$ and increased iteration counts produce progressively stronger adversarial perturbations, as reflected in greater drops in top-1 and top-5 accuracy. Under the same hyperparameters, PGD-based patch attacks consistently outperform their MI-FGSM counterparts, underscoring the efficacy of the projection step. Although random patch placement can introduce run-to-run variability, the overall trend remains clear. Future work could involve fixing the patch location or exhaustively sliding a $32 \times 32$ window across each image to identify the most vulnerable regions and maximize attack potency. Alternatively, one could distribute the perturbation budget across multiple $1 \times 1$ patches collectively occupying a $32 \times 32$ area and apply a similar iterative search to uncover pixel-level configurations that yield even stronger adversarial effects. In white box scenarios, patch based attacks can identify and perturb the most influential pixel regions to maximize missclassification and precisely map decision boundaries. We then evaluate how effectively these perturbations transfer in black-box settings.

## Task 5: Transferability to DenseNet-121

We evaluate the transferability of our patch-based adversarial attacks on DenseNet-121, a deeper convolutional network relative to Resnet32, whose baseline performance on our 500-image subset is Top-1: 74.80% and Top-5: 93.60%. Table 8 reports transfer accuracies under varied patch configurations.

Overall, patch attack scaling has a modest but consistent effect on transferability: the lowest Top-1 accuracy of 68.6%

Table 8: Transferability of patch attacks to DenseNet-121 under varied $\alpha$ and iteration counts ($\epsilon = 0.3$).

| | | | MI-FGSM Patch | | PGD Patch | |
|---|---|---|---|---|---|---|
| $\epsilon$ | $\alpha$ | Iterations | Top-1 | Top-5 | Top-1 | Top-5 |
| 0.3 | 0.03 | 10 | 73.00 | 92.20 | 71.80 | 92.60 |
| 0.3 | 0.05 | 10 | 73.00 | 92.60 | 71.80 | 91.80 |
| 0.3 | 0.10 | 10 | 72.20 | 92.60 | 72.00 | 91.80 |
| 0.3 | 0.03 | 30 | 72.00 | 92.60 | 69.80 | 92.00 |
| 0.3 | 0.05 | 30 | 73.40 | 92.40 | 69.20 | 92.60 |
| 0.3 | 0.10 | 30 | 71.40 | 92.60 | 71.40 | 92.60 |
| 0.3 | 0.03 | 50 | 74.60 | 92.00 | 70.40 | 91.20 |
| 0.3 | 0.05 | 50 | 71.80 | 92.00 | 70.80 | 92.60 |
| 0.3 | 0.10 | 50 | 71.40 | 91.20 | 70.60 | 92.20 |
| 0.3 | 0.03 | 100 | 72.20 | 91.20 | 71.40 | 91.80 |
| 0.3 | 0.05 | 100 | 71.00 | 92.20 | 71.00 | 92.20 |
| 0.3 | 0.10 | 100 | 71.60 | 91.40 | 71.40 | 92.20 |
| 0.3 | 0.20 | 100 | 70.40 | 92.80 | 68.60 | 92.00 |
| 0.3 | 0.30 | 100 | 71.00 | 91.60 | 72.40 | 91.40 |
| 0.3 | 0.40 | 100 | 71.40 | 92.60 | 71.20 | 91.40 |
| 0.3 | 0.50 | 100 | 70.20 | 91.40 | 70.80 | 91.60 |
| 0.3 | 1.00 | 100 | 71.20 | 91.80 | 70.60 | 91.60 |

(a 6.2% drop from baseline) occurs for PGD Patch at ($\epsilon = 0.3$, $\alpha = 0.2$, $iterations = 100$), while the lowest Top-5 accuracy of 91.2% (a 2.4% drop) occurs for MI-FGSM Patch at ($\epsilon = 0.3$, $\alpha = 0.03$, $iterations = 100$). Although further iterations or patch refinements could yield stronger transfer attacks, we limit our experiments to 100 iterations to preserve visual fidelity and avoid excessive distortion.

Please note that although we focused on patch based attacks, evaluating the transferability of the full-image PGD and MI-FGSM methods would likely reveal even greater adversarial potency than their patch based counterparts.

In conclusion, we find that patch-based perturbations reveal that modifying just a 32 × 32 pixel region can substantially degrade ResNet-34's classification accuracy. When applied to a deeper, parameter-rich architecture like DenseNet-121, these localized attacks remain highly effective, underscoring their efficiency across model scales.We observe that adversarial examples optimized in white-box scenarios transfer effectively to black-box targets, suggesting that probing and maximizing perturbations along white box decision boundaries can serve as a proxy for approximating and mapping the boundary landscape and thus internal decision processes of black-box models. Future work should investigate advanced patch optimization strategies such as adaptive region selection and multi patch configurations to identify the most vulnerable pixel areas and maximize adversarial potency for each image, specifically allowing us to map black box settings similarly to our experiments with the transferability onto DenseNet-121.

## References


[1] I.J.Goodfellow, J.Shlens, and C.Szegedy. Explaining and Harnessing Adversarial Examples. arXiv preprint arXiv:1412.6572, 2014

[2] A.Madry, A.Makelov, L.Schmidt, D.Tsipras, and A.Vladu. Towards Deep Learning Models Resistant to Adversarial Examples. arXiv preprint arXiv:1706.06083, 2017

[3] Y.Dong, F.Liao, T.Pang, H.Su, J.Zhu, X.Hu, and J.Li. Boosting Adversarial Attacks with Momentum. arXiv preprint arXiv:1710.06081, 2017