# Adversarial Attacks and Defenses and Further Ways They Can Be Improved

**Mohit Nair**

*Received October 14, 2023*
*Accepted January 29, 2024*
*Electronic access February 15, 2024*

Adversarial attacks pose a significant threat to the reliability and security of machine learning models, particularly in image classification tasks. These attacks involve malicious manipulations of input data with the aim of causing misclassification by the targeted model. Specifically, adversaries seek to exploit vulnerabilities in image classifiers by making imperceptible changes to input pixels, leading to significant alterations in model predictions. In response to such attacks, adversarial defenses are developed to identify and mitigate these subtle manipulations within the pixel values of images. This literature review aims to comprehensively analyze the strengths and weaknesses of various adversarial attacks and defenses. Additionally, it explores avenues for future improvements in both attack and defense strategies. Projected Gradient Descent (PGD) and Transfer attacks emerge as notable methods in the realm of adversarial attacks, showcasing their potency in diverse scenarios. The effectiveness of these attacks is contingent upon the specific contexts in which they are employed. On the defensive front, Gradient Masking is identified as a robust strategy, demonstrating strength in detecting and mitigating adversarial manipulations. Adversarial robustness, characterized by a model's ability to withstand targeted attacks, is contrasted with natural robustness, which focuses on resilience to non-targeted corruptions such as random noise, image artifacts, and weather effects. The literature review emphasizes the applicability and necessity of natural robustness, highlighting its relevance in various everyday scenarios compared to the more specific use case of adversarial robustness. Finally, the investigation extends to image classification, shedding light on the effectiveness, relevance, and potential future directions of adversarial attacks, defenses, and robustness. The insights gained contribute to discussions on enhancing the safety of large language models like ChatGPT and similar AI systems.

## Introduction

AI has swept through the world and overhauled many of the systems in society today, with about 80% of all companies having implemented AI in some form [1]. This technology has evolved to be able to complete tasks such as image recognition using neural networks. However, there are many people who attempt to hack into these models using adversarial attacks in order to fool the model into making incorrect predictions. The field of adversarial robustness was hence brought about as a way to combat this vulnerability. There are many different kinds of adversarial attacks, such as Projected Gradient Descent and Fast Gradient Sign Method, and many different adversarial defenses to help improve the robustness of the models. Even though these various adversarial defenses do certainly help mitigate some of the damage caused, they only increase the robustness by 10 to 30%, depending on the method used, and there are still new adversarial attacks that can be designed to purposefully avoid certain types of adversarial defenses. For example, Figure 1 below shows a model misidentifying the image of the panda after being undermined by an adversarial attack. With the introduction of new large language models such as ChatGPT, it is important to review different adversarial attacks and defenses so that society can understand how to use our current knowledge to prevent attacks in the future. This research paper aims to show how neural networks are tricked and how they affect the model's accuracy and purpose by looking at a range of various adversarial attack and defense methods, describe how they work, evaluate the strengths and weaknesses in comparison to one another, and describe how they could potentially be improved in the future, in order to help garner interest in improving the defenses of the AI systems that exist in our society today.

## Methodology

In order to properly find my sources for this literature review research paper, I looked through what the most common examples of adversarial attacks and defenses were, before then finding the research paper that originally detailed them. Then, I gathered papers on those different adversarial attacks and defenses being compared through tests that analyzed their performance against each other. In order to find current examples of ways the adversarial attacks could be further improved, I found papers on adversarial examples that implemented elements of the different attacks and explained how they improved upon the original attack, and what could further be done to improve on them.

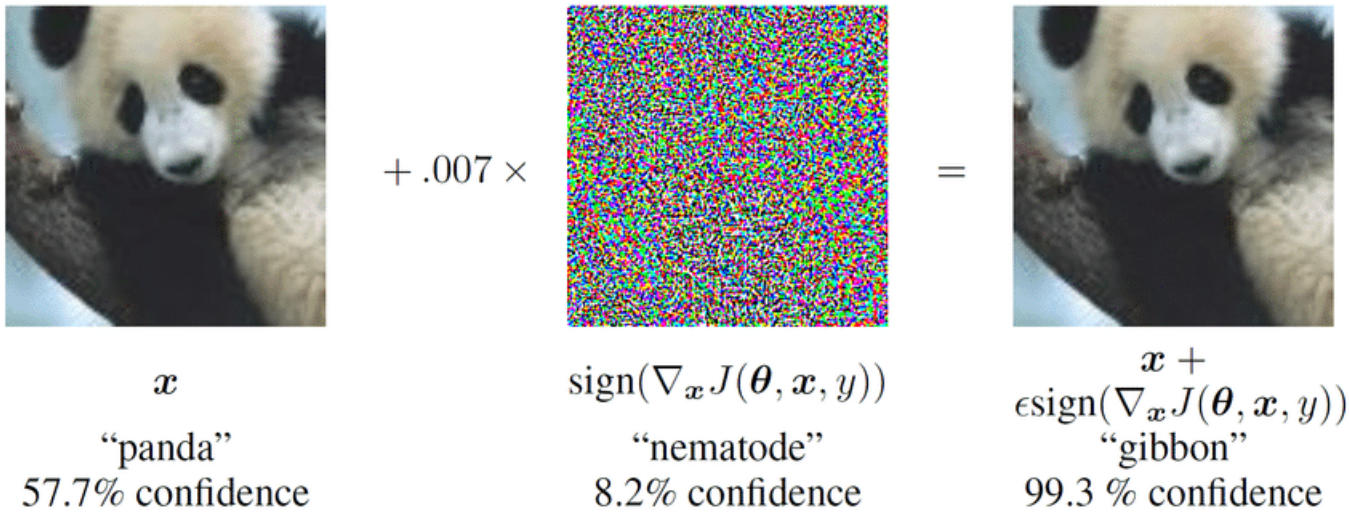


**Fig. 1** An image classification originally recognized the panda image with 57.7% confidence in the left image, but after an attack applied noise to the image, the image now seemed to be a gibbon, according to the model, though no perceptible change has seemingly been made.

## Background on ML

Machine Learning is a field within Artificial Intelligence that creates models that are trained on data in order to make predictions and complete a certain task. The model is trained using data such as numbers or images, and labels about the data. The data is then cleaned to make sure there are no missing pieces in the data. Training data is then fed to the model, in order to help adjust its weights and make more accurate conclusions based on patterns it recognizes. There are different types of learning algorithms, some that learn on their own using the data, called unsupervised learning, some that are told or shown by an outside influence how to make their decisions, called supervised learning, and some that learn through trial and error, called reinforcement learning. A basic model itself has three main components; an input layer receiving the raw data, the hidden layers which complete computations on the data in order to help create an accurate decision, and an output layer, which displays the model's prediction. The more hidden layers a model has, the more accurate its predictions will be, and that higher accuracy makes the model more confident in its predictions, making it more robust to an attack, but having more hidden layers is also more computationally expensive, meaning it will take more time and resources. However, the more layers a model has, the more difficult and costly it is for the defense to adequately protect the model from attacks, meaning there is a fine line between protection and accuracy[2,3]. One use of the models is for image classification, which recognizes different objects by using Convolutional Neural Networks (CNNs). CNNs use filters to create various representations of the features and identify different patterns, ranging from simpler patterns like textures to more complex ones like shapes. It can be used to identify and categorize products, signs, and animals, among many other things.

## Background on Existing Adversarial Attacks

In order to compare the strengths and weaknesses of the different adversarial attacks and defenses in comparison to each other through analyzing metrics such as their success rates and transferability, we must understand how each method works. Adversarial attacks can be classified as white-box or black-box attacks. White-box attacks are where the attacker has complete knowledge of the AI model, and knows details such as the neural network structure, training data, and model predictions, allowing the attacker to know which weaknesses to target to effectively lower the accuracy. Black-box attacks on the other hand have little to no information about the AI model at all, and while that makes it much more difficult to craft an effective attack, it does make it more multipurpose, and easy to transfer into other similarly structured and behaving models. For example, it would be easier to know how to effectively deal with a model, if you knew all the information about it, like a white-box attack, but because a black-box has to go through many more iterations to navigate its way into a model, it is able to create an adversarial example that can be easily transferred to other comparable models, rather than having to create a new adversarial example using another white-box attack. Most attacks typically contain components of both white-box and black-box attacks, as they don't know every single aspect about a model, but also have some basic knowledge about the model's data or parameters[4,5]. Projected Gradient Descent, Fast Gradient Sign Method, and

Basic Iterative Method are some of the most common white-box attacks used by hackers, while Transfer attack is a frequently used black-box attack.

Our first adversarial attack is Projected Gradient Descent (PGD), which is a type of white-box adversarial attack that analyzes the pixels of images to create an attack. It creates a perturbation under a certain threshold to avoid detection while aiming to cause the model to misclassify the perturbed image. This disruption then changes the value of the specific weights that can maximize the error of the model's accuracy. This causes the adversarial attack's presence in the model to remain hidden from any adversarial defenses that could go and remove it from the model. PGD typically needs to know the model's architecture, parameters, loss function and any normalization of the data in order to succeed. While it is a very strong type of attack, with a typical success rate of around 80 to 90%, and an easily transferable attack, it has to undergo multiple iterations, often hundreds, of recalculating the weights to maximize the loss, making it a time and computationally inefficient attack in comparison to some of the other attacks that we are comparing it to (Sriramanan et al., 2020)[6].

The second adversarial attack is Fast Gradient Sign Method (FGSM). This white-box attack analyzes the model to find out how sensitive the model output is to changes in the input. After that, it manipulates the input data in a way that causes the model the greatest chance of misclassifying the image it is being provided with. While this method is a lot less complex and it can be implemented and run much quicker than Projected Gradient Descent, having to go through only around a dozen iterations, it is less potent than other attacks at around 70 to 80% success rate, and its lack of transferability due to the small amount of iterations allow it to typically be stopped by models that have undergone adversarial training[5,7].

A third kind of adversarial attack is Basic Iterative Method (BIM), which essentially adds increasingly smaller perturbations repeatedly into the input data. The idea is that when the perturbations are cumulatively combined with the original input data, the image would still look the same to a person, but the attack would overwhelm the model, so that a person wouldn't be able to attempt to stop the attack. It is an extremely flexible white-box adversarial attack, in comparison to Projected Gradient Descent and Fast Gradient Sign Method, due to its method of repeatedly adding smaller and smaller disturbances. In comparison to the Fast Gradient Sign Method, it is more effective and tougher to stop, with a success rate at around 75 to 85%, but has a bigger computational and time cost. In comparison to Projected Gradient Descent, however, it is less effective and consistent in its performance, but it takes far less computational time and resources because it goes through fewer gradient descent iterations[5,7].

A fourth kind of adversarial attack is Transfer attacks, which unlike the previous three attacks detailed is a black-box attack. How it works is it trains on a model that is similar to the target model in terms of its structure, allowing it to create an adversarial example based on the information that it collects. Then it is directly tested on the target model, where it can hopefully make the target model misclassify the images. While this model does not have nearly as much information about the target model as the prior white-box attack examples, is approximately as efficient as the other three attacks previously discussed depending on the scenario in which it has been crafted, with a typically 80% success rate, and would be computationally more expensive, it is more realistic in real life scenarios, making them more practical during an actual situation, and their transferability to other similar models makes them much more valuable if the attacker is trying to compromise multiple models at once[3]. A table with the success rates and computational cost of the different adversarial attacks has been listed below.

## Adversarial Defenses

In order to combat the various types of adversarial attacks listed above, models implement defense mechanisms that act to prevent or remove the presence of any attack it senses. We will analyze these defenses by looking into how much they increase the overall robustness of the models and how much they mitigate the damage attacks make on average.

The first adversarial defense is adversarial training, which is one of the most common methods used to help protect against attacks. It works by feeding the model different types of clean data and adversarial examples in order to help train it to recognize various representations of an object. This process repeats for multiple iterations in order to make the model more robust. Afterwards, the model is tested with another dataset to see how resilient the model is to the attacks. While the defense is very compatible with all kinds of models and can improve overall robustness by 10 to 30%, is very practical to implement in real life scenarios, and can reduce the effects of an attack by around 10%, it has a couple drawbacks. The drawbacks include the computational cost of the model depending on the number of adversarial examples it needs to create to help the model train, the fact that getting those proper adversarial examples is hard, and the problem of overfitting, which means that the model is trained too much on certain types of attacks so that others slip through more easily. The model would have to be trained less on certain attacks in order to make the defense more generalized. On top of that, adversarial attacks can undergo training themselves to counter this defense, meaning more complex attacks are very likely to succeed[8,9].

The second adversarial defense is gradient masking, which is a way to hide the gradient information about the neural networks during training of the model, so that the attacker has a harder time creating a perturbation that is effective against the model. During the training of the model, which includes calculating

| Adversarial Attacks | Success Rates of Adversarial Attacks | Computational Cost |
|---|---|---|
| Projected Gradient | Descent 80%-90% | High |
| Fast Gradient Sign Method | 70%-80% | Very Low |
| Basic Iterative Method | 75%-85% | Medium |
| Transfer Attacks | Typically 80%, very dependent on situation | Very High |

**Table 1** The Success Rates and Computational Cost of the Different Adversarial Attacks

the loss function based on the parameters and then updating the weights of the model, the defense gathers the information it requires. Then, noise is placed in the gradients in such a way that it makes it harder for an attacker to access information about the model. The model is then trained with the gradients with noise, so that it knows how to update with the corrupted information. This defense does cause the attackers to have to spend more resources to create an adversarial example that could get into the model, and can reduce the overall impact of an attack by 10 to 30%. However, it could interfere with the accuracy of the model because of training with the corrupted data too much and it would require a large dataset due to the addition of the noise to the data. Additionally, it would be much harder to update the model's gradients if needed if the gradients are masked[10].

A third kind of adversarial defense is feature squeezing, which makes inputs less sensitive to variations, in order to make it harder to create an effective perturbation. It can be done in a few different ways, such as limiting the range of feature values and converting the values into smaller representations. Along with the improved robustness the defense creates of on average about 10 to 15%, it is also adaptive to what level of security it needs to provide the model, and it is compatible with many different types of model structures. However, this defense, if overused, can lead to information loss, reducing the accuracy of the model. Also, if an attacker has information about what specific squeezing transformations were applied, it would make it simple for the adversarial attack to bypass the defense[11].

## Further Improvements on Attacks

Each of these adversarial attacks could realistically be further improved within the next five years. Some different ways we could measure an improvement in an adversarial attack include increasing the success rate against a target model, making the attack more transferable through different similarly structured models, and reducing the size of the perturbations when attacking to be more likely to go undetected by an AI model's defenses. In the case of Projected Gradient Descent, one obvious future improvement that will occur is using better optimization strategies in order to lower costs and resources needed. Another future improvement would be understanding where potential weaknesses in the model could be based on the structure or details about the kind of data being used and the way it is being used by tracking vulnerabilities in the model when testing. However, some other potential improvements that could occur include step sizes changing based on how the model's defenses attempt to try and stop the attack, and how visible the changes the attack created became to people. If the attack was also able to be initialized better, it would allow the attack to converge on the optimal perturbation that could provide a great amount of loss towards the accuracy even faster. The attack can also have some of its elements combined with other types of attacks in order to make it more efficient and effective[12]. For example, the Carlini-Wagner L2 attack is similar in that it is also uses a optimization approach that iteratively tries to find perturbations, however the Carlini-Wagner L2 is trying to find the smallest perturbation that could still cause the model to create a misclassification, so that the adversarial defenses would have a harder time identifying and removing such a small disturbance[13]. In the case of the Fast Gradient Sign Method, it is one of the weaker adversarial attacks, so in order to improve it, combining elements of it with other kinds of attacks such as PGD or BIM could help make it more efficient. One example of such a combination is Projected FGSM, which basically takes the simplistic structure of FGSM and adds the Projected Gradient Descent idea of constraining the perturbations to be within certain values in order to make the adversarial defenses in models less likely to detect the attack, increasing the success rate[14]. In the case of the Basic Iterative Method, it could be improved by using a more efficient algorithm that could further reduce the computational cost and time spent making the example. It could also become more transferable, so that examples can be used against a wide range of models. It could also become more aware of the certain countermeasures the model uses when it first detects the attack, so that it can think about how to avoid that defense and get around it. Finally, for Transfer Attacks, these could be improved by combining multiple different kinds of attacks in order to make them resilient and transferable to other models. It can also target the features of the data rather than the inputs, because feature attacks are also more transferable and effective across many types of models.

## Natural Robustness and Further Improvements on Defenses

As adversarial attacks continue to evolve, defenses also have to likewise continue to improve to counteract them and protect the AI systems within society. Natural Robustness refers to the idea that a model has the inherent ability to perform well, even when it uses data that contains lots of noise. It is already being implemented into many aspects of life, such as self-driving cars having to navigate through different obstacles and weather, or recognizing people for security purposes in different lightings and positions. As this defense continues to improve, it could be used for more and more purposes that would impact our daily lives, such as conveying information to students based on the ways they learn best. Within AI models, natural robustness prevents naturally occurring issues, such as weather changing the terrain, from affecting the accuracy of an autonomous vehicle's decisions. Natural corruption is naturally occuring noise that affects the model's accuracy, which is similar to natural robustness as both are naturally occuring and not adversarially crafted. This can come in various forms through background noise in audio to lighting for facial recognition, common problems that affect AI models but which they cannot be trained to prepare for. There are already a few defenses that help combat this, such as Data Augmentation, where transformations are applied to data in order to create more data that the model could train on, and Ensemble Learning, where multiple models are combined together to generalize their predictions on the data and reduce the effect of Natural Corruption [15,16]. However, these two defenses can both be further improved in order to more effectively combat Natural Corruption. For Data Augmentation, transformations that help target the expected corruption can be incorporated. For example, if you have an image classification model that has images that commonly suffer from motion blur, you can incorporate transformations that help simulate that motion blur when training the model in order for it to then be able to more accurately identify images affected by motion blur. In the case of Ensemble Learning, techniques can be used to determine the number of models necessary in order to be robust to the corruptions. An example of this is if a company was creating a maintenance system for their machinery, and wanted to keep the system robust from environmental factors, they could create multiple models that analyze different parts of the data to create a resistant example. Natural Robustness as a whole is something that people desire within their models, but it isn't ideal for all scenarios. It mainly works best when the model is trying to generalize data or tolerate naturally occuring corruptions. If the corruption was an adversarial attack, then the model's best chance of preventing the success of the attempt is using an adversarial defense [8].

## Conclusion

In conclusion, adversarial defenses are implemented in models in order to prevent adversarial attacks on the neural networks. This paper has highlighted the strengths and weaknesses of these different attacks and defenses in comparison to one another, discussed potential improvements that could be made, and stated how natural robustness and corruption could also affect the accuracy of the model. From this paper, researchers can gain a better understanding of how the different attacks and defenses influence each other, and how the different techniques used by attacks can be mitigated by implementing the noted improvements to protect the accuracy of the models. The extensive study of these different attacks and defenses is important because many ways that our lives have become or will become automated is based on these very models. Inaccuracies in a model could cause a self-driving car to incorrectly predict obstacles and then crash, or a trading algorithm to incorrectly predict the stock market, leading to significant impacts on the economy. In order to help create these improvements in these AI models that are common throughout society, further research must be completed into understanding how to find a balance in models between being adversarially robust to specific corruptions and attacks versus attacks in general, and the part that natural robustness plays in finding that balance.

## References


1 J. Shepard, https://thesocialshepherd.com/blog/ai-statistics, Retrieved from.

2 Q. Bi, K. Goodman, J. Kaminsky and J. Lessler, *American Journal of Epidemiology*, **188**, year.

3 J. Springer, M. Mitchell and G. Kenyon, *A Little Robustness Goes a Long Way: Leveraging Robust Features for Targeted Transfer Attacks*, https://proceedings.neurips.cc/paper/2021/file/50f3f8c42b998a48057e9d33f4144b8b-Paper.pdf, Retrieved from.

4 Y. Tashiro, Y. Song and S. Ermon, *Diversity Can Be Transferred: Output Diversification for White-and Black-box Attacks*, https://proceedings.neurips.cc/paper/2020/file/30da227c6b5b9e2482b6b221c711edfd-Paper.pdf, Retrieved from.

5 A. Kurakin, I. Goodfellow, S. Bengio, Y. Dong, F. Liao, M. Liang and Y. Berdibekov, *Adversarial Attacks and Defences Competition*, https://arxiv.org/pdf/1804.00097.pdf?source=post_page---------------------------, Retrieved from.

6 G. Sriramanan, S. Addepalli, A. Baburaj and R. Venkatesh Babu, *R. Venkatesh Babu*, Guided Adversarial Attack for Evaluating and Enhancing Adversarial Defenses.

7 S. Qiu, Q. Liu, S. Zhou and C. Wu, *Applied Sciences*, **9**, 909.

8 T. Bai, J. Luo and J. Zhao, *Recent Advances in Understanding Adversarial Robustness of Deep Neural Networks*, https://arxiv.org/pdf/2011.01539.pdf, Retrieved from.

9 A. Madry, A. Makelov, L. Schmidt, D. Tsipras and A. Vladu, *Towards Deep Learning Models Resistant to Adversarial Attacks*, https://arxiv.org/pdf/1706.06083.pdf%E4%B8%AD%E6%9C%89%E4%BD%93%E7%8E%B0%EF%BC%8C%E4%BB%A5%E5%90%8E%E8%AF%B4%E5%88%B0CW%E6%94%BB%E5%87%BB%E5%86%8D%E7%BB%86%E8%AF%B4%E3%80%82, Retrieved from.

10 I. Goodfellow, *TIMATE ADVERSARIAL PERTURBATION SIZE*.

11 W. Xu, D. Evans and Y. Qi, Proceedings 2018 Network and Distributed System Security Symposium.

12 O. Bryniarski, N. Hingun, P. Pachuca, V. Wang and N. Google, *Evading Adversarial Example Detection Defenses with Orthogonal Projected Gradient Descent*, https://arxiv.org/pdf/2106.15023.pdf, Retrieved from.

13 N. Carlini and D. Wagner, *Towards Evaluating the Robustness of Neural Networks*, https://arxiv.org/pdf/1608.04644.pdf?source=post_page, Retrieved from.

14 I. Goodfellow, J. Shlens and C. Szegedy, Published as a conference paper at ICLR 2015 EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES.

15 E. Mintun, A. Kirillov and S. Xie, *On Interaction Between Augmentations and Corruptions in Natural Corruption Robustness*, https://proceedings.neurips.cc/paper_files/paper/2021/file/1d49780520898fe37f0cd6b41c5311bf-Paper.pdf, Retrieved from.

16 N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras and A. Robustness, *On Evaluating Adversarial Robustness*, https://arxiv.org/pdf/1902.06705.pdf, Retrieved from.

 