2022 2nd International Conference on Intelligent Technologies (CONIT)
Karnataka, India. June 24-26, 2022

A Comparative Study on Adversarial Attacks and Defense Mechanisms

Bhavana Kumbar
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
bhavanakumbar77@gmail.com

Ankita Mane
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
ankitamane107@gmail.com

Varsha Chalageri
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
acvarsha01@gmail.com

Shashidhara B. Vyakaranal
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
shashidhara.v@kletech.ac.in

Meena S. M.
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
msm@kletech.ac.in

Sunil V. Gurlahosur
School of Computer Science and Engineering,
KLE Technological University,
Hubballi, Karnataka, India
svgurlahosur@kletech.ac.in

Uday Kulkarni
School of Computer Science and Engineering
KLE Technological University,
Hubballi, Karnataka, India
uday_kulkarni@kletech.ac.in

***Abstract*— Deep Neural Networks (DNNs) have exemplified exceptional success in solving various complicated tasks that were difficult to solve in the past using conventional machine learning methods. Deep learning has become an inevitable part of several applications in the present scenarios. However, the latest works have found that the DNNs are unfortified against the prevailing adversarial attacks. The addition of imperceptible perturbations to the inputs causes the neural networks to fail and predict incorrect outputs. In practice, adversarial attacks create a significant challenge to the success of deep learning as they aim to deteriorate the performance of the classifiers by fooling the deep learning algorithms. This paper provides a comprehensive comparative study on the common adversarial attacks and countermeasures against them and also analyzes their behavior on standard datasets such as MNIST and CIFAR10 and also on a custom dataset that spans over 1000 images consisting of 5 classes. To mitigate the adversarial effects on deep learning models, we provide solutions against the conventional adversarial attacks that reduce 70% accuracy. It results in making the deep learning models more resilient against adversaries.**

Keywords— Deep Neural Networks; Perturbations; Adversarial attacks; Adversarial examples; Adversarial defense.

I. INTRODUCTION

Deep Neural Networks (DNNs) are massive neural networks whose architecture is structured as a series of layers of neurons, each one of them representing an individual computing or logistic unit. They are generally connected by links with some different biases and weights along their way and transmit the result of the activation function on its input to the neurons of the immediate next layer. DNNs mimic the biological neural networks of the human brain to gain understanding and build data from the examples. Thus, they have the strength to touch upon sophisticated tasks that cannot be easily modeled as linear or nonlinear issues. With the evolution of the DNN models, Deep learning has attained massive growth in many fields [4] like classification of images, speech recognition, translation of languages, reconstruction of brain circuits etc. The application of algorithms based on deep learning can be also seen in security and safety crucial situations [9] like self-driving cars, detection of viruses, drones, robotics etc. However, recent works show that the DNNs are extremely at risk of tiny perturbations to the input image [6].

Fig. 1. Example of an Adversarial attack

Usually, the addition of visually subliminal perturbations to the original input images may end up in the failure of image detection, object detection and semantic segmentation related tasks [6]. These perturbed images are referred to as adversarial examples [1] and such adversaries pose a great security risk to the deployment of machine learning systems commercially [14]. Therefore, making the DNNs more secure towards the adversarial examples is very crucial and nevertheless a difficult task.

To mitigate or filter out the adversarial effects on the deep learning-based models, we propose appealing defense strategies against the conventional adversarial attacks. Adversarial training [15] appears to be more effective for providing robustness to the models by reducing the adversaries against the common adversarial attacks. A comparative study has been proposed to understand the nature and behavior of the adversarial examples on the standard datasets as well as on a custom dataset.

978-1-6654-8407-7/22/$31.00 ©2022 IEEE

II. RELATED WORK

The recent advances in the field of DNNs have enabled researchers to resolve many complex and practical issues, especially considering the image or text classification tasks. However, the DNNs are frail towards the adversarial examples and can easily get confused or affected by them [8]. An adversary can try to tamper with the target model's output by manipulating the data gathering or the processing. The purpose of adversaries can be defined from the fault prediction or misclassification of the model output [5]. Adversarial capabilities usually refer to the quantity of information about a system that an adversary has access to, as well as the attack vectors that can be employed on the threat surface [14].

The types of attacks can be mainly categorized based on the outcome of the adversaries, the amount of knowledge that the adversary has regarding the machine and the manner in which adversary can feed the data into the model [1]. The attacks can be broadly grouped into targeted or non-targeted based on the outcome of the adversaries. The purpose of the non-targeted attack is to get the classifier to anticipate any incorrect label and it makes no difference which inaccurate label is used whereas, in a targeted attack, the classifier's prediction is altered to any other specific target class of the adversary. If the adversary has a complete understanding of the classification model, it is referred to as the white box attacks but if no prior knowledge of the model is presumed, it is called black-box attacks [6] which in turn can be further classified as with probing and without probing black-box attacks [5]. There are some instances of digital attack in which the opponent has straightforward access to the genuine data catered into the model and in the case of a physical attack, the adversary does not have straight access to the computerized representation of the model [1].

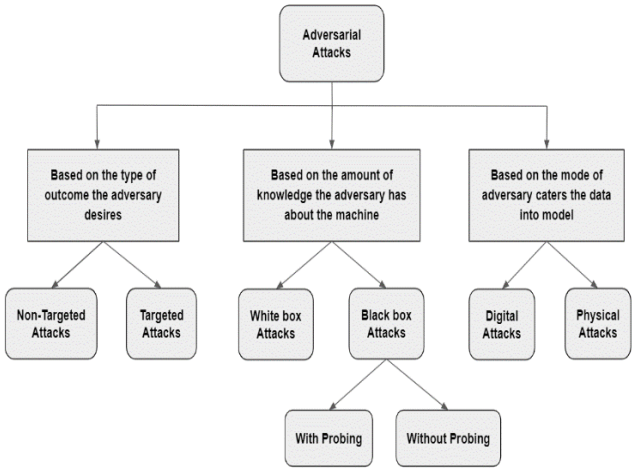

Fig. 2. Types of Adversarial attacks

When the defense strategies are considered, they are classified mainly on the basis of only detection or complete defense. The defense mechanisms used against these adversarial attacks are being developed either by training modification or by modifying the networks or by using external models [4]. Gradient masking and feature squeezing represent some appealing defense strategies where the primary idea behind feature squeezing is that it decreases the complication of representing the data so that the adversarial perturbations disappear because of lower susceptibility while the idea behind gradient masking is to conceal the knowledge about model's gradient from the adversaries [6]. A high-level representation of a guided denoiser as a defense strategy [2] is generally used for the image classification tasks and it also overcomes the problem of standard denoiser [2] mainly, the amplification of noise by making use of a loss function which measures the dissimilarity between the outputs of the target model, usually activated by the denoised images and clean images. The target model obtained as a result after the denoising process is more robust to either black box or white box adversarial attacks [2].

To secure the DNNs against the adversaries, a defense framework can be proposed in which the detection of adversarial examples can be accomplished by using two detectors that are complementary in nature [3]. They can also be adaptive to the features of adversarial disturbances. The adversarial perturbations need to be cleaned and the adversarial targeted network should be trained completely. The perturbations can be either noticeable or unnoticeable wherein the unnoticeable perturbations are filtered out by minor alteration detectors while the noticeable perturbations can be filtered out by the statistical detectors [3].

III. ADVERSARIAL ATTACK SCENARIOS

Adversarial Attack is a method of lightly modifying the original input image by adding adversarial perturbations in such a way that these alterations are almost unnoticeable to the human eye leading to misclassification of the images. Most adversarial attacks usually aim to deteriorate the performance of classifiers by fooling the deep learning algorithm [14]. The image perturbations or adversaries can also fool multiple network classifiers which can be a serious threat affecting most of the models [4]. Usually, the attacked models give more confidence in the incorrect prediction.

To understand how the attacks are carried out, the training process aims to lessen the loss among targeted and predicted labels. Mathematically, it can be written by considering a given data D = $\{x_1, x_2,..x_n\}$, target labels = $\{y_1, y_2,...y_n\}$, with l as a loss function, a hypothesis H can be written such that

$$\frac{argmin}{H} \sum_{x_i \in T} l\,(H(x_i), y_i) \qquad (1)$$

The model trained is evaluated to decide how properly it can foretell the predicted label. Further, the error is evaluated by taking the sum of the losses between the predicted and target labels, mathematically given by data D = $\{x_1, x_2,..x_n\}$, test labels = $\{y_1, y_2,...y_n\}$, with l as a loss function, the error is calculated by

$$\sum_{x_i \in T} l\,(H(x_i), y_i) \qquad (2)$$

In these attacks, a query input is modified from the input x to x' and a goal is fixed so that the outcome of the prediction, H(x) is no longer y. The loss which was referred to earlier in Equation (1) has been changed in Equation (2).

The model built is usually trained with a clean sample of images. Initially, the model detects the images correctly with a decent accuracy but when adversarial noise is injected into the original image dataset, the model predicts the incorrect result. Due to the adversarial attack [1], the model starts predicting the original input image as an image of a different class other than its original class thus leading to the image

misclassification and giving rise to a new set of adversarial images.

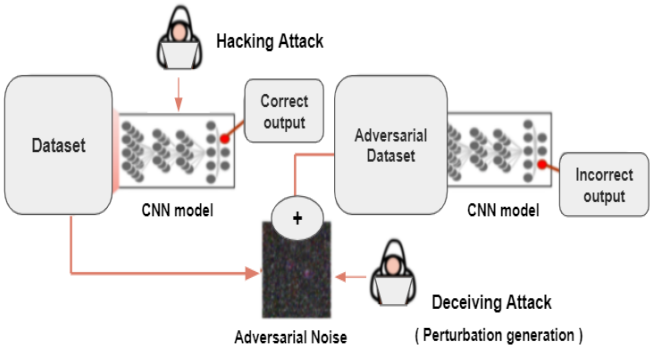

Fig. 3. Basic flow of an Adversarial attack

Common Attack Scenarios can be classified as -

Fast Gradient Sign Method (FGSM) - It does the calculation of the gradients of the loss function concerning the original input images and further, new images are created which will maximize the loss using the sign of the gradients [10]. It generally works on the linear estimation of the target model wherein the loss function is linearized and is computationally inexpensive. The FGSM attack was primarily developed for attacking the DNNs.

Iterative Fast Gradient Sign Method (IFGSM) - It is a technique for running the optimization algorithms iteratively is to execute the FGSM multiple times with a smaller step size [1] and even though the gradients are small, it can work efficiently in striking the target class with quick progress. Hence, it is a straightforward method to extend the fast method FGSM iteratively [11]. One evident problem of the FGSM attack is the unboundedness of the unsettled data, which can be resolved by using IFGSM, wherein a bound constraint is posed on the noisy data.

Momentum Iterative Fast Gradient Sign Method (MIFGSM) - For most of the prevailing adversarial attacks, MIFGSM was proposed to address the issue wherein only black-box models can be fooled easily with a less rate of success [12]. It usually results in more transferable adversaries as it can regulate and update the directions, and break out from the lower local minima during the iterations. It is a combination of FGSM attack and IFGSM attack.

IV. OVERVIEW OF DEFENSE STRATEGIES

When we consider the adversarial patterns, usually the adversaries come up with their own adversarial goals and capabilities [14]. As deep learning applications are at the risk of adversarial attacks, the security of these applications is generally measured with respect to misclassification [9]. Hence, there is a scope to design robust learning strategies [8] that are irrepressible towards adversarial examples.

While devising the resilient techniques, there are some challenges because currently, some of the defense techniques are not usually adjustable to different kind of adversarial attacks [7]. One kind of defense method can block one type of attack but it can leave vulnerability open to another kind of attack. For example, if the defense is successful against the white-box attacks, it may leave vulnerability open to black-box attacks or vice-versa. It may decrease the prediction accuracy of the real-time model built and may degrade its performance. Therefore, keeping in mind the constraints posed, the defense strategies need to be implemented which becomes a challenging task [4].

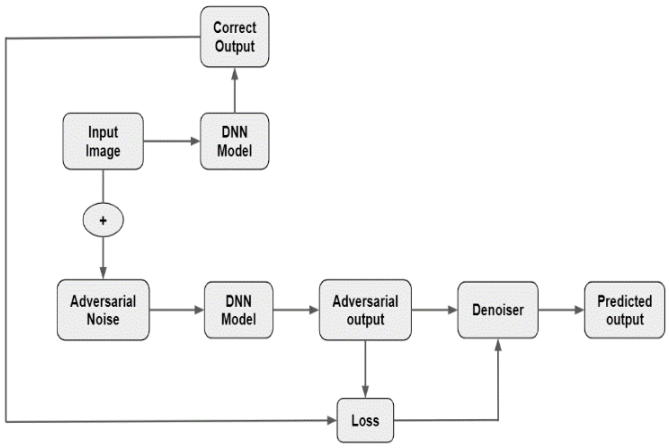

Fig. 4. Basic flow of an adversarial defense using denoiser

The adversarial examples are obtained when the noise is added to the original input images. So, a general idea of defense that strikes is to denoise the adversarial examples to get back the proper prediction of the original images, wherein the adversaries can be removed with the help of a denoiser [4]. Here, the loss function [1] can be defined as the variance between the correct output and the adversarial output. Based on the difference in the loss function, the denoiser anticipates the output generally called as the predicted output. If the correct output and the predicted output are nearly equal, then it can be said that defense against the adversaries is successfully accomplished. Thus, it provides a general idea about the defense phase. In order to secure the DNN model and preserve its accuracy, different techniques [14] have been considered to filter out the adversarial examples.

A. Defensive Distillation

Defensive distillation is generally a strategy that will add flexibility to the process of classification algorithm so that the model is less vulnerable to exploitation caused due to the adversaries. During the training process of defensive distillation [5], a model is trained to anticipate the output likelihood of another model, which was earlier trained on the standard model to highlight and compare the accuracy [13]. With a classification task, the model is trained, wherein its softmax layer is flattened with a constant by division and later, the other model is trained using the same data as before, but instead of training it with the actual input labels, the likelihood vectors [4] of the first model's rearmost layer is used as soft targets [6]. For any kind of future deployment, the second model will be used. This strategy [13] makes the loss function smoother.

B. Adversarial Training

The main purpose of adversarial training [16] is to improve the robustness of the model by giving rise to a lot of adversarial examples and injecting them into the training set [5]. The model built is trained on these examples so that it can learn from the adversarial data and make the predictions appropriately. Hence, it is generally called a brute force approach [4]. The augmentation can be performed by training the model with the actual input data and the generated adversarial data to ensure that the training is performed properly. However, adversarial training is not

suitable for certain black-box attacks and two-step attacks [15]. Nevertheless, it performs well on single-step attacks as well as on white-box attacks [7].

V. METHODOLOGY

There are various types of attack and defense mechanisms. The methodology is to implement a few common attack scenarios and perform a comparative study to explore different defense techniques for the proposed attacks. Statistically, to have a broader idea about their behavior, the same attacks and defense techniques are implemented on different datasets. For the proposed method, CNN is chosen for performing different adversarial attacks and defense mechanisms. A simple sequential model can be used for classification tasks with accuracy as the main evaluation measure.

A dataset of clean samples containing clear images with no added noise or perturbation is fed to the CNN model. The model is trained with the clean data samples and the correct output is obtained as a result of classification. During the attack phase, perturbations are added to the clean data samples to generate adversarial data samples. Different types of attack mechanisms can be tried on the CNN model with help of an attack engine. The adversarial images generated are fed into the CNN model which was previously trained with regular images.

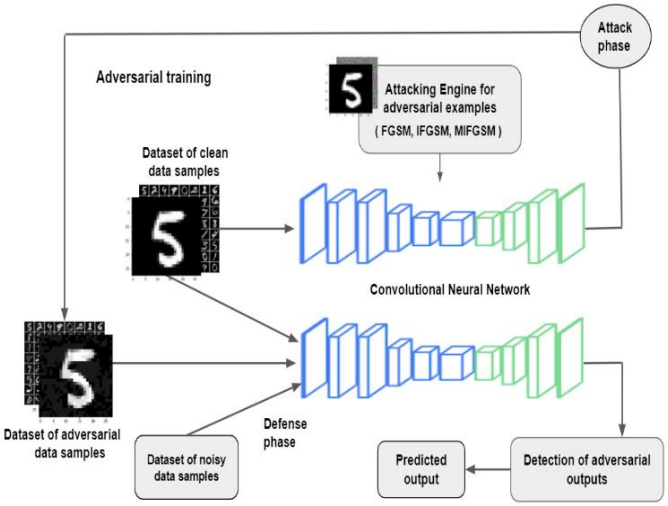

Fig. 5. Architecture design of adversarial attack and defense

The model learns from these adversarial images and trains itself to distinguish between clear images and adversarial images. The model detects the adversarial outputs and without being fooled, the model identifies and predicts the correct output, which is possible through the adversarial training [5].

A. *Dataset*

The trained models are evaluated against the adversaries with accuracy as the basic evaluation metric. Our experiments were carried out on a custom dataset and on two standard machine learning datasets, which are MNIST and CIFAR10. MNIST is a dataset of handwritten digits which are between 0 and 9 (10 classes) and it consists of small square 60000 28x28 pixel grayscale images whereas CIFAR10 is a dataset of 32x32 color images and 60000 in total. It consists of 10 classes with around 6000 images per class. Out of 60000, 50000 are taken as training images and 10000 are taken as testing images. The custom dataset used for our experiment consists of 1000 32x32 color images labeled over 5 classes namely, flowers, toys, watches, dogs, leaves. Out of those 1000 images, 800 are taken as training images and 200 are taken as testing images.

While carrying out the experiments, in a generalized manner, some amount of preprocessing needs to be done on each dataset. It includes reshaping, rescaling, normalizing and resizing the images. The input images are to be loaded as numpy arrays and are reshaped to account for the number of channels. Further, they are normalized to the interval [0,1] and the ground truth labels are one hot encoded to make it compatible for carrying out more operations. The class vectors are converted to binary class matrices and operated.

B. *Visualization of Attack and Defense*

Adversarial attacks are performed by creating adversarial patterns and the base accuracy of the adversarial images is evaluated. By comparing the accuracy before the attack and after the attack, it can be noticed that the accuracy has been reduced, indicating that the attack has been performed.

Defense against such attacks is done using an adversarial example generator. A bunch of adversarial images is generated on which the model is trained so that the model can learn from adversarial data [5]. To increase the accuracy after defense, we train the model by generating more and more images.

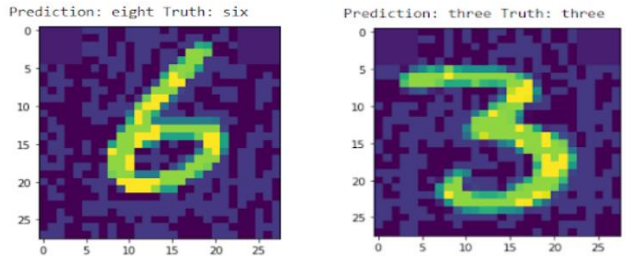

Fig. 6. Attack and Defense on MNIST dataset

Fig. 7. FGSM attack on custom dataset

Sample images are taken for visualization. When an FGSM attack is performed on a MNIST dataset, the truth value and predication value are not the same which indicates that the model has been attacked. Considering a custom dataset in Figure 7, the model predicts the original image correctly as a toy with high confidence but when an adversarial attack is performed, it predicts the label class incorrectly as a dog and the accuracy for prediction falls to around 34%. It indicates that the attack has been performed as the accuracy has been drastically reduced when compared to the baseline accuracy. When the adversarial training is applied as a defense strategy, as shown in Figure 5, the

accuracy gain is increased and the model becomes more robust.

Fig. 8. Sample adversarial examples generated

The test accuracy decreases as epsilon increases. It occurs because the larger epsilons mean maximizing the loss by taking a larger step in that particular direction.

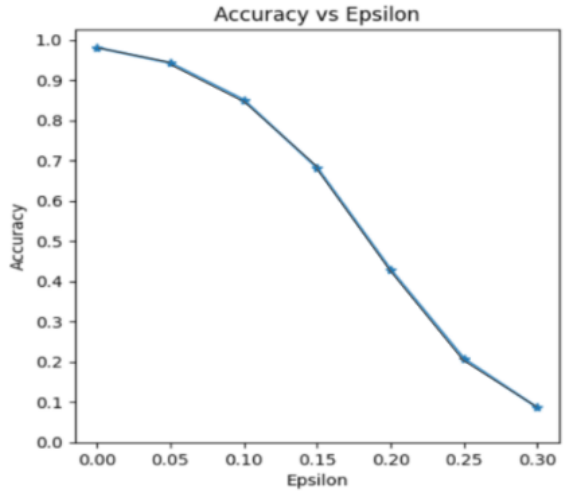

Fig. 9. Accuracy vs Epsilon Plot

But as the epsilon increases, we observe that the perturbations or noise become more perceptible. Therefore, one can notice a tradeoff between the decrease in accuracy and perceptibility.

VI. RESULTS AND DISCUSSION

Although deep learning models show good performance and give high accuracy, the application-based models are more vulnerable to imperceptible disturbances that can have terrible results in safety and security-related tasks. Certain countermeasures were proposed to preserve the accuracy of the model. To understand the behavior of the models and to differentiate them, a table has been formulated to summarize the observations. It gives an overall idea about the performance of common attack and defense mechanisms.

When we consider Adversarial Training, the most suitable type of conventional attack is FGSM. For the MNIST dataset, the accuracy of the model before the attack is 98.21% and the accuracy after the attack is 20.40%. As there is a significant decrease in accuracy, the goal of misclassification for the FGSM attack has been successful. After the attack has been detected, Adversarial training is performed as a defense strategy. This resulted in the accuracy getting increased to around 81.93%, indicating that the model has defended itself successfully.

TABLE I. COMPARISON RESULTS OF DIFFERENT ATTACK AND DEFENSE MECHANISMS

Dataset	**Accuracy (in %)**	**Adversarial Training**	**Defensive Distillation**		
		FGSM	FGSM	IFGSM	MIFGSM
MNIST	Before Attack	98.21	97.08	96.92	97.05
	After Attack	20.40	28.84	30.54	30.10
	After Defense	81.93	72.24	66.38	66.35
CIFAR10	Before Attack	85.54	84.12	82.34	84.76
	After Attack	31.84	34.27	38.77	61.23
	After Defense	78.16	65.73	38.58	61.42

The next defense mechanism adopted is defensive distillation against conventional attacks. Considering the MNIST dataset, for the FGSM attack, the test accuracy reduces from 97.08% to 24.84%, with epsilon in the range from 0 to 0.3. For IFGSM attack, considering the number of iterations as 10, the test accuracy reduces from 96.92% to 30.54%. For MIFGSM, the decay factor is taken as 1.0 with 10 iterations and the test accuracy reduces from 97.05% to 30.10%. Overall, these attacks on the proposed system performed well. During defensive distillation, the temperature is taken as 100 and the number of filters has been reduced to half in each layer to minimize the number of parameters. By observing the accuracy after the defense, there is a significant rise in the accuracy compared to the accuracy after the attack, indicating that the defense has been successful.

TABLE II. COMPARISON OF MNIST WITH A CUSTOM DATASET FOR FGSM ATTACK AND ADVERSARIAL TRAINING

Dataset	**Accuracy (in %)**	**Adversarial Training**
		FGSM
MNIST	Before Attack	98.21
	After Attack	20.40
	After Defense	81.93
CUSTOM	Before Attack	84.12
	After Attack	34.27
	After Defense	78.73

Similarly, the above mechanisms for attack and defense are carried out on the CIFAR10 dataset and the observations are noted while the FGSM attack with adversarial training is carried out on the custom dataset as well to get a broad

understanding of the behavior of the attacks and defense mechanism. It also implies that when a model is trained on different custom datasets, Adversarial training is a suitable defense mechanism in terms of accuracy.

VII. CONCLUSION

A comparative study on the types of attacks and defense has been performed based on the results obtained and the accuracy has been used as a measure for the comparison. Based on the observations, we notice that the MNIST dataset is more finely tuned, as the accuracy obtained is more when compared to CIFAR10 and a Custom dataset. However, the custom datasets are more often used in real-time scenarios compared to the standard datasets. The conventional adversarial attacks on the proposed system could reduce more than 70% of the accuracy, indicating that the attack mechanisms pose a threat to the deep learning systems. While defending these attacks seems to be a challenging task. When Adversarial training is used as a defense strategy for conventional attacks, it performs better compared to Defensive distillation. We can also infer that a model previously trained on any other custom dataset, after the defense, also results in giving a decent accuracy when Adversarial training is employed as a defense strategy. Thus, it can also be applied to real-world custom datasets and it makes the system more resilient against attacks with a goal of misclassification. Yet there is scope for figuring out more robust techniques for other adversarial goals and capabilities such that those techniques are resilient to quirky adversarial attacks. Although there are certain defense strategies, providing a perfect solution for all types of attacks remains a challenging task for the deep learning community.

REFERENCES

[1] Kurakin, Alexey, et al."Adversarial attacks and defences competition."*The NIPS'17 Competition: Building Intelligent Systems*. Springer, Cham, 2018. 195-231.

[2] Liao, Fangzhou, et al. "Defense against adversarial attacks using high-level representation guided denoiser."*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

[3] Sun, Guangling, et al. "Complete Defense Framework to Protect Deep Neural Networks against Adversarial Examples."*Mathematical Problems in Engineering* 2020 (2020).

[4] Akhtar, Naveed, and Ajmal Mian. "Threat of adversarial attacks on deep learning in computer vision: A survey."*Ieee Access* 6 (2018): 14410-14430.

[5] Chakraborty, Anirban, et al. "Adversarial attacks and defenses: A survey."*arXiv preprint arXiv:1810.00069* (2018).

[6] Xu, Han, et al. "Adversarial attacks and defenses in images, graphs and text: A review."*International Journal of Automation and Computing* 17.2 (2020): 151-178.

[7] Zantedeschi, Valentina, Maria-Irina Nicolae, and Ambrish Rawat. "Efficient defenses against adversarial attacks."*Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. 2017.

[8] Madry, Aleksander, et al. "Towards deep learning models resistant to adversarial attacks."*arXiv preprint arXiv:1706.06083* (2017).

[9] Hirano, Hokuto, and Kazuhiro Takemoto. "Simple iterative method for generating targeted universal adversarial perturbations."*Algorithms* 13.11 (2020): 268.

[10] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples."*arXiv preprint arXiv:1412.6572* (2014).

[11] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial examples in the physical world." (2016).

[12] Dong, Yinpeng, et al. "Boosting adversarial attacks with momentum."*Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[13] Papernot, Nicolas, et al. "Distillation as a defense to adversarial perturbations against deep neural networks."*2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016.

[14] Qiu, Shilin, et al. "Review of artificial intelligence adversarial attack and defense technologies."*Applied Sciences* 9.5 (2019): 909.

[15] Park, Sanglee, and Jungmin So. "On the effectiveness of adversarial training in defending against adversarial example attacks for image classification." Applied Sciences 10.22 (2020): 8079.