Adversarial Defense via Region-Adaptive Perlin Noise Training

Javier Fernandez
Purdue ID: 0038136209
Purdue University
ferna321@purdue.edu

1 Introduction

Deep neural networks show impressive performance in image classification tasks but are still weak to adversarial attacks, where undetectable perturbations to input images can cause misclassification. This represents a major security concern for real-world applications (e.g., autonomous vehicles or medical images). There is a trade-off between accuracy and security because traditional methods can't maintain performance on clean images while obtaining high metrics against adversarial input.

My goal in this project is to learn $p(y|x)$, where y represents class labels and x represents input images, assuming that both the training and test data contain approximately 50% adversarial examples, maintaining a consistent distribution in both sets. The main objective is to create a defense mechanism using Perlin noise, a procedural noise function that produces smooth patterns. This type of noise creates structured perturbations that may be able to break adversarial patterns while preserving semantic features.

This approach applies Perlin noise based on region importance within images. By identifying critical regions using gradient-weighted class activation mapping (Grad-CAM) and applying lower intensity noise to these areas, the model can learn to distinguish between natural variations and adversarial manipulations. This strategy aims to enhance performance against procedural noise attacks while minimizing the impact on accuracy for clean images. The problem can be further formulated in statistical learning terms: when an adversarial perturbation function $\mathcal{A}(x)$ is applied to an input x, it creates x' such that $||x - x'||_p < \epsilon$ for some small ϵ, yet a model f produces $f(x) \neq f(x')$.

My approach addresses this issue by including these distributional shifts during training. By augmenting training data with region-adaptive Perlin noise, I create a mixed distribution that represents both clean and potential adversarial inputs. This mixed distribution training helps the network learn decision boundaries that are less sensitive to small perturbations that would otherwise cause misclassification. The statistical challenge in this defense mechanism centers on balancing two objectives: maintaining the original conditional probability distribution $p(y|x)$ for clean images while simultaneously learning a robust distribution $p(y|x')$ for perturbed images. Traditional defenses often sacrifice performance on one distribution to improve on the other. My region-adaptive approach aims to optimize performance across both distributions by preserving critical features in important regions while disrupting potential adversarial patterns in less important areas.

Additionally, Perlin noise presents specific advantages over other types of noise due to its coherent structure. The smooth gradients in Perlin noise better approximate natural image variations, making it more suitable for data augmentation that preserves semantic content. By tuning parameters like octaves, frequency or persistency, Perlin noise can be adjusted to match different scales of image features (e.g., from textures to larger structures) making it versatile for defending against various types of attacks.

2 Dataset

This project uses the CIFAR-10 dataset, which consists of 60,000 32×32 color images across 10 classes (6,000 images per class), divided into 50,000 training and 10,000 testing obser-

vations. The images in this dataset have a relatively low resolution that allows for experimentation with noise parameters without requiring excessive computational resources. The diverse classes (e.g., animals, vehicles, and objects) let us evaluate the defense mechanism with different content.

The dataset's small image size ensures that important features occupy a significant part of each image, making the adaptive-region approach more effective. The adversarial examples will be generated from this dataset using procedural noise with randomized parameters, such as octaves, scale, persistency and lacunarity to evaluate defense performance. Both the training and testing datasets are augmented with perturbed samples, creating a balanced 50/50 distribution of clean and noisy images.

3 Proposed approach

The proposed defense mechanism uses an approach centered around region-adaptive Perlin noise augmentation. First, we use Grad-CAM to generate importance maps for input images. Grad-CAM produces heatmaps highlighting influential regions in the network's predictions by looking at gradient information flowing into the final convolutional layer.

Figure 1 shows an example of a CIFAR-10 image and its corresponding Grad-CAM heatmap. The heatmap clearly highlights the critical regions that influence the network's classification decision, with red areas indicating higher importance and blue areas indicating lower importance. In this example, we can observe that the network primarily focuses on the central animal in the image for making its classification decision, while the background areas receive less attention.

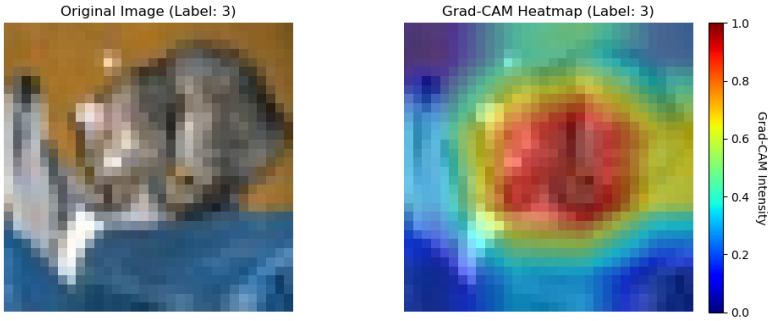

Figure 1: Grad-CAM Heatmap

Once these regions are identified through Grad-CAM, Perlin noise is applied with different intensities based on the heatmap values. Critical regions have reduced noise amplitude (by scaling noise with $1 - n \times \text{gradcam\_map}$), while non-critical regions retain full amplitude. This selective application preserves the semantic content necessary for correct classification while still disrupting potential adversarial patterns in less important areas.

Perlin noise is a procedural noise function that produces naturally-appearing random patterns with smooth variations. As previously mentioned, the noise is defined by several parameters:

- **Scale**: Controls the size of the noise features.
- **Octaves**: Determines the number of noise layers combined.
- **Persistence**: Controls the amplitude reduction for each octave.
- **Lacunarity**: Controls the frequency increase for each octave.

Figure 2 demonstrates how different parameter configurations significantly affect the generated noise patterns:

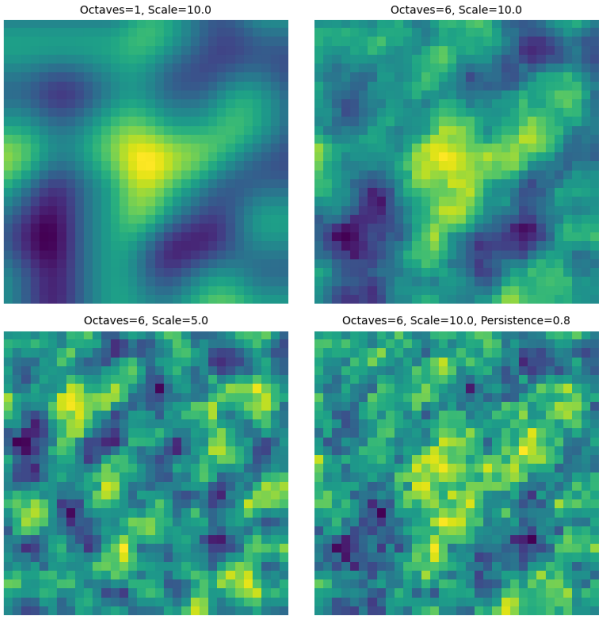

Figure 2: Noise Patterns

During training, Perlin noise parameters are randomly sampled to enhance generalization. These parameters are limited to a predefined range to generate patterns that disrupt adversarial perturbations while maintaining image clarity. The randomization prevents the model from adapting to a specific noise pattern. Instead, it forces the model to learn features that are invariant to a range of procedural noise patterns, improving generalization to unseen adversarial examples.

For the neural network architecture, this project uses ResNet-18 (CIFAR variant), a deep convolutional neural network with residual connections. The network consists of an initial 3x3 convolutional layer followed by max pooling, four residual blocks (each containing multiple convolutional layers with batch normalization and ReLU activations), and a final fully connected layer. The residual connections allow the network to learn without suffering from vanishing gradient problems.

The training process follows these steps:

1. Pre-train ResNet-18 on clean CIFAR-10 images to establish a baseline model.
2. Generate Grad-CAM importance maps for all training images using the pre-trained model.
3. Create an augmented dataset containing both original images and perturbed versions with region-adaptive Perlin noise.
4. Fine-tune the pre-trained model on this augmented dataset using a mixed batch approach.

The augmented training dataset contains 100,000 images: the original 50,000 CIFAR-10 training images and 50,000 perturbed versions created using the region-adaptive Perlin noise approach. During training, batches are constructed to contain an equal mix of clean and perturbed images, maintaining a balanced representation that helps the network learn features that are robust across both distributions.

Figure 3 compares the adaptive approach to uniform noise application. The original image (left) shows a frog visible against its background. When regular Perlin noise is applied uniformly (middle), important features (i.e, frog's body) are concealed. However, with

adaptive Perlin noise (right), the noise intensity is reduced in critical regions identified by Grad-CAM while maintaining stronger noise in less important areas (e.g., the background).

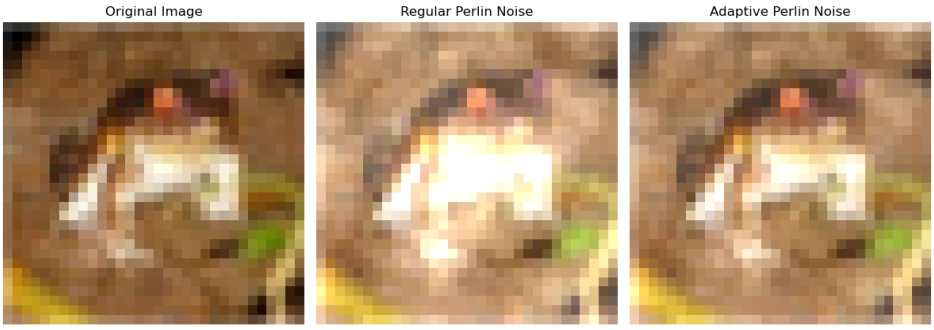

Figure 3: Uniform Perlin noise vs Adaptive Perlin noise

The optimization objective follows the standard cross-entropy loss for classification tasks:

$$L(y, \hat{y}) = -\sum_{i=1}^{10} y_i \log(\hat{y}_i) \tag{1}$$

Where y is the one-hot encoded ground truth label and $\hat{y}$ is the predicted probability distribution over the 10 CIFAR-10 classes. Because our goal is to maximize classification accuracy on this task, we minimize the categorical cross-entropy between the predicted distribution and the one-hot ground-truth label. This penalizes probability mass assigned to incorrect classes, aligning with the training objective. The model is trained using Stochastic Gradient Descent (SGD) with momentum 0.9, learning rate of 1×10^{-3} and weight decay of 1×10^{-4}.

After training, the model is evaluated against various adversarial attacks, including procedural noise attacks with fully randomized parameters and Fast Gradient Sign Method (FGSM) attacks. The evaluation will measure both clean accuracy and adversarial accuracy to ensure that the defense maintains performance on clean images while improving robustness against adversarial examples.

4 Related Work

Adversarial defenses have been studied since the vulnerability of neural networks to perturbations was discovered (Goodfellow et al., 2014). The authors introduced the Fast Gradient Sign Method, which generates examples by taking a step in the direction of the gradient of the loss function. Defensive strategies have followed two approaches. Adversarial training uses adversarial examples in the training, but suffers from high computational costs and reduced clean accuracy. Meanwhile, input transformation methods try to remove perturbations before classification.

Research has demonstrated that Perlin noise can improve adversarial robustness against procedural noise attacks, though previous approaches applied uniform noise across entire images. The Perlin noise-based data augmentation strategy has shown promise in improving CNN model performance for classifying high resolution images (Tang et al., 2021).

Moreover, computational efficiency is still a challenge, as adversarial training requires generating adversarial examples during each training iteration, often on GPUs. My approach with Perlin noise offers potential advantages, as the noise patterns can be precomputed or generated efficiently on CPUs, reducing overhead.

5 Results

The success of this project depends on several metrics to compare the proposed defense against baseline models:

- The defense should maintain clean test accuracy within 10% of the baseline model.
- The defense should achieve at least 35% adversarial accuracy under procedural noise attacks (i.e., randomized Perlin noise attacks).
- The defense should attain at least 20% adversarial accuracy under a gradient-based attack like FGSM.

Additionally, computational efficiency is tested, aiming for a low training time and GPU memory usage, which is measured by analyzing the total training duration and resource requirements.

To evaluate whether the defense maintains classification accuracy on clean images, we compare the test accuracy of the baseline ResNet-18 model against the model trained with region-adaptive Perlin noise:

Model	**Clean Test Accuracy (%)**
ResNet-18 Baseline	93.07
Region-Adaptive Perlin Noise ($n = 0.7$)	91.12

Table 1: Classification accuracy on clean CIFAR-10 test images

The defended model achieves 91.12% accuracy on clean images, which represents a drop of only 1.95% compared to the baseline.

To evaluate robustness against procedural noise attacks, we test both models on a dataset where 50% of the images are perturbed with randomized Perlin noise and 50% are clean:

Model	**Accuracy Under Perlin Noise Attack (%)**
ResNet-18 Baseline	81.44
Region-Adaptive Perlin Noise ($n = 0.7$)	88.82

Table 2: Classification accuracy under Perlin noise attack

The region-adaptive defense outperforms the baseline model, achieving 88.82% accuracy against Perlin noise attacks compared to 81.44% for the undefended model.

To evaluate performance against gradient-based attacks, both models are tested on a dataset where 50% of the images are perturbed with FGSM (Fast Gradient Sign Method) attacks using $\epsilon = 0.03$, and 50% are clean.

Model	**Accuracy Under FGSM Attack (%)**
ResNet-18 Baseline	36.98
Region-Adaptive Perlin Noise ($n = 0.7$)	74.34

Table 3: Classification accuracy under FGSM attack ($\epsilon = 0.03$)

The defended model achieves 74.34% accuracy against FGSM attacks, outperforming the baseline by 37.36%. This satisfies the third success criterion of achieving at least 20% adversarial accuracy against gradient-based attacks.

Furthermore, different variations of the approach were tested, and results demonstrate that the region-adaptive approach with $n = 0.7$ provides the optimal balance between clean accuracy and adversarial robustness. Training efficiency was also measured in terms of total training time and GPU memory usage:

Model	Training Time (hours)	Peak GPU Memory (MB)
Region-Adaptive ($n = 0.5$)	0.14	832.60
Region-Adaptive ($n = 0.7$)	0.14	834.35
Region-Adaptive ($n = 0.9$)	0.14	834.35

Table 4: Computational efficiency comparison

Model Variant	Clean Acc. (%)	Perlin Attack Acc. (%)
Region-Adaptive (n=0.5)	90.51	87.78
Region-Adaptive (n=0.7)	91.12	88.82
Region-Adaptive (n=0.9)	90.29	87.70

Table 5: Ablation study results comparing different noise adaptation strategies

The results confirm that the region-adaptive Perlin noise defense successfully achieves all success criteria. Furthermore, the defense demonstrates good computational efficiency compared to traditional adversarial training methods like FGSM.

Overall, procedural noise attacks were found to be relatively ineffective at reducing the accuracy of the baseline model, as it already shows strong resistance with an accuracy of 81.44%. My adaptive defense only provides an improvement of 7.38%, reaching 88.82%. Nevertheless, the small decrease in accuracy on clean data (from 93.07% to 91.12%) shows that the defense mechanism does not seriously hurt overall model performance, making it practical to implement in specific cases.

More importantly, this fine-tuning approach showed a significant improvement against gradient-based adversarial attacks (FGSM), surpassing the baseline model's accuracy of only 36.98%. Thus, despite the limited advantage against procedural noise attacks, this new strategy is highly effective as a general adversarial defense, particularly against FGSM, and presents a solution for enhancing robustness in real-life scenarios.

6 References

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and Harnessing Adversarial Examples. https://arxiv.org/abs/1412.6572

Huy Phan. (2021). huyvnphan/PyTorch_CIFAR10 (v3.0.1). Zenodo. https://doi.org/10.5281/zenodo.4431043

Tang, C., Zhang, K., Xing, C., Ding, Y., & Xu, Z. (2021, December 26). Perlin Noise Improve Adversarial Robustness. https://doi.org/10.48550/arXiv.2112.13408