# Adversarial Attacks on Watermarks

**Andy Truong**
amt007@ucsd.edu

**Anushka Purohit**
apurohit@ucsd.edu

**Wang, Yu-Xiang**
yuw272@ucsd.edu

## Abstract

This project offers an innovative technique in the creation of an invisible watermark, leveraging the adversarial attack Projected Gradient Descent (PGD) that adds small perturbations to an image to misclassify it. As the rise of generative AI has caused for a raise of concerns on determining if a certain form of media is real. There needs to be a watermark that can be robust to sophisticated forms of watermark removal attacks. Therefore, our approach is designed in rivalry of the Stable Diffusion Regeneration Attack (SDRA) that we explored in our Project 1. The PGD watermark allows for a dual functionality: targeted incorrect labeling, fooling a classification model, for verification and preservation of the original classification for content authenticity. We used two methods of watermarking for this approach, a Black-Box and White-Box approach, which seems to have no significant difference in effectiveness. We purposely select k random labels to perturbed noise towards onto the original image, and the original image will still maintain its dominant classification. We verify if the image is watermarked to check if a certain percentage of target labels logit scores increase. Running it through the SDRA with a certain number of k and an epsilon of maximum allowed perturbation, we were able to withstand the attack. With this approach, we are able to offer a new solution to the battle of watermark removal in order to maintain and protect the authenticity of a digital image to avoid misrepresentation and the spread of misinformation.

Website: https://ndxdxd.github.io/AdversarialAttacksOnWatermarks/
Code: https://github.com/ndxdxd/Capstone-Q2/tree/main

# 1 Introduction

In an age where digital media runs rampant, there is a proliferating need for a method to protect the rights and authenticity of said media. Especially with the rise of Generative AI, users can produce content nearly identical to real life that could potentially fool an uneducated consumer. Thus, invisible watermarking remains as one of the most viable methods to achieve this goal. They maintain the quality of the image and are least likely to be removed by the general public. Nevertheless, people who are aware of these watermarks can execute watermark attacks, getting rid of the invisible watermarks from the AI-generated images which can lead to the problems such as misrepresentation and misinformation.

There are attacks such as destructive attacks which significantly reduces the quality of the image but are effective at removing the invisible watermark, such as Gaussian Blur, Gaussian Noise, and Cropping. There are also regeneration attacks, where a watermarked image is destroyed then reconstructed using a Generative AI model, such as the Stable Diffusion Watermark Removal Attack we explored in our Project 1. This attack is the strongest attack within the watermarking field. As a result, there needs to be another innovative watermarking technique that could be potentially robust to this attacks.

In this project, we are focusing on creating an innovative watermark with the goal of it being robust to the regeneration attack. The watermark that we have created is guided by an adversarial attack. An adversarial attack is an attack which exploits vulnerabilities in machine learning systems by introducing small perturbations that lead to large errors that causes misclassification. The adversarial attack which we are using is known as the Projected Gradient Descent attack (PGD). PGD works by repeatedly applying small perturbations to an image in the direction that maximally confuses a given model, while ensuring the modifications remain within a defined constraint to maintain image integrity.

There has been previous work that has attempted to explore more about adversarial attacks such as the "Adversarial Machine Learning at Scale" paper by Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio, where they discussed the different types of adversarial attacks such Fast Gradient Sign Method (FGSM) as well as the results of them. They found out that training on adversarial data showed robustness to adversarial examples and that the increase of parameters in a model could also increase the robustness of adversarial examples as well. However, there has not been any work where instead of making a watermark robust to adversarial attacks, making a watermark based off of adversarial attacks.(Kurakin, Goodfellow and Bengio 2016)

By incorporating adversarial perturbations into our watermarking technique, we aim to create a watermark that remains resilient even after an image undergoes regeneration attacks. Since generative models often rely on learned patterns and statistical correlations to reconstruct images, our adversarially crafted watermark disrupts these patterns, making it significantly harder for the model to remove or ignore the watermark. This ensures that ownership and authenticity can still be verified even after AI-based modifications.

For our data, we will be using the ImageNet dataset and for the classification model, we will be using the ResNet50 model to use to build our watermarking technique, as well as an ImageNet Validation Set which we found on Kaggle that contains 1000 images.

# 2 Methods

We began by exploring the PGD adversarial attack to understand the algorithm and how to begin developing our watermark. We understood that adversarial attacks are able to fool machine learning models so we wanted to create a watermark inspired by an adversarial attack. We believe that this watermark would be robust to watermark removal attempts from neural network models such as the regenerative stable diffusion removal attempt. (Madry et al. 2017)

## 2.1 Targeted PGD attack

The Projected Gradient Descent (PGD) attack is a widely used adversarial attack that iteratively perturbs an input sample to maximize the loss of a deep learning model while keeping the perturbation within a specified norm constraint. This method is an extension of the Fast Gradient Sign Method (FGSM) but applies multiple iterative updates with projection to ensure that the perturbation remains within a bounded region. PGD is particularly effective in generating adversarial examples that can mislead neural networks while remaining imperceptible to the human eye, which is how we plan to embed our watermark. Given a neural network classifier $f(\cdot)$ with parameters $\theta$, the PGD attack aims to find an adversarial perturbation $\delta$ such that the perturbed input $x' = x + \delta$ is misclassified while ensuring $\delta$ remains within a bounded set.

For a targeted attack, where the goal is to force the model to classify $x'$ as a specific target class $y_{\text{target}}$, the loss function is modified as follows:

$$\delta^* = \arg\min_{\delta \in S} \mathscr{L}(f(x + \delta; \theta), y_{\text{target}}) \tag{1}$$

where:

- $x$ is the original input,
- $y$ is the true class label,
- $\mathscr{L}$ is the loss function (e.g., cross-entropy loss),
- $S$ is the feasible set of perturbations, often defined as $S = \delta \mid |\delta|_p \leq \epsilon$ for some norm $p$ and bound $\epsilon$.

This ensures that the perturbed sample is classified as $y_{\text{target}}$ instead of its true label $y$.

We begin our approach by testing out two different types of methods. We are going to be exploring both White-Box, where we can edit the Projected Gradient Descent attack, and Black-Box, where we iteratively call on the function without changing its base function.

## 2.2 White-Box Approach Watermark

For our new watermark, we use the PGD attack as a base. However, we pick a random subset of $k$ labels and aim to increase the logit scores (model's predicted probability) for these labels. The watermark is designed to subtly alter the image such that the model's output probabilities for specific target classes are significantly affected, while the overall visual appearance of the image remains largely unchanged.

The watermark is applied as an adversarial perturbation $\delta$ to the original image $x$. The perturbation is generated using a targeted adversarial attack, which aims to maximize the model's confidence in a set of target classes while keeping the primary classification stable. The perturbation is constrained by an $\ell_\infty$-norm bound to ensure that it remains imperceptible.

Given a pre-trained classifier $f(\cdot)$, an input image $x$, a true class label $y_{\text{true}}$, and a set of target class labels $\mathscr{T}$, we define the objective function as:

$$\begin{aligned} \delta^* &= \arg\max_{\|\delta\|_\infty \leq \epsilon} \mathscr{L}(f(x+\delta), y_{\text{target}}) \\ \mathscr{L}(x+\delta) &= \mathscr{L}_{\text{true}} + \lambda \cdot \mathscr{L}_{\text{target}}, \end{aligned} \tag{2}$$

where:

- $\mathscr{L}_{\text{true}} = \mathscr{L}_{\text{SCC}}(f(x+\delta), y_{\text{true}})$ ensures that the model still classifies the image correctly,
- $\mathscr{L}_{\text{target}} = \frac{1}{|\mathscr{T}|}\sum_{y\in\mathscr{T}} \mathscr{L}_{\text{SCC}}(f(x+\delta), y)$ encourages the model to increase logits for the secret target labels,
- $\lambda$ is a weighting parameter to balance the two objectives.

The perturbation $\delta$ is optimized iteratively using gradient descent:

$$\delta \leftarrow \delta - \alpha \frac{\partial \mathscr{L}}{\partial \delta} \tag{3}$$

where $\alpha$ is the step size. To ensure imperceptibility, we apply clipping:

$$\delta \leftarrow \text{clip}_{[-\epsilon,\epsilon]}(\delta). \tag{4}$$

After optimization, the final watermarked image is:

$$x^* = \text{clip}(x+\delta, 0, 255). \tag{5}$$

This ensures the pixel values remain within the valid image range.

The effectiveness of the watermark is evaluated by checking the logits of the target labels before and after applying $\delta$, ensuring a significant increase while keeping the original classification intact.

## 2.3 Black-Box Watermarking Approach

In our black-box watermarking approach, we introduce a perturbation that alters the model's output probabilities for a randomly selected subset of $k$ labels without direct access to the model's gradients. Unlike the white-box approach, where the adversarial perturbation is optimized using gradients, the black-box method relies on an iterative query-based optimization process.

The watermarking process involves modifying the input image $x$ to maximize the probability of specific target labels. Given a pre-trained model $f$, we define the perturbation $\delta$ as follows:

$$\delta^* = \arg\max_{\|\delta\|_\infty \leq \epsilon} \mathscr{L}(f(x+\delta), y_{\text{target}})$$

where:

- $f$ is the pre-trained ResNet-50 model,
- $\mathscr{L}$ is a surrogate loss function based on model output probabilities rather than gradients,
- $y_{\text{target}}$ represents one of the selected target class labels,
- $\epsilon$ is the maximum allowed perturbation magnitude.

Since we do not have access to the model's internal gradients, we iteratively perturb the image and measure the change in logits. Specifically, we:

1. Select $k$ target labels at random.
2. Query the model with the original image $x$ to obtain the baseline logits.
3. Apply a small perturbation $\delta$ and measure the resulting logits for each of the target labels in the loop.
4. Update $\delta$ using an optimizer (e.g., Adam) to maximize the probability of the target labels.
5. Scale the perturbation by $\sqrt{k}$ to ensure the adjustment accounts for multiple target labels:

$$\delta_{\text{scaled}} = \delta \cdot \sqrt{k}$$

## 2.4 Watermark Verification

For both of these approaches, we used the same watermark verification scheme. While there were many possibilities for what our watermark verification could be, we decided to verify our watermark by setting a percentage threshold for the amount of target labels logits scores that were raised. If a certain amount of target labels logit scores were raised, then we would say that the image is watermarked as we are simply looking if watermarked image is labeled incorrectly.

$$P = \frac{|\{i \mid \Delta\ell_i > 0\}|}{|S|}$$

$$\text{Watermark verified if } P \geq p.$$

where:

- $P$ is the proportion of target labels with increased logit scores.

- $S$ is the set of all target labels.

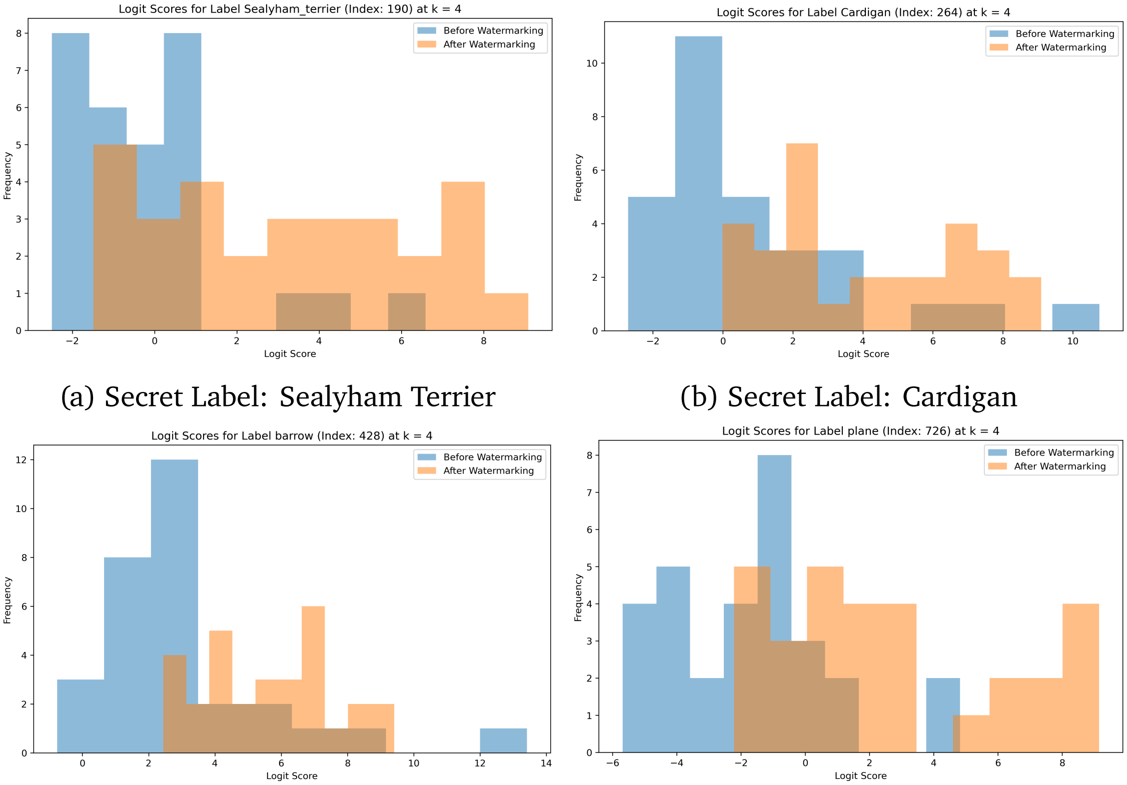
- $\Delta\ell_i = \ell_{\text{watermarked},i} - \ell_{\text{original},i}$ is the logit difference for the target label $i$.

- $p$ is the predefined percentage threshold for verification

For the purposes of consistency for our report and our experimentation, we decided to use k = 4, and the secret labels are Plane, Cardigan, Barrow, Sealyham Terrier, and the $\epsilon$ value for white-box is 3.5/255 and black-box is .7/255 as well as a threshold of 50%

## 2.5 Logit Distributions

To evaluate the effectiveness of our watermarking scheme in increasing logit scores across images, we generated distribution plots for each individual label as well as the overall distribution across all tested images. This allowed us to analyze how the watermarking process influenced the model's confidence in its predictions and assess whether it consistently raised the logit scores. Additionally, our distribution was made from 30 images, all put under the same target label for the attack.



(a) Secret Label: Sealyham Terrier

(b) Secret Label: Cardigan

(c) Secret Label: Barrow

(d) Secret Label: Plane

Figure 1: Logit Distribution of Secret Labels

### 2.5.1 White-Box

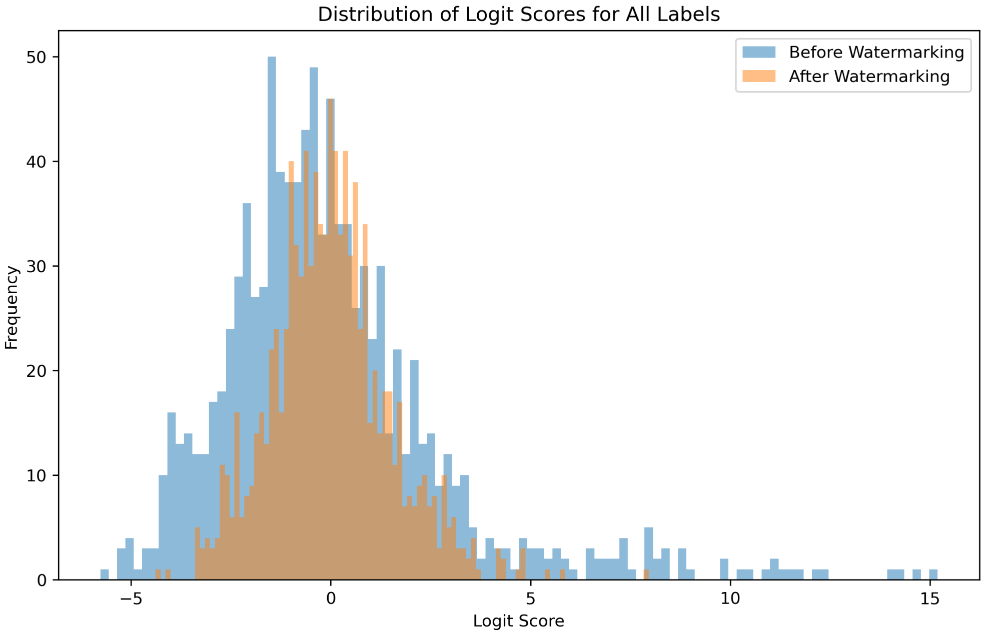
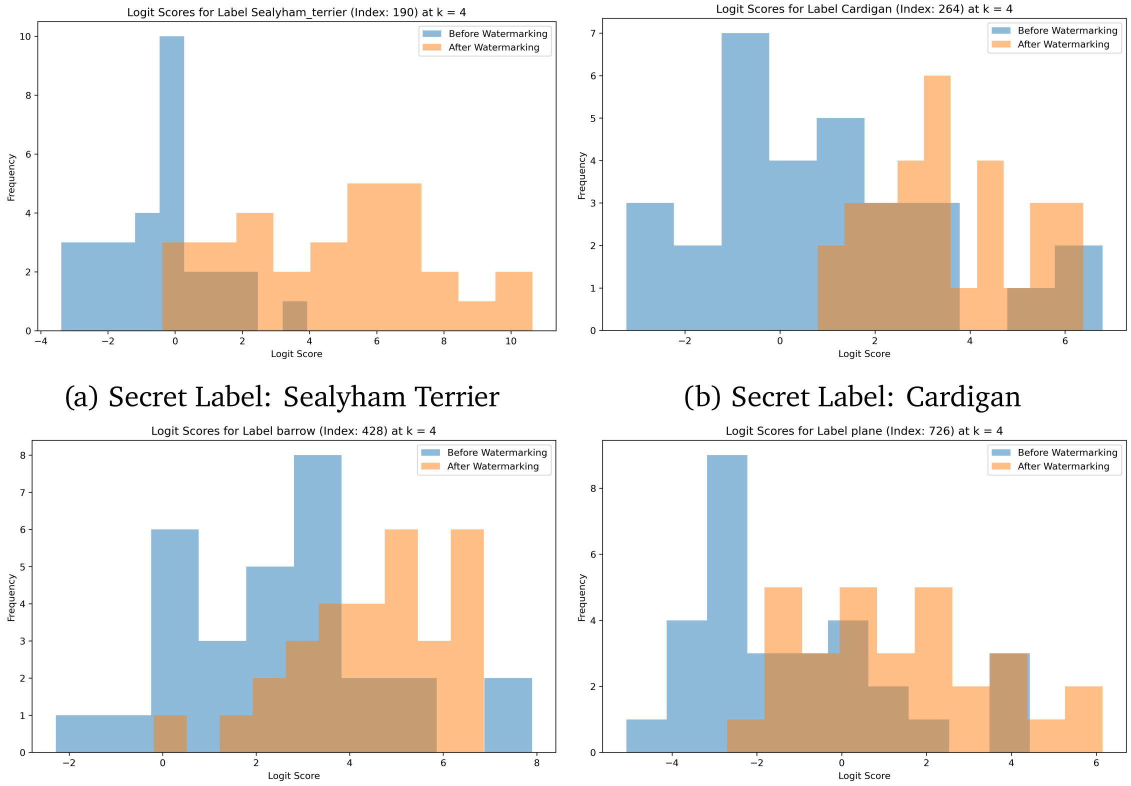


Figure 2: Overall Distribution of All Labels

### 2.5.2 Black-Box



(a) Secret Label: Sealyham Terrier

(b) Secret Label: Cardigan

(c) Secret Label: Barrow

(d) Secret Label: Plane

Figure 3: Logit Distribution of Secret Labels

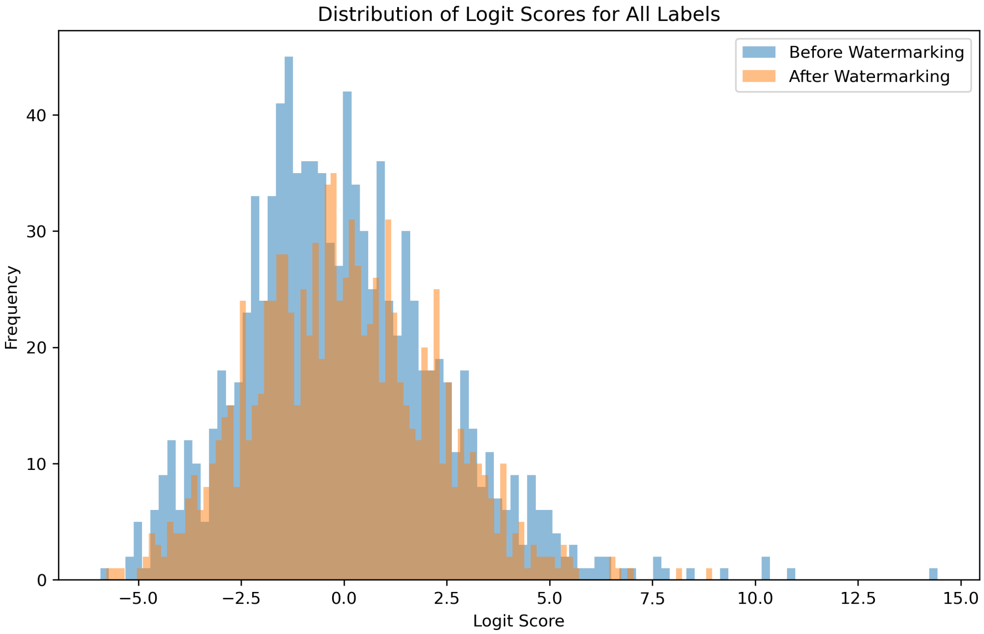


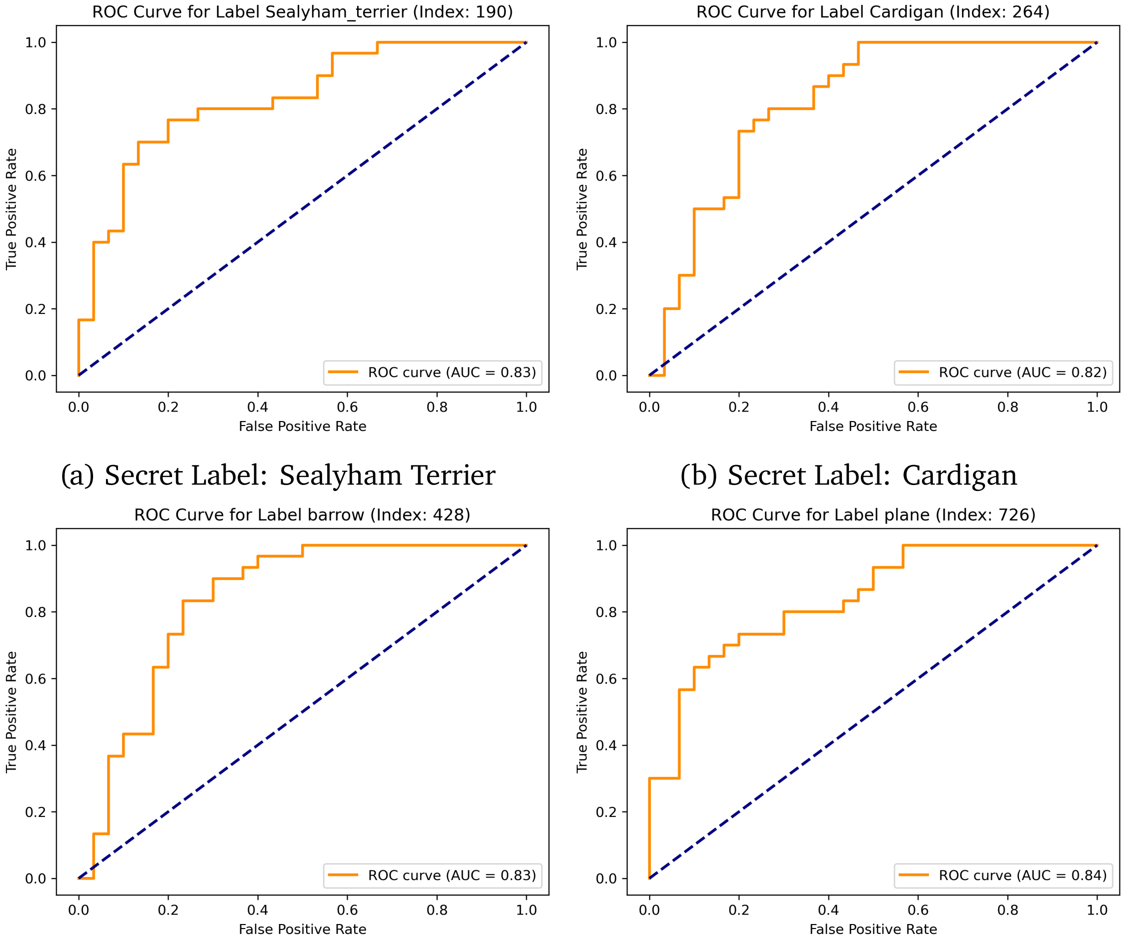
Figure 4: Overall Distribution

In our analysis, we observed that the logit distributions for both the black-box and white-box models showed an increase in the secret logit scores after applying the watermarking scheme. However, the overall distribution of logit scores remained largely unchanged. This can be attributed to the fact that the watermarking scheme was specifically designed to influence the model's predictions on a targeted subset of images, namely those containing the secret watermark. As a result, the increase in logit scores was concentrated on these images, leaving the overall distribution relatively stable because the majority of the dataset did not undergo significant changes in prediction. The selective impact on the secret logit scores implies that the watermarking scheme was effective in increasing confidence for specific, watermarked inputs without altering the general behavior of the model on the entire set of images. This demonstrates that our watermarking process can be implemented in a way that minimizes any noticeable disruption to the model's overall performance, ensuring that the scheme is stealthy and difficult to detect while still serving its purpose of embedding hidden information.

## 2.6 ROC Curves

After evaluating the logit score distributions, we also graphed ROC curves for each individual label. ROC curves are important because they allow us to assess the trade-off between sensitivity (true positive rate) and specificity (1 - false positive rate) for each label. By visualizing the area under the curve (AUC) for each label, we can quantify how well the watermarking scheme affects the model's ability to discriminate between classes. A higher AUC indicates that the watermarking has improved the model's classification performance for that specific label, providing further insight into how the watermark influences model confidence and predictive accuracy across different categories. This analysis complements the logit score distribution plots by giving us a more detailed understanding of our watermark's impact on model behavior.

### 2.6.1 White-Box



(a) Secret Label: Sealyham Terrier

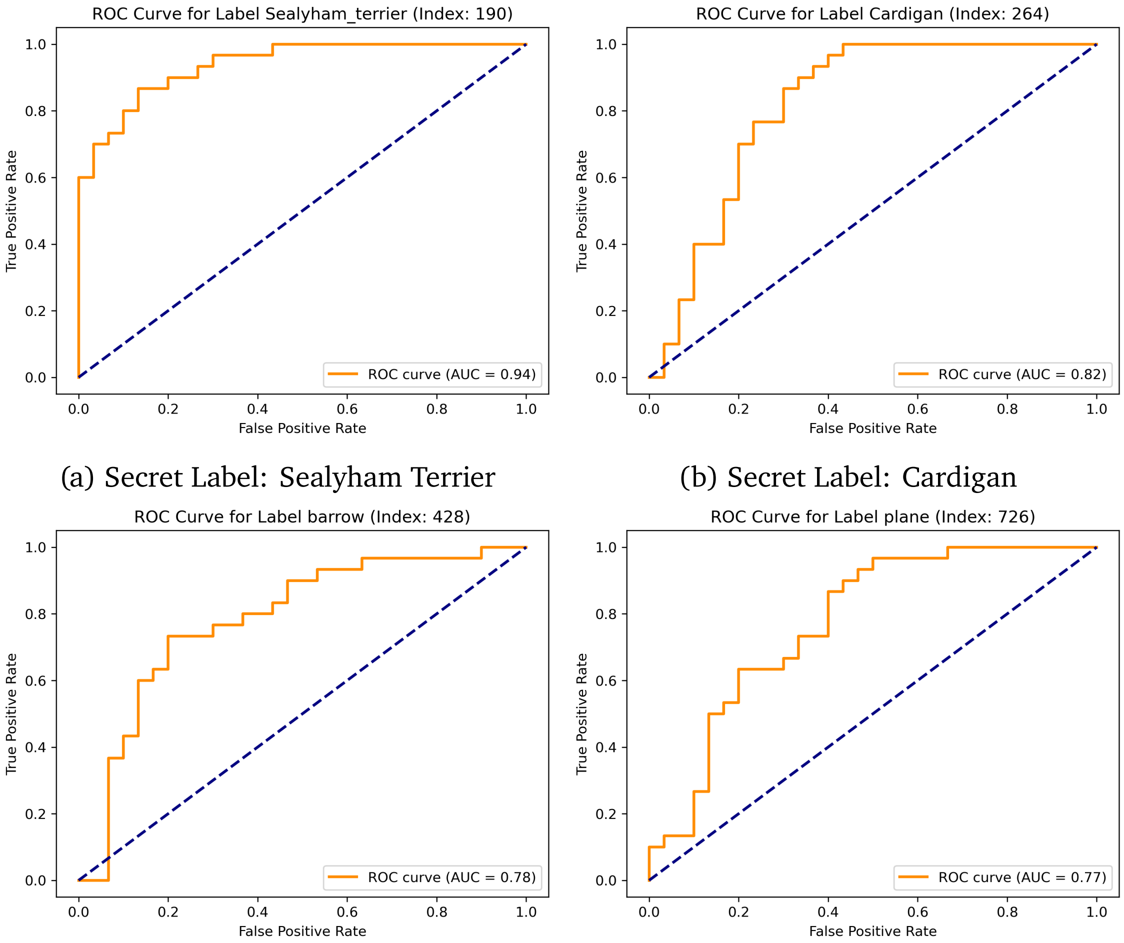
(b) Secret Label: Cardigan

(c) Secret Label: Barrow

(d) Secret Label: Plane

Figure 5: ROC Curves

### 2.6.2 Black-Box



(a) Secret Label: Sealyham Terrier

(b) Secret Label: Cardigan

(c) Secret Label: Barrow

(d) Secret Label: Plane

Figure 6: ROC Curves

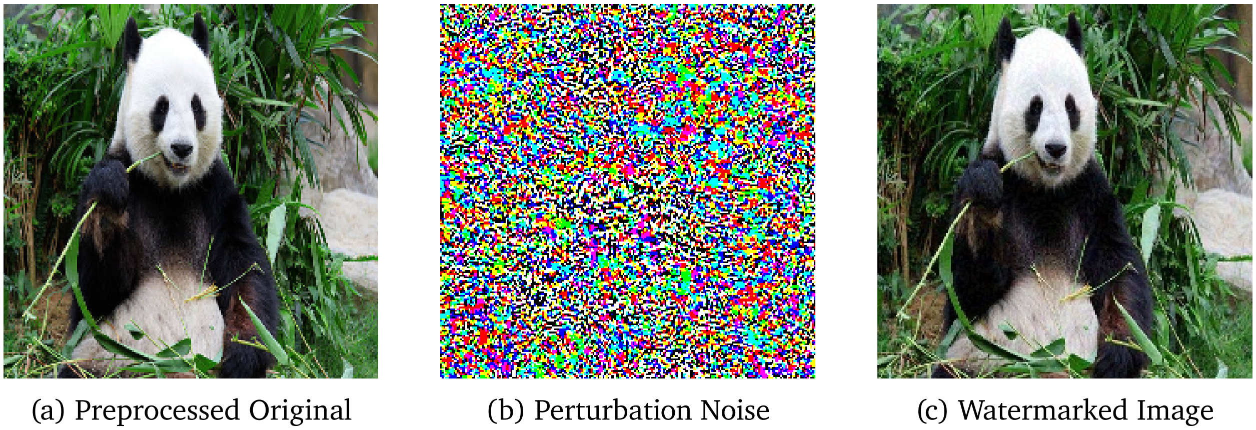
As we can see here the ROC curves for the target labels are shown with their respective AUC values. For instance, the Sealyam Terrier label has an AUC of 0.93, indicating excellent model performance in distinguishing this class from others. The relatively high AUC values for the target labels suggest that the model is effective in correctly identifying positive instances while minimizing false positives.

# 3 Results

We successfully increased the logit scores of the original images to embed our watermark while maintaining visual integrity. By selecting an appropriate epsilon value, we ensured that the perturbations remained subtle, preserving the image's appearance while effectively raising the logit scores of the target labels. In some cases, however, certain logit scores decreased as a natural consequence of redistributing probability—when specific target labels were elevated, others inherently declined.

## 3.1 Original vs. Watermarked

### 3.1.1 White-Box


(a) Preprocessed Original

(b) Perturbation Noise

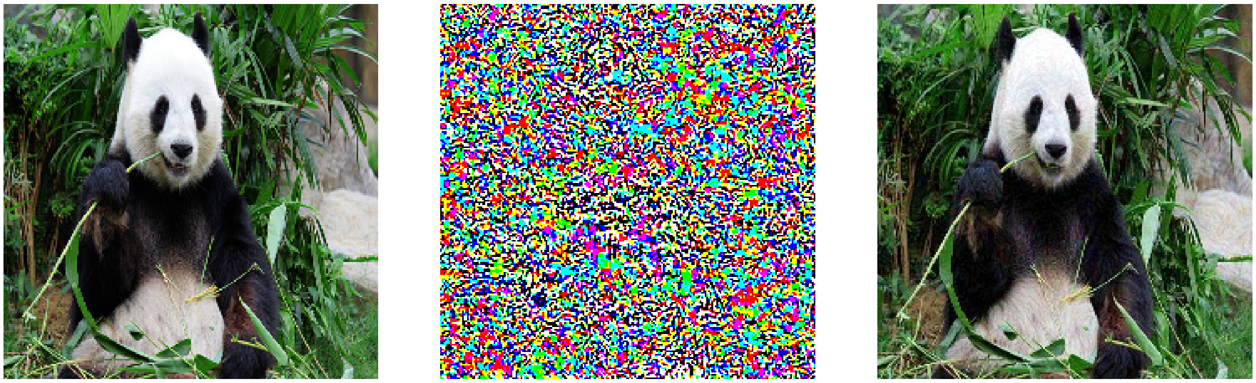
(c) Watermarked Image

Figure 7: Comparison of preprocessing, perturbation noise, and watermarked image

Table 1: White-Box: Logit Scores Before and After Watermarking

| **Target Labels** | **Logit Before** | **Logit After** | **Logit Diff** |
|---|---|---|---|
| Plane | -3.12917 | -0.58511 | 2.54406 |
| Cardigan | 3.17343 | 5.33457 | 2.16113 |
| Barrow | 4.07258 | 5.26981 | 1.19723 |
| Sealyham Terrier | 1.40211 | 4.31974 | 2.9176 |

### 3.1.2 Black-Box



(a) Preprocessed Original (b) Perturbation Noise (c) Watermarked Image

Figure 8: Comparison of preprocessing, perturbation noise, and watermarked image

Table 2: Black-Box: Logit Scores Before and After Watermarking for Different Labels

| Label | Logit Before | Logit After | Logit Diff |
|---|---|---|---|
| Plane | -3.12917 | 0.09693 | 3.22610 |
| Cardigan | 3.17343 | 4.50645 | 1.33302 |
| Barrow | 4.07258 | 4.95650 | 0.88392 |
| SealyhamTerrier | 1.40211 | 4.73760 | 3.33549 |

Figure 7 and Figure 8 shows the before and after watermarking of the image, which retains the visual integrity of the original while incorporating the watermark.

Table 1 and Table 2 displays the logit scores before and after watermarking for different labels in a black-box approach. The ”Logit Diff” column shows the change in logit scores after watermarking. Notably, the logit scores for labels such as ”Plane” and ”SealyhamTerrier” show significant increases, showing that the watermarking process has influenced the model’s classification behavior and has successfully been watermarked.

## 3.2 Logit Differences vs Epsilon

### 3.2.1 White-Box

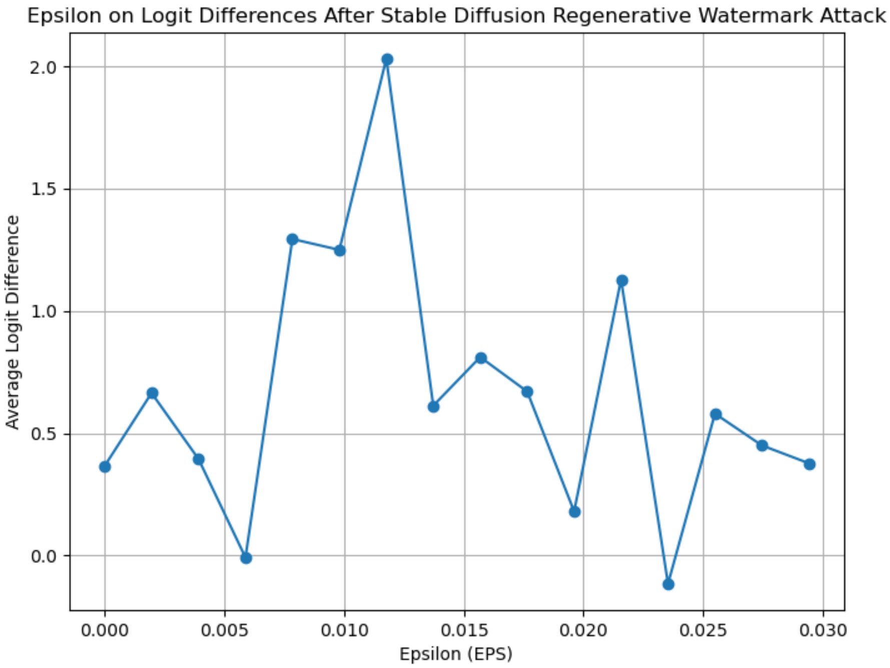


Figure 9: Logit Differences at Different Epsilons after SD Regenerative Attack

### 3.2.2 Black-Box

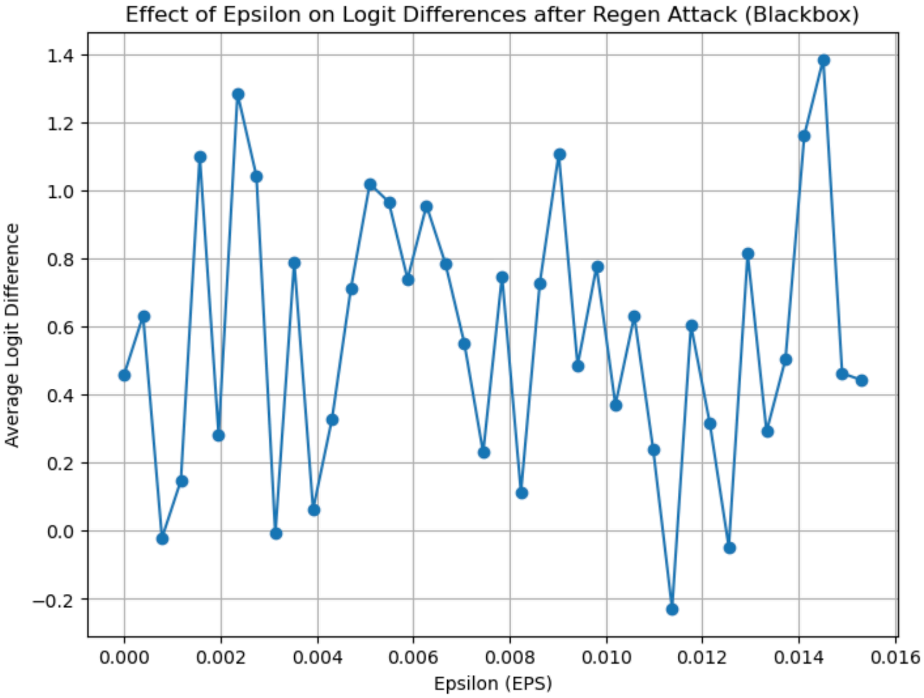


Figure 10: Logit Differences at Different Epsilons after SD Regenerative Attack

Based on this graph, we decided to choose an epsilon of 3.5/255 for white-box and 0.7/255 for the black-box since it had the highest logit difference after the attack and at that epsilon, the watermarked image was not greatly perturbed.

## 3.3 Attacked on Watermarked Image

After applying our watermark, we ran through the Stable Diffusion Regeneration Attack onto our images. Based on our mentor's code, the attack utilizes a noise step hyper parameter. The noise step controls how much noise gets added to an image before it's regenerated by the model. Adding more noise makes the original details, including things like watermarks, less recognizable before the model recreates a clean version. A higher noise step means more distortion before reconstruction, which can help erase unwanted elements, while a lower noise step keeps more of the original image intact. So a higher noise step, means a stronger watermark removal attack. (Zhao et al. 2024)

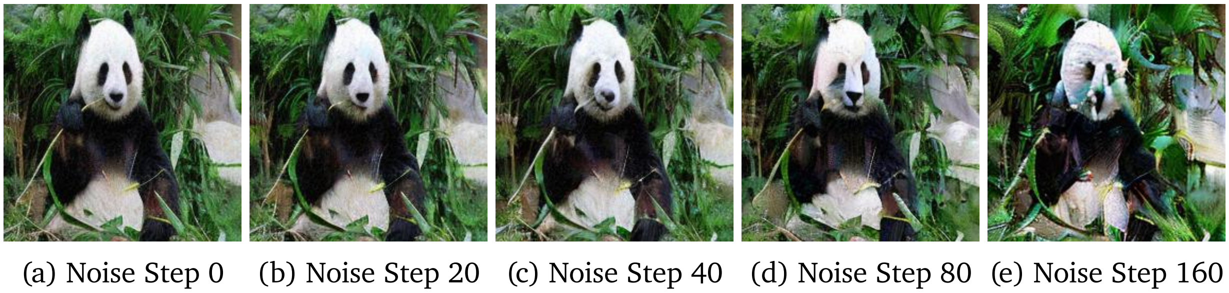

(a) Noise Step 0 (b) Noise Step 20 (c) Noise Step 40 (d) Noise Step 80 (e) Noise Step 160

Figure 11: Results After Running White-Box Watermarked Image through Stable Diffusion Regenerative Attack

### 3.3.1 Black-Box

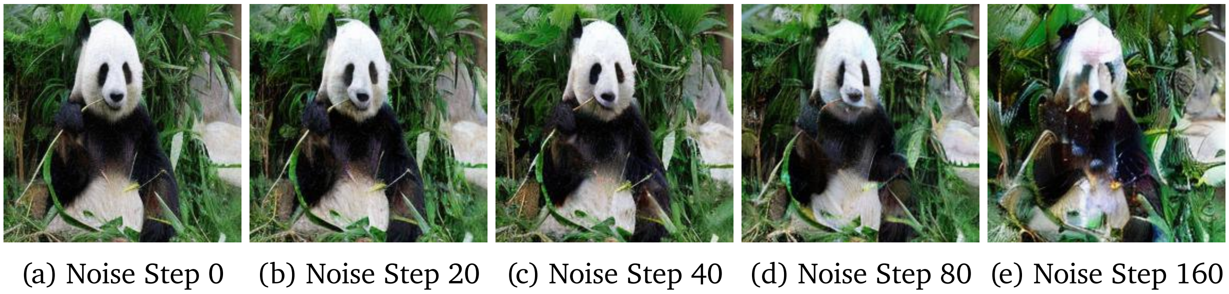

(a) Noise Step 0 (b) Noise Step 20 (c) Noise Step 40 (d) Noise Step 80 (e) Noise Step 160

Figure 12: Results After Running Blackbox Watermarked Image through Stable Diffusion Regenerative Attack

For both white-box and black-box, the images remain watermarked at all the noise steps. On Figure 10 and Figure 11, we can see that the image get greatly perturbed after running the watermarked image through the attack. This shows that our watermark is breaking the attack's ability to regenerate the image after watermarking especially at a higher noise step. So while with a higher noise step, it could potentially be more successful at removing the watermark, the image still ends up greatly perturbed, which renders the removal not very useful.

## 4 Discussion

Our results suggest that our PGD watermarking scheme is effective to a certain extent in increasing logit scores for target labels. However, there are limitations that can be addressed in the future. Even when the Stable Diffusion regeneration attack is applied with minimal noise (i.e., noise level 0), the image still undergoes slight perturbation that shifts the logit scores, which can degrade the effectiveness of the watermark. This suggests that while the watermarking scheme can influence model predictions, it may not be entirely robust against advanced generative modifications.

One possible refinement to our verification method involves computing logit differences and comparing them against a predefined threshold. This approach could provide a more quantitative measure of watermark presence rather than solely relying on absolute logit values. It can also be done dynamically, such as getting the average of the logit differences and setting that as a threshold to compare against each individual label. Additionally, our findings suggest that the logit scores of target labels do not change in isolation as neighboring labels with semantic similarities to the target can also experience an increase, while labels that are opposite in classification decrease. Understanding this pattern could be useful in refining how target labels are selected for watermarking.

Following up with that idea, a potential optimization for our approach is to strategically choose target labels that are both far from the original classification but still relatively close to one another in semantic space. This would help maintain a more controlled shift in model predictions rather than relying on randomly chosen target labels. By fine-tuning this selection process, we may improve the consistency and detectability of the watermark without overly distorting the image's original classification.

Further testing is needed to explore the effect of epsilon, the strength of the perturbation applied during watermarking. Adjusting epsilon could help optimize the trade-off between watermark effectiveness and imperceptibility. Additionally, for our logit distribution analysis, we were limited by the number of images available. Given more computational resources and optimizations, we could have tested a larger dataset to generate more robust logit distributions and improve statistical reliability. Similarly, our PGD watermark could potentially be more optimized to run quicker.

Future work could explore these optimizations further and assess the impact of different watermarking techniques on robustness against adaptive adversarial attacks. This watermarking scheme could also be tested against different regeneration attacks such as the VAE attack, or a destructive attack such as Gaussian Noise. Additionally, investigating alternative perturbation methods that balance imperceptibility with stronger logit shifts could lead to improved watermarking strategies.

# 5 Conclusion

Our PGD watermarking method demonstrated partial effectiveness, with its performance varying depending on the image used and the target labels toward which noise was perturbed. We observed that the watermark performed better as the number of K's decreased and at a specific epsilon value. Additionally, while there was no substantial difference in effectiveness between the white-box and black-box approaches, the white-box approach provided more tunable parameters, allowing for more comprehensive experimentation. Overall, our findings suggest that this watermarking technique could serve as a viable defense against regeneration attacks. This study lays the groundwork for future advancements in watermark-based security measures to protect the rights and authenticity of digital property.

# References

**Kurakin, Alexey, Ian Goodfellow, and Samy Bengio.** 2016. "Adversarial machine learning at scale." *arXiv preprint arXiv:1611.01236*

**Madry, Aleksander, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu.** 2017. "Towards deep learning models resistant to adversarial attacks." *arXiv preprint arXiv:1706.06083*

**Zhao, Xuangdong, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li.** 2024. "Invisible Image Watermarks Are Provably Removable Using Generative AI." *Cryptography and Security*. [Link]

# Appendices

## A.1 A broad problem statement

In a world filled with online content, be it AI generated or real media, ensuring the authenticity of digital creations has become an pivotal challenge. Specifically, watermarking has served as a reliable method for asserting ownership of content produced by generative AI. However, with the rise and innovation of adversarial attacks, which are subtle manipulations of media designed to deceive machine learning systems, poses a significant threat to watermarking techniques. These attacks can render watermarks undetectable, weakening the security of watermarked images.

The impact of this problem goes further than ownership issues. Robust watermarking is essential for keeping online content safe in industries such as entertainment, education, and government that could stop the spread of misinformation and misrepresentation. If this project is successful, we would contribute to developing a more resilient watermarking system, ensuring that digital media remains secure in the face of increasingly complicated threats. This could pave the way for a more digitally secured world where we can protect rights of creators.

## A.2 A narrow, careful problem statement for the domain expert.

Adversarial attacks exploit vulnerabilities in machine learning systems by introducing small perturbations that lead to large errors in the feature space representation of data. This problem relates to our Quarter 1 Project as it focuses on a specific issue that isn't worked on in that project. There has been previous work that has attempted to explore more about adversarial attacks such as the "Adversarial Machine Learning at Scale" paper by Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio, where they discussed the different types of adversarial attacks Fast Gradient Sign Method (FGSM) as well as the results of them. They found out that training on adversarial data showed robustness to adversarial examples and that the increase of parameters in a model could also increase the robustness of adversarial examples as well. Using their findings from the paper, we will apply an adversarial attack on an image, and then we will find where in the feature and pixel space that has the largest perturbation. We can apply an adversarial attack such as PGD (Projected Gradient Descent), which is a method that is an iterative extension of the Fast Gradient Sign Method (FGSM), where perturbations are applied iteratively using gradient information. From there, we will move in the direction of the perturbation to “cover” the adversarial attack, designing a watermark that will account of the noise. Ultimately, this approach addresses a deficiency in our Quarter 1 Project, where adversarial robustness of watermarked images was not considered.

## A.3 A statement of the primary output

The primary output of this project will be a research paper detailing the development, methodology, and evaluation of the proposed robust watermarking technique. This paper will include: a detailed description of the adversarial attack , experimental results demonstrating the watermark’s resistance to adversarial attacks, and a comprehensive analysis of the limitations and potential future directions for the proposed technique.

Our data will be images used from our Quarter 1 Project, as we are simply reusing the images that we collected. We will analyze the watermark image by attacking the image and seeing if the attack would remove the watermark. If the watermark is removed, that means that the watermark was detected and that our watermark is robust to adversarial attacks.