Attacking and Defending Neural Networks

Robert Grgac: s2710099 Gabriel Gausachs: s3436012
Octaviana Cheteles: s2854740 Rebecca Dallabetta: s3538354

January 31, 2025

1 Introduction

This project focuses on implementing an adversarial attack on a pretrained neural network for a classification task, followed by the application of a defense mechanism to enhance the network's robustness. For the attack, we use the Iterative Fast Gradient Sign Method (I-FGSM), an improvement of the Fast Gradient Sign Method (FGSM), which creates adversarial examples by making small adjustments to images. I-FGSM refines this by applying FGSM iteratively with smaller steps, making the attack stronger while keeping changes less noticeable. To defend against these attacks, we applied the Class Label Guided Denoiser (CGD), a model that reduces adversarial noise using a Denoising U-Net (DUNET) structure. The CGD uses the target model's classification loss to guide the removal of adversarial noise, improving the model's ability to resist adversarial interference. The goal is to evaluate the effectiveness of the defense in improving the model's robustness against adversarial attacks.

1.1 Data Set Description

For this project, we used the `ImageNet1k_V1` dataset, which contains a large collection of labeled images categories. The pretrained `ResNet-18` model, trained on this dataset, served as our target classifier. During the adversarial attack phase, we focused on a subset of 15,000 images from ImageNet, applying the Iterative Fast Gradient Sign Method (I-FGSM) to generate adversarial examples. These images were preprocessed to match the input requirements of `ResNet-18`, including resizing, center cropping, and normalization. In the defense phase, the 15,000 adversarial images were used to train the Class Label Guided Denoiser (CGD), with the dataset split into training (80%), validation (15%), and test (5%) sets.

2 Literature Overview

Adversarial attacks on neural networks can be roughly categorized into white-box and black-box attacks. White-box attacks assume full access to the architecture of the model, its parameters, and gradients, allowing attackers to make precise perturbations. Some examples are Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD), which use gradient information to maximize misclassification. Black-box attacks, on the other hand, do not require internal knowledge of the model, but rather rely on asking for the model's outputs or transferring adversarial examples from a surrogate model.

The Iterative Fast Gradient Sign Method (I-FGSM) is an adversarial attack introduced in the paper "Adversarial Examples in the Physical World" by Kurakin et al. (2016) [1]. I is based on the the Fast Gradient Sign Method (FGSM), which was proposed by Goodfellow et al. in 2014 [2]. The difference between these two types of attacks is that FGSM applies a single gradient step, while the other one adjusts the perturbation in smaller steps and iteratively, making it stronger.

This has been an interesting method in the world of adversarial attacks, and multiple researchers deloveped its optimization. For example, Madry et al. (2018) introduced the idea of Project Gradient Descent (PGD), that randomizes the starting point of the perturbation for more effective attacks. Further improvements have been brought by Carlini and Wagner (2017), who improved the security of Neural Networks by finding adversarial attacks that always perform better than I-FGSM [3]. Regardless of these recent findings, and given its strength and computational efficiency, I-FGSM still remains one of the most used adversarial attacks.

Defenses against this type of attack have also been explored. Some examples include adversarial training [4], randomized input transformations [5], and gradient masking [6]. These have been developed with the purpose of disrupting the gradients used in attack or hardening the models against perturbations.

3 Adversarial Attack

We implemented an Iterative Fast Gradient Sign Method (I-FGSM), which is a type of adversarial attack on deep neural networks. It is a targeted perturbation technique applied to input images to make it harder for neural networks to make correct predictions. Usually, the input data is slightly modified to cause incorrect predictions. Our goal was to generate an adversarial example that has undetectable changes to the human eye, and seems visually similar to the original input but is intentionally created to fool the model.

I-FGS applies the FGSM attack multiple times repetitively. It makes small perturbations to the original image on each step, pushing it progressively to the misclassification zone, the decision boundary, while ensuring that the total damage stays within a certain limit. The formula for this attack can be seen below:

$$X_0^{adv} = X, X_{\mathrm{N}+1}^{adv} = \mathrm{clip}_{X,\epsilon}\left(X_{\mathrm{N}}^{adv} + \alpha \cdot \mathrm{sign}\left(\nabla_x J(X_{\mathrm{N}}^{adv}, y_{true})\right)\right)$$

where X is the original input image, y is the true label of the input. $X_{\mathrm{N}+1}^{adv}$ is the adversarial image after the N-th iteration, the step size for each iteration is represented by α, $\nabla_x J(X_{\mathrm{N}}^{adv}, y_{true}$ is the gradient of the loss function J with respect to the input X, and $\mathrm{clip}_{X,\epsilon}$ ensures that the adversarial image stays within the ϵ-ball and valid image bounds with values between 0 and 1.

We chose the I-FGSM attack because, compared to FGSM, which is a single-step attack that fails to cross the decision boundary fully, it is more effective in generating adversarial examples with small steps. Moreover, we can adjust the strength of the attack and the computational cost with the step size α and the number of iterations N. As I-FGSM is a white-box attack, the attacker has complete knowledge of the target model, including its architecture, parameters, and gradients, which allows the attacker to use powerful techniques to craft inputs that fool the model [7]. This type of attack performs well on complex, deep models, especially those trained on large datasets such as ImageNet.

4 Defense

To mitigate the IFGSM attack described in the previous section, we propose a denoising autoencoder-style defense as the most effective solution among the various defense strategies we evaluated. This approach aligns with the methodology presented in the work of Fangzhou et al. (2018), which also secured first place in the Adversarial Attacks and Defenses Competition [8]. The architecture and design principles adopted in this paper are heavily inspired by the aforementioned research.

The proposed defense addresses adversarial images generated by the IFGSM attack by processing them through a denoising autoencoder, which aims to remove adversarial noise, and then passing the resulting images through a ResNet-18 model for classification. Recognizing the challenge of preserving image information through the autoencoder's bottleneck, we implement the U-Net-inspired DUNET architecture.

DUNET, as described in Fangzhou et al. (2018), incorporates a feedforward (encoder) and feedback (decoder) structure. Each component is composed of several layers, with each layer consisting of multiple blocks. Each block includes a 3x3 convolution, batch normalization, and a rectified linear unit (ReLU) activation.

DUNET uses two types of layers: C2 and C3. The C2 layer comprises two connected blocks with a stride of 1×1, while the C3 layer includes three connected blocks with a stride of 2×2. The encoder consists of one C2 layer and four C3 layers, with the final C3 layer serving as the bottleneck. Conversely, the decoder comprises three C3 layers and one C2 layer, followed by a 1×1 convolution. Lateral connections are added between corresponding encoder and decoder layers. In this design, the output tensor from the decoder layer is bilinearly interpolated to match the dimensions of the lateral connection tensor, and these tensors are concatenated as input to the next layer. The architecture is illustrated in Figure 1.

One limitation of traditional denoising autoencoders lies in their loss function, which can inadvertently amplify adversarial noise. The standard loss function aims to minimize the pixel-level noise difference between adversarial and clean images. However, this approach can result in the autoencoder learning and amplifying adversarial noise rather than mitigating it. The work by Fangzhou et al. (2018) highlights this issue and proposes three alternative loss functions based on high-level feature differences:

(a) **Feature Guided Denoiser (FGD):** This loss function compares the feature maps of the topmost convolutional layer (prior to classification) in the ResNet, contrasting features of the denoised image with those of the clean image.

(b) **Logits Guided Denoiser (LGD):** Similar to FGD, but focuses on comparing the logits (outputs from the layer just before the SoftMax function) rather than the convolutional features.

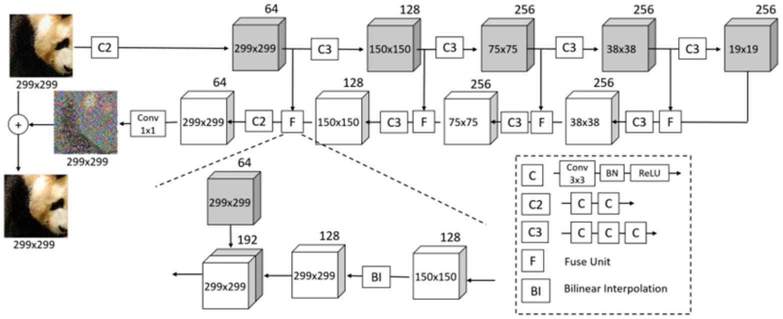

Figure 1: Figure of the DUNET Architecture from [9]. The numbers on top of each cube represent the output channels for that layer. C_k denotes the type of layer, and F represents the concatenation of the lateral tensor with the hidden layer tensor. Also, note that since the input size of our images is not the same as the input used by the Fangzhou et al. (2018), the shape that can be found inside the cubes differs from our implementation

(c) **Class Label Guided Denoiser (CGD):** This compares the classification labels of the clean image with those of the denoised adversarial image.

For our defense strategy, we adopt DUNET with the CGD loss function due to its superior performance compared to FGD and LGD in white-box attack test scenarios, as demonstrated in [9]. The remaining hyperparameters of the DUNET architecture were optimized specifically for this project.

5 Results

5.1 Attack Results

For this project, we implemented both an adversarial attack and a defense mechanism on the ImageNet dataset, using the ResNet-18 model as our target classifier. Our primary objective was to conduct an Iterative Fast Gradient Sign Method (I-FGSM) attack on 15,000 images, generating adversarial examples designed to maximize the model's prediction error while ensuring that the introduced perturbations remained minimal and imperceptible. These adversarial examples were then saved for subsequent analysis and to support the development of our defense strategy.

Before launching the attack, as we mentioned in Section 1.1, we applied standard preprocessing transformations to align the images with the input requirements of the ResNet-18 model. These transformations included resizing the shorter side of the image to 256 pixels, followed by a center crop to a resolution of 224×224. The images were then converted into tensors and normalized using ImageNet's mean and standard deviation.

The I-FGSM attack was carried out using an epsilon value of 0.03, restricting the maximum perturbation applied to each pixel. A step size (alpha) of 0.005 was used, and the attack was performed iteratively over 10 steps, with a batch size of 16 images. The process for each image involved computing the predicted label of the original (non-perturbed) version using the ResNet-18 model before applying the adversarial attack. The I-FGSM algorithm iteratively modified the image, introducing subtle perturbations aimed at altering the model's predictions. After 10 iterations, the resulting adversarial example was saved, along with relevant metadata detailing the image's ground truth label, the predicted label before the attack, and the predicted label after perturbation.

All generated adversarial examples and their corresponding metadata were stored in a dedicated folder, facilitating further analysis and serving as the foundation for evaluating the effectiveness of our defense mechanism. The results demonstrated the substantial impact of adversarial perturbations on the ResNet-18 model's predictions. Out of the 15,000 images attacked, 96% experienced a change in their predicted labels, underscoring the vulnerability of the model to even minor perturbations confined within an epsilon of 0.03.

To better illustrate the impact of the I-FGSM attack, we provide two visualizations comparing the original images with their corresponding adversarial examples. Each figure consists of two images: the original image and its adversarially perturbed version. Additionally, we include the true label of the image, along with the predicted label before and after the attack, highlighting the changes induced by the adversarial perturbation.

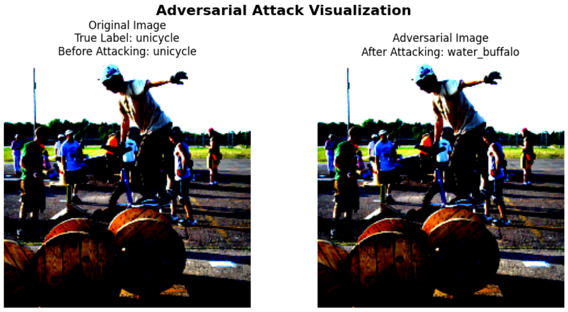
Adversarial Attack Visualization
Original Image
True Label: unicycle
Before Attacking: unicycle
Adversarial Image
After Attacking: water_buffalo

Figure 2: Adversarial Attack Visualization (fooling the model)

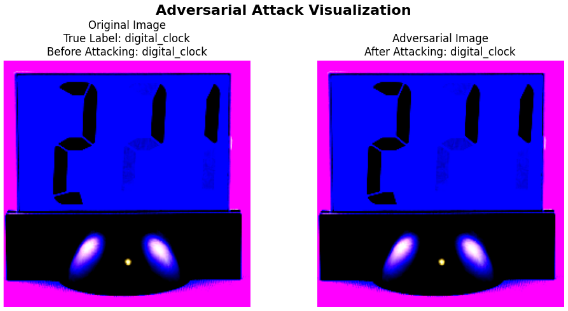
Adversarial Attack Visualization
Original Image
True Label: digital_clock
Before Attacking: digital_clock
Adversarial Image
After Attacking: digital_clock

Figure 3: Adversarial Attack Visualization (not fooling the model)

As shown in Figure 2, the adversarial perturbations are effective in altering the model's predictions. However, as observed in Figure 3, not all images are successfully fooled by the attack, indicating that some examples are more resistant to adversarial perturbations than others.The plots demonstrate the subtle nature of adversarial attacks and their effectiveness in misleading the model, emphasizing the need for robust defense mechanisms to mitigate such vulnerabilities.

5.2 Defense Results

To mitigate the impact of adversarial perturbations, we implemented a defense mechanism using the Class Label Guided Denoiser (CGD), as discussed in 4. The CGD model consists of a Denoised UNet and leverages the classification loss of the target model as the denoising loss function, incorporating supervised learning with ground truth labels. With a total of 11,033,987 trainable parameters, the CGD model provides a computationally efficient yet powerful denoising solution.

For training, we used the 15,000 adversarial images generated during the attack phase, as mentioned in 1.1. The dataset was split into training (80%), validation (15%), and test (5%) sets. The training process followed specific settings, utilizing CrossEntropyLoss (calculated from the target model) as the loss function, the Adam optimizer, and a learning rate of 0.005 over 20 epochs.

To evaluate the effectiveness of the CGD model, we considered two primary metrics: loss and accuracy. The loss was computed using the target model's classification loss (CrossEntropyLoss), measuring the difference between predicted and true labels. Accuracy was determined by calculating the percentage of correctly classified images after denoising.

After training for 20 epochs, we analyzed the performance of the CGD model. The following plots illustrate the training and validation loss, as well as the accuracy progression over epochs.

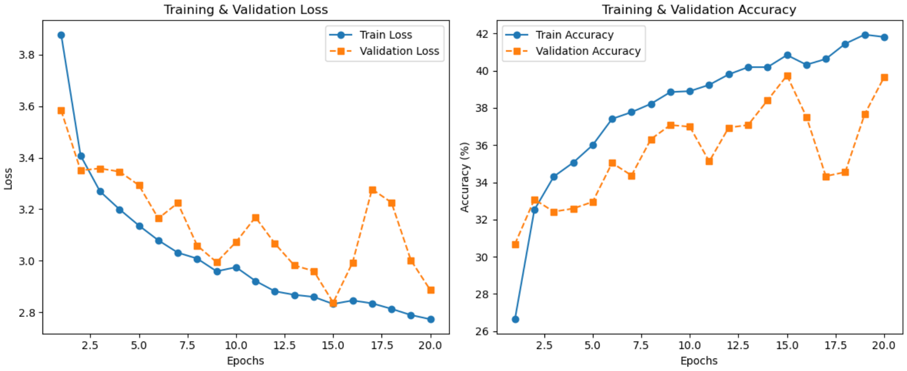

Figure 4: Training and Validation Loss and Accuracy over 20 Epochs

The results presented in Figure 4 show a clear trend of improvement over the 20 epochs. Both the training and validation losses steadily decrease, while accuracy in each set progressively increases. Starting with a training accuracy of 26.5% in the first epoch, the model achieves a final accuracy of 41% by epoch 20. Similarly, the validation accuracy follows a comparable pattern, reaching almost 40% at the end of the training.

Finally, when classifying the test set of adversarial images, the target model achieved an accuracy of **27%** without any defense. With the inclusion of the denoised UNet, this accuracy increased to **40%**, demonstrating the effectiveness of the denoising approach in mitigating the impact of adversarial perturbations. These findings emphasize the importance of incorporating denoising techniques as a defense strategy, highlighting the potential of CGD for enhancing model robustness against adversarial attacks.

6 Discussion

As observed in 5, the I-FGSM attack was successful in altering the model's predictions for a significant proportion of the images. The adversarial examples maintained a high degree of visual similarity to the original images, demonstrating the effectiveness of the attack in creating imperceptible perturbations.

To counteract these adversarial perturbations, we employed the Class Label Guided Denoiser (CGD), which consists of a Denoised UNet. Our results indicate that the CGD model successfully increased classification accuracy on adversarial examples, restoring correct predictions for a significant proportion of the perturbed images. While the defense mechanism proved effective, some adversarial examples remained difficult to correct, suggesting that further improvements could be made. One potential avenue for enhancing performance could involve the use of additional training data or the application of data augmentation techniques. By increasing the diversity of the training set, the model would be exposed to a broader range of examples, potentially improving its ability to generalize and resist adversarial attacks.

It is important to note that while the accuracy of our model is not exceptionally high, we acknowledge the possibility of further optimization and improvements in future work. However, despite this, the model with the defense mechanism performs significantly better than the baseline (without defense), and that is the most critical takeaway—demonstrating that the defense strategy provides tangible benefits in combating adversarial threats.

Additionally, incorporating alternative loss functions, as discussed in 4, may further strengthen the model's resilience. These strategies, along with the current defense, may provide more comprehensive protection against a wider range of adversarial threats.

7 Conclusions

In this project, we successfully implemented the Iterative Fast Gradient Sign Method (I-FGSM) attack on a pretrained ResNet-18 model using the ImageNet dataset. The attack was effective, changing the model's predictions for 96% of the adversarial examples, even with small perturbations. Visualizations showed how these modifications are indeed very subtle, demonstrating the attack's ability to mislead the model while maintaining image similarity.

To counteract these adversarial perturbations, we implemented the Class Label Guided Denoiser (CGD), which showed promising results in enhancing the model's robustness. The CGD model improved classification accuracy on adversarial examples, effectively restoring the correct predictions for a significant number of the perturbed images.

However, some adversarial examples remained resistant to the defense, indicating that further improvements, such as adversarial training or hybrid defense strategies, could be explored to further strengthen the model's resistance to attacks. These results highlight the importance of developing more robust defense mechanisms to defend deep learning models from adversarial threats.

You can explore the complete project, including the code and additional resources, in our repository [10].

References

[1] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *CoRR*, abs/1607.02533, 2016.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[3] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks, 2017.

[4] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[5] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan Yuille. Improving transferability of adversarial examples with input diversity, 2019.

[6] Franziska Boenisch, Philip Sperl, and Konstantin Böttinger. Gradient masking and the underestimated robustness threats of differential privacy in deep learning, 2021.

[7] Aidan Thompson. White-box adversarial attacks in ai, 2024.

[8] Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, Alan Yuille, Sangxia Huang, Yao Zhao, Yuzhe Zhao, Zhonglin Han, Junjiajia Long, Yerkebulan Berdibekov, Takuya Akiba, Seiya Tokui, and Motoki Abe. Adversarial attacks and defences competition. In Sergio Escalera and Markus Weimer, editors, *The NIPS '17 Competition: Building Intelligent Systems*, pages 195–231, Cham, 2018. Springer International Publishing.

[9] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018.

[10] Robert Grgac, Gabriel Gausachs, Octaviana Cheteles, and Rebecca Dallabetta. Attacking and defending neural networks, 2025. If you use this code in your work, please cite it as follows.

[11] Simon J.D. Prince. *Understanding Deep Learning*. The MIT Press, 2023.

[12] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019.

AI statement

During the preparation of this work, we used ChatGPT, Claude and Grammarly to rephrase and restructure sections of this document and for coding. Furthermore, Perplexity was occasionally used as a search engine to find and cite sources and references. After using these tools/services, we thoroughly reviewed and edited the content as needed, taking full responsibility for the final outcome.