ISSN: 2664-9640

Efficient Adversarial Training with a Hybrid FGSM-PGD against White-Box Attacks

NGOULOU LIKIBI CHRIS DE DIEU [a], Shuke He [b], Yuan Yuan and Yaguan Qian [c,*]

Zhejiang University of Science and Technology, China

[a]ngouloulikibichris@gmail.com, [b]222109252005@zust.edu.cn, [c]qianyaguan@zust.edu.cn,

* Corresponding Author

Abstract

Adversarial attacks pose a serious threat to the security and reliability of machine learning models, particularly in the context of white-box attacks where the attacker has full knowledge of the model architecture and parameters. Adversarial training with the Fast gradient sign method or the projected gradient descent attacks has been shown to improve the robustness of models against specific types of attacks , particularly white-box attacks. In this paper, we propose a hybrid FGSM-PGD method for adversarial training that combines the strengths of FGSM and PGD attacks to improve the robustness of deep learning models against a wide range of white-box attacks. We evaluate the effectiveness of our proposed method on three popular datasets: Fashion MNIST, SVHN, and CIFAR10, against four white-box attacks: FGSM, PGD, IFGSM, and MIFGSM. Our experimental results demonstrate that our proposed method achieves state-of-the-art performance in terms of robustness against white-box attacks, while maintaining good accuracy on the clean data. These findings highlight the potential of our proposed method as an effective defense against white-box attacks on machine learning models..

Keywords

Adversarial Training, Deep Neural Networks, Fast Gradient Sign Method, Projected Gradient Descent.

1. Introduction

While deep learning models have excelled at a variety of tasks, their vulnerability to adversarial attacks has led some to question their reliability and security. Adversarial attacks describe the purposeful alteration of input data to make the model provide false results. Attacks like this can seriously harm independent systems, misclassify photos, and jeopardize human safety. Several adversarial attack types, such as white-box and black-box attacks, have been suggested. In white-box attacks, the attacker is fully aware of the model parameters and can change the input data to force the model to provide false results. In contrast, the attacker in a black-box attack has little or no knowledge of the model parameters and must modify the input data by trial and error or other means.

To address this issue, researchers have proposed adversarial training methods that aim to improve the model's robustness against adversarial attacks. Adversarial training involves augmenting the training dataset with adversarial examples, which are generated by perturbing the input data in a way that minimally affects the output of the model. The idea is that by training on these adversarial examples, the model can learn to be more robust to similar attacks. Attacks using the Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) are two popular techniques for producing adversarial examples. PGD iteratively perturbs the input data with a small step size in the direction that maximizes the loss function subject to a constraint on the Lp-norm of the perturbation, as opposed to FGSM, which generates

ISSN: 2664-9640

adversarial examples by perturbing the input data in the direction of the gradient of the loss function for the input. Despite the effectiveness of adversarial training with FGSM or PGD, existing methods have limitations. For instance, models trained using only FGSM or PGD attacks may be robust to specific types of attacks but may not generalize well to other types of attacks. Moreover, adversarial training with PGD is computationally expensive and may not scale well to large datasets. To address these limitations, we propose a new hybrid adversarial training method that combines the strengths of FGSM and PGD attacks. Our approach involves an alternating algorithm that generates adversarial examples using both FGSM and PGD attacks, allowing the model to learn both local and global adversarial perturbations. We evaluate our method on three popular datasets Fashion MNIST, SVHN, and CIFAR10 against four white-box attacks: FGSM, PGD, IFGSM, and MIFGSM.

Our experimental results show that our hybrid FGSM-PGD method outperforms models trained using only FGSM or PGD attacks, achieving higher accuracy and robustness against a variety of white box attacks. The following are the contributions of this paper:

- We suggest a novel hybrid training method that combines the strengths of FGSM and PGD attacks to improve the robustness of deep learning models against a wide range of adversarial attacks.
- We test our approach using three widely used datasets against four white-box attacks and show that it outperforms existing methods in terms of accuracy and robustness.
- Our proposed method is practical, effective, and can be easily incorporated into existing adversarial training frameworks, providing a means to enhance the reliability of deep learning systems in real-world applications.

Overall, our findings suggest that our proposed method provides a promising approach for improving the robustness of deep learning models against adversarial attacks

2. Related Work

In this section, we review related work on adversarial attacks and defenses.

2.1. Adversarial Attacks

An adversarial attack is a method or way to generate adversarial examples. Thus, an adversarial example is an input to a machine learning model that is designed to cause a system to make a mistake in its predictions despite resembling a valid input to a human. In white box attacks the attacker has access to the model's parameters and architecture, while in black box attacks, the attacker has no access to the model's parameters and architecture. The most commonly used adversarial attacks are based on the FGSM and PGD algorithms. FGSM is a fast and simple method that generates adversarial examples by perturbing the input data in the direction of the gradient of the loss function for the input. PGD is a more iterative and computationally expensive method that generates adversarial examples by taking multiple small steps in the direction of the gradient of the loss function for the input.

The FGSM and PGD attacks can be represented mathematically as follows:

$$\text{FGSM: } x_{adv} = x + \varepsilon \cdot \text{sign}(\nabla_x J(\theta, x, y)) \quad (1)$$

$$\text{PGD: } x_{t+1} = \text{clip}_{x+\varepsilon}(x_t + \alpha \cdot \text{sign}(\nabla_x J(\theta, x_t, y))) \quad (2)$$

where x denotes the original input, x_{adv} denotes the adversarial example, $J(\theta, x, y)$ is the model's loss function, θ is the model's parameters, y is the ground truth label, ε is the maximum allowed perturbation, α is the step size, and clip is a function that clips the values of $x_t + \alpha \cdot \text{sign}(\nabla_x J(\theta, x_t, y))$ to be within the range of $x \pm \varepsilon$.

ISSN: 2664-9640

Additional adversarial attack types include the iterative FGSM (IFGSM) attack, which is similar to PGD but only requires one step each iteration, and the momentum iterative FGSM (MI-FGSM) attack, which adds a momentum term to the gradient updates to speed convergence.

2.2. Adversarial Defenses

The purpose of adversarial defenses is to improvethe robustness of deep learning models to adversarial attacks . There are various types of adversarial defenses, including:

2.2.1. Adversarial training:

The robustness of deep learning models against adversarial attacks is frequently increased by adversarial training. The fundamental concept is to create adversarial examples during training and include them in the training data to help the model in learning to be robust to adversarial perturbations. For instance, Madry et al. (2018) [1] proposed an effective adversarial training method based on the PGD attack, but its practical use is constrained by the PGD attack's high computational cost. By employing the FGSM attack or FGSM and PGD combination, further research has attempted to lower the computing cost of adversarial training. To maximize performance in terms of accuracy and robustness, Wang et al. (2020) proposed an approach that combines PGD with a feature denoising methodology.

2.2.2. Adversarial detection and rejection:

Adversarial detection and rejection methods, which aim to identify and eliminate confrontational examples from the input before feeding it to the model, are further strategies for thwarting adversarial attacks. The feature squeezing defense is one such technique that reduces the input space of the model by combining several colors into a single color. Another method is to utilize randomized smoothing, which makes the decision boundaries smoother by adding noise to the input.

Recent studies, however, have revealed that many of these defense strategies, are ineffective against more powerful attacks (MI-FGSM). There is an increasing interest in creating stronger defenses that can survive these attacks.

To improve the robustness of deep learning models against a variety of adversarial attacks, we propose in this paper a hybrid adversarial training strategy that combines FGSM and PGD attacks. In terms of accuracy and robustness, our method outperforms other approaches, proving the value of the suggested strategy.

3. Methodology

3.1. Overview of Hybrid FGSM-PGD:

Our proposed hybrid FGSM-PGD method involves three key steps to generate adversarial examples in order to improve the robustness of deep neural networks.

First, we generate n FGSM adversarial examples with a perturbation magnitude of εf gsm. Second, we generate m PGD adversarial examples with a perturbation magnitude of εpgd and a maximum of k iterations. Finally, we combine these adversarial examples with the original clean data to form a new dataset with n + m adversarial examples and N clean examples.

We then train the deep neural network on this augmented dataset using a standard back-propagation algorithm and the standard cross-entropy loss function. The hybrid approach of using both FGSM and PGD methods helps to mitigate the limitations of each approach and improve the overall robustness of the model.

By combining FGSM and PGD, we can generate a larger and diverse set of adversarial examples, which can lead to improved performance and reduced overfitting. Additionally, using both methods can reduce the computational overhead required for training, as generating a large number of PGD adversarial examples can be computationally expensive. Overall, our proposed

ISSN: 2664-9640

hybrid FGSM-PGD method can help to improve the robustness and efficiency of deep neural networks in various applications.

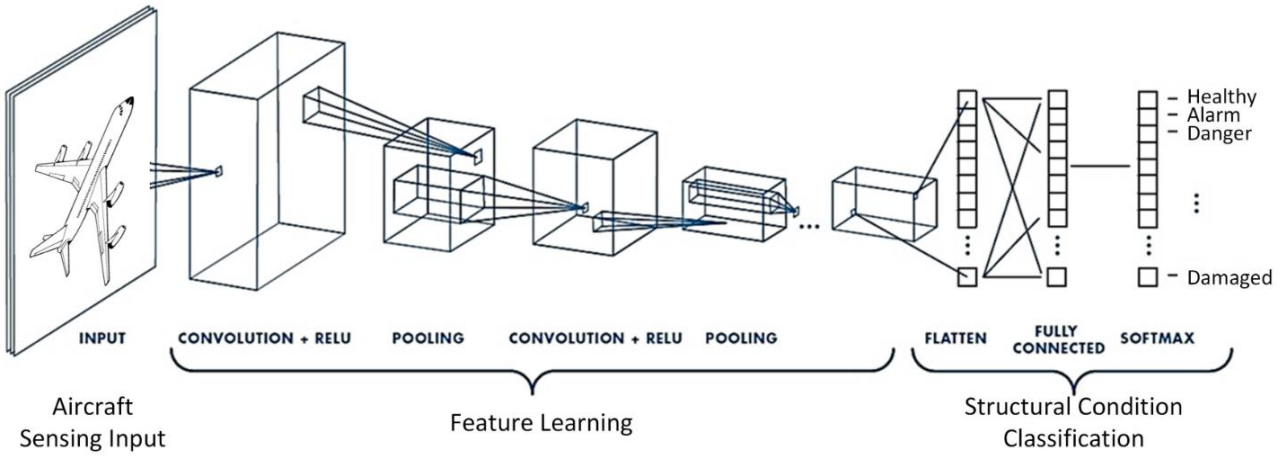


Figure 1: Convolutional neural network architecture for a classification problem. The network comprises of several layers, including two convolutional layers with ReLU activation and max pooling, followed by a flatten layer, a fully connected layer, and an output layer with softmax activation. This architecture is commonly used for image classification tasks in computer vision.

3.2. Optimization Process:

The optimization process for the hybrid FGSM-PGD method can be formulated as:

$$\text{minimize } \theta \; E(x, y) \sim D \; [\max \delta \in SL(f\theta\,(x+\delta), y)] \quad (3)$$

where θ are the parameters of the model, D is the data distribution, S is the set of allowed perturbations, δ is the adversarial perturbation, fθ is the model with parameters θ, L is the loss function, and (x, y) is a training example.

The adversarial examples are generated by the following process:

$$\text{FGSM: } \delta FGSM = \varepsilon \cdot \text{sign}(\nabla x L(f\theta\,(x), y)) \quad (4)$$

$$\text{PGD: } \delta PGD = \text{clip}\varepsilon(\delta PGD + \alpha \cdot \text{sign}(\nabla x L(f\theta\,(x+\delta PGD), y))) \quad (5)$$

where ε is the maximum allowable perturbation, α is the step size for PGD, clipε clips the perturbation to be within ε, and ∇x is the gradient with respect to the input x

The optimization is performed using the ADAM optimizer, which is a popular gradient descent optimization algorithm that uses both the gradient of the loss function and the exponential moving average of past gradients to update the model parameters. The ADAM optimizer updates the parameters with the following formula:

$$\theta t+1 = \theta t - \eta \; \sqrt{\hat{}}vt + \varepsilon \; \hat{}mt \quad (6)$$

where θt is the model parameters at iteration t, η is the learning rate, ˆmt is the estimate of the first moment of the gradients, ˆvt is the estimate of the second moment of the gradients, and ε is a small constant to prevent division by zero. The estimates of the first and second moments are calculated as follows:

$$mt = \beta 1 mt-1 + (1 - \beta 1) gt \; (7) \; vt = \beta 2 vt-1 + (1 - \beta 2) g2\,t \quad (8)$$

where gt is the gradient of the loss function at iteration t, β1 and β2 are the exponential decay rates for the first and second moments, respectively.

ISSN: 2664-9640

3.3. Implementation:

We implemented our approach as follows. First, we generated adversarial examples using FGSM and PGD attacks on the training dataset. Specifically, we used the FGSM attack with a perturbation limit of ε = 8/255 and the PGD attack with a perturbation limit of ε = 8/255 as well, step size of α = 6/255, and K = 30 iterations. We combined the generated adversarial examples with clean data to create a new dataset for training.

We split the combined dataset into 80% for training and 20% for validation. We used a learning rate of η = 0.001 and set up early stopping on the validation loss with patience of 5 to prevent overfitting of the model. We employed the Adam optimizer to train the model with a batch size of 64 and for 60 epochs.

We have trained the model to minimize the crossentropy loss function, which is defined as follows:

$$L(\theta) = 1\ N\ N\sum i=1\ \ell(f\theta\ (xi), yi) \quad (9)$$

where θ denotes the model parameters, N is the number of samples, xi and yi are the input and target label of the i-th sample, respectively, fθ is the neural network model, and ℓ is the cross-entropy loss function defined as follows:

$$\ell(\hat{}y, y) = -\ C\sum j=1\ yj\ log(\hat{}yj\ ) \quad (10)$$

where C is the number of classes, ˆy is the predicted probability distribution, and y is the one-hot encoded target label. During training, we updated the model parameters by computing the gradients of the loss function for the model parameters using back-propagation:

$$\nabla\theta\ L(\theta) = 1\ N\ N\sum i=1\ \nabla\theta\ \ell(f\theta\ (xi), yi) \quad (11)$$

We used early stopping to prevent overfitting by monitoring the validation loss during training.

4. Experiments

4.1. Experiment Setup

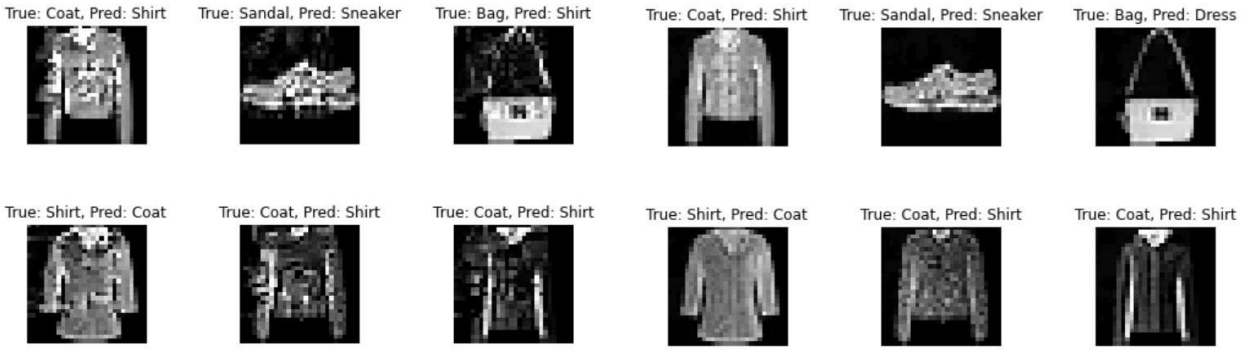
4.1.1. Datasets

We conducted experiments on three widely-used datasets; Fashion MNIST, SVHN(street View House Numbers)and CIFAR-10 datasets. Fashion MNIST contains 70k grayscale images with 60k training examples and 10k examples for testing with the size of 28x28 and belongs to 10 classes; SVHN has 600k color images of house numbers captured by Google Street view of different sizes, it contains 73257 training examples, 26032 testing examples and 531131 additional images for extra training; CIFAR-10 is a dataset of 60k color images with the size of 32x32 that belongs to 10 classes, it contains 50k training examples a 10k testing examples. We trained and tested our model on each dataset separately to showcase its versatility and effectiveness across different image types. Our results are presented in the ensuing sections.

4.1.2. Neural network architecture

In this study, we utilized a basic convolutional neural network (CNN) [31] architecture for the Fashion MNIST and CIFAR-10 datasets. The CNN architecture consists of three convolutional layers, each followed by a max-pooling layer with Rectified Linear Unit (ReLU) as the activation function. The output from the convolutional and pooling layers is then flattened using a flatten layer, followed by two fully connected dense layers. The first dense layer uses ReLU as the activation function, and the final output layer uses SoftMax for classification purposes.

ISSN: 2664-9640



(a) Adversarial images generated from Fashion MNIST dataset using the MI-FGSM attack with ε = 0.03 and k = 30 iterations, alpha α = 0.02 and decay value of 0.9

(b) Adversarial images generated from Fashion MNIST dataset using the PGD attack with ε = 0.03 and k 20 iterations, and a step size α = 0.02

Figure 2: Standard model's predictions on the adversarial images generated from Fashion MNIST dataset with the MI-FGSM and PGD attacks

For the SVHN dataset, we employed the same CNN architecture as for the Fashion MNIST and CIFAR-10 datasets. However, we modified by adding an additional convolutional layer, resulting in a total of four convolutional layers. Each convolutional layer was followed by a max-pooling layer, and the activation function used for each convolutional layer was Rectified Linear Unit (ReLU). After the convolutional and pooling layers, the output was flattened using a flatten layer, and then two fully connected dense layers were used with ReLU as the activation function for the first layer, and SoftMax for the final output layer.

4.1.3. Robust Evaluation:

We evaluated the performance of our proposed Hybrid FGSM-PGD method against four white-box attacks: Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), Iterative FGSM (IFGSM) and Momentum Iterative FGSM (MIFGSM). These attacks are commonly used to generate adversarial examples that can fool image classification models. By subjecting our model to these attacks, we aimed to evaluate its resilience and robustness against them.

4.1.4. Robust Performance Comparison

In order to evaluate the robustness of the Hybrid FGSM-PGD against white-box attacks, we compare its performance against other adversarial training methods, including the standard FGSM and PGD adversarial training and MART adversarial training approach.

Implementation Details

Among all attacks in our experiments, we evaluated the robustness of the hybrid FGSM-PGD adversarial training method against four different white box attacks: FGSM, PGD, IFGSM, and MIFGSM. For all of these attacks, we used a maximum perturbation of 8/255, 20 iterations, a step size of 16/255, and a decay factor of 0.9. The FGSM attack generates adversarial examples by perturbing each input feature by the sign of the gradient of the loss function for that feature. The PGD attack generates adversarial examples by iteratively taking a step in the direction of the gradient of the loss function for the input, and projecting the result onto the l-infinity ball of radius 8/255 centered at the original input. The IFGSM attack is a variant of PGD that performs only one step of gradient descent with the given step size, and then clips the result to lie within the l-infinity ball of radius 8/255 centered at the original input. Finally, the MIFGSM attack is a multi-step version of IFGSM that uses the decay factor to gradually reduce the step size throughout the 20 iterations.

Performance Comparison

ISSN: 2664-9640

Table 1: Fashion-MNIST Performance Comparison Results.

Models	Clean	FGSM	PGD-20	IFGSM	MI-FGSM
Standard	90.05%	43.40%	0.00	26.03%	20.64%
FGSM Adv. T	88.39%	91.90%	3.74%	88.05%	87.57%
PGD Adv. T	86.92%	50.99%	89.13%	43.45%	40.29%
MART	87.17%	57.22%	88.14%	55.34%	46.90%
Hybrid FGSM-PGD	87.95%	86.25%	88.40%	89.15%	89.06%

Table 2: SVHN Performance Comparison Results.

Models	Clean	FGSM	PGD-20	IFGSM	MI-FGSM
Standard	86.63%	11.12%	0.00%	0.42%	0.28%
FGSM Adv. T	82.22%	84.28%	16.08%	82.20%	81.58%
PGD Adv. T	78.83%	41.60%	85.29%	19.41%	16.67%
MART	77.17%	44.22%	85.14%	27.34%	20.90%
Hybrid FGSM-PGD	78.22%	87.28%	83.93%	86.01%	82.18%

Table 1 shows the performance comparison of different models on the Fashion MNIST dataset against various white-box attacks, including FGSM, PGD, IFGSM and MI-FGSM. Table 1 shows the accuracy of each model on clean data and the accuracy on data that has been perturbed by each attack. According to the results presented in Table 1, we can make the following observations: First, the standard or baseline model achieved 90.05% accuracy on clean data, but its accuracy decreased significantly under all the considered attacks. The FGSM adversarial training model achieved a higher accuracy on clean data (88.39%) and as expected it showed better robustness to FGSM adversarial examples with an accuracy of 91.90%. However, its accuracy decreased substantially under PGD attack. The PGD adversarial training model was the most robust model against PGD, but its accuracy significantly decreased under other attacks. The Hybrid FGSM-PGD model achieved an accuracy of 87.95% on clean data and demonstrated higher robustness against FGSM and PGD attacks, with accuracy rate of 86.25% and 88.40%. Moreover, the hybrid model achieved higher accuracy rates on IFGSM and MI-FGSM attacks than the other models and also had a good accuracy on the PGD adversarial attacks. Overall, the Hybrid model's results indicate its superiority to the other models against white box attacks on the Fashion MNIST dataset.

Table 2 shows a performance comparison using the SVHN dataset, and similar conclusions can be drawn from it. as those on the Fashion MNIST in Table 1. The Hybrid FGSMPGD model achieves good accuracy rates on adversarial examples among all evaluated models with accuracy rates above 83% for all attack methods while maintaining a fairly high clean image accuracy of 78.22%. These results suggest that the hybrid FGSM-PGD model is a promising approach for the robustness of deep learning models.

Table 3: CIFAR-10 Performance Comparison Results.

Models	Clean	FGSM	PGD-20	IFGSM	MI-FGSM
Standard	76.63%	8.02%	0.00%	0.79%	0.50%

ISSN: 2664-9640

FGSM Adv. T	66.02%	60.96%	8.91%	58.44%	57.09%
PGD Adv. T	71.17%	20.22%	74.14%	6.34%	4.90%
MART	72.26%	25.33%	72.27%	11.87%	9.12%
Hybrid FGSM-PGD	68.45%	61.14%	75.26%	58.90%	58.22%

As shown in table 3 for the CIFAR-10 dataset, the hybrid FGSM-PGD model performs better than the other models under most of the attack scenarios, with the highest accuracy under the PGD-20 attack (75.26%). even so, it still has a lower accuracy than the standard model on clean images, indicating the trade-off between robustness and accuracy. Overall, the results demonstrate the effectiveness of the hybrid FGSM-PGD method in improving the robustness of machine learning models against adversarial attacks while maintaining a reasonable level of accuracy on clean images.

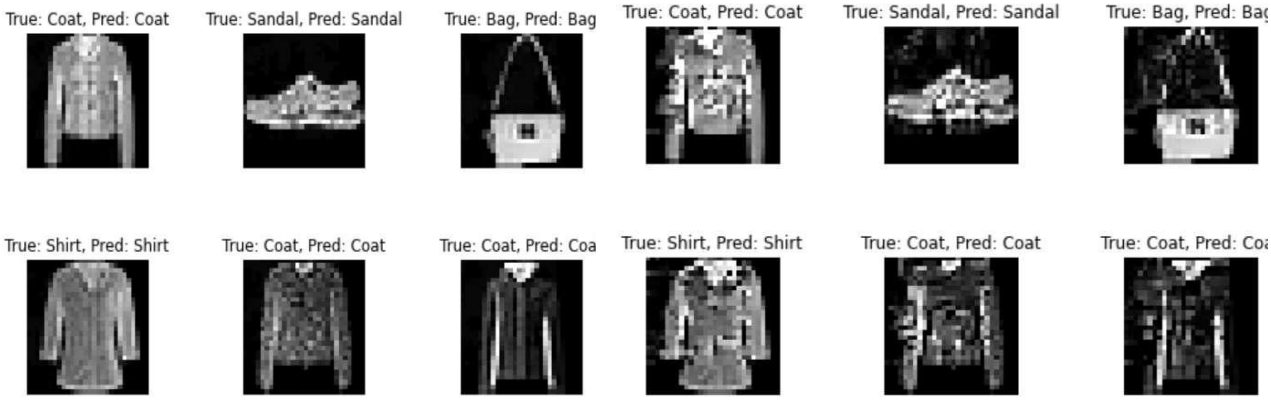


Figure 3: Hybrid model's performance on the adversarial images generated from Fashion MNIST dataset with the MI-FGSM and PGD attacks

5. Discussion

The presented experimental results in the tables provide evidence that the hybrid FGSM-PGD adversarial training method is highly effective in improving the robustness of deep learning models against various adversarial attacks. The hybrid FGSM-PGD model demonstrates higher accuracy rates than the standard model in all three datasets, namely CIFAR-10, SVHN, and Fashion-MNIST datasets. Additionally, the hybrid model outperforms the FGSM adversarial training model (FGSM Adv.T), PGD adversarial training model (PGD Adv.T), and MART models in certain experiments. For instance, in the Fashion-MNIST dataset, the hybrid model surpasses the MART model in the PGD-20, IFGSM, and MI-FGSM experiments. Similarly, in the SVHN dataset, the hybrid model shows better results than the FGSM adversarial training method and MART models in the PGD-20 experiment.

One reason for the hybrid FGSM-PGD model's superior performance is its ability to combine the FGSM and PGD attacks' strengths. The FGSM attack is known for its fast and efficient approach to creating adversarial examples, but it produces weak ones. Meanwhile, the PGD attack is computationally expensive but generates robust adversarial examples. However, the hybrid model exploits the benefits of both attacks during the training process to improve the model's ability to withstand various types of white box adversarial attacks.

The hybrid model also uses a scheduled training process that gradually introduces the PGD attack into the training process. This gradual introduction helps the model learn progressively

ISSN: 2664-9640

more complex decision boundaries, leading to better robustness against white-box adversarial attacks. Overall, the hybrid FGSM-PGD model provides a promising approach to improving the robustness of deep learning models against adversarial attacks, and its performance merits further investigation.

6. Conclusion

In conclusion, this paper proposed a novel adversarial training technique, the Hybrid FGSM-PGD method, which combines the FGSM and PGD approaches to enhance the robustness of deep learning models against various white box-attacks. Our experiments on Fashion MNIST, SVHN, and CIFAR-10 datasets showed that our method outperforms other adversarial training techniques in terms of practicality and effectiveness. Furthermore, our proposed method can be easily integrated into existing adversarial training frameworks. These results suggest that the Hybrid FGSM-PGD method can be a promising solution for improving the security of deep learning applications.

References

[1] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In Proceedings of the international conference on learning representations. Retrieved from https://openreview.net/forum?id=rJzIBfZAb

[2] Mansour, Y., Mohri, M., & Rostamizadeh, A. (2009). Domain adaptation: Learning bounds and algorithms. arXiv preprint arXiv:0902.3430.

[3] Mishkin, D., Sergievskiy, N., & Matas, J. (2017). Systematic evaluation of convolution neural network advances on the imagenet. Computer Vision and Image Understanding.

[4] Carlini, N., & Wagner, D. (2017). Adversarial examples are not easily detected: Bypassing ten detection methods. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (pp. 3-14). ACM.

[5] Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy (SP) (pp. 39-57). IEEE.

[6] Feinman, R., Curtin, R. R., Shintre, S., & Gardner, A. B. (2017). Detecting adversarial samples from artifacts. arXiv preprint arXiv:1703.00410.

[7] Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In International Conference on Machine Learning (pp. 10501059).

[8] Gowal, S., Qin, C., Uesato, J., Mann, T., and Kohli, P. Uncovering the limits of adversarial training against norm-bounded adversarial examples, 2021.

[9] Sehwag, V., Mahloujifar, S., Handina, T., Dai, S., Xiang, C., Chiang, M., and Mittal, P. Improving adversarial robustness using proxy distributions. arXiv preprint arXiv:2104.09425, 2021.

[10] Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. In International Conference on Learning Representations, 2020.

[11] Wu, D., Xia, S.-T., and Wang, Y. Adversarial weight perturbation helps robust generalization. In Advances in Neural Information Processing Systems, 2020. X

[12] Ren, H., Huang, T., & Yan, H. (2021). Adversarial examples: attacks and defenses in the physical world. International Journal of Machine Learning and Cybernetics, 1-12.

[13] Rony, J., Hafemann, L. G., Oliveira, L. S., Ayed, I. B., Sabourin, R., & Granger, E. (2018). Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. arXiv preprint arXiv:1811.09600.

[14] Shi, Y., Han, Y., Zhang, Q., & Kuang, X. (2020). Adaptive iterative attack towards explainable adversarial robustness. Pattern Recognition, 107309.

ISSN: 2664-9640

[15] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929-1958.

[16] Sutanto, R. E., & Lee, S. (2021). Real-time adversarial attack detection with deep image prior initialized as a high-level representation based blurring network. Electronics, 10(1), 52.

[17] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.

[18] Theagarajan, R., & Bhanu, B. (2020). Defending black box facial recognition classifiers against adversarial attacks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 812-813).

[19] Yang, Y.-Y., Rashtchian, C., Zhang, H., Salakhutdinov, R. R., and Chaudhuri, K. A closer look at accuracy vs. robustness. In Advances in Neural Information Processing Systems, 2020.

[20] Zhang, H., Yu, Y., Jiao, J., Xing, E., Ghaoui, L. E., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In Proceedings of the 36th International Conference on Machine Learning, 2019.

[21] Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., and Kankanhalli, M. Attacks which do not kill training make adversarial learning stronger. In Proceedings of the 37th International Conference on Machine Learning, 2020.

[22] Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. Geometry-aware instance-reweighted adversarial training. In International Conference on Learning Representations, 2021.

[23] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. arXiv preprint arXiv:0902.3430, 2009.

[24] Dmytro Mishkin, Nikolay Sergievskiy, and Jiri Matas. Systematic evaluation of convolution neural network advances on the ImageNet. Computer Vision and Image Understanding, 2017.

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

[26] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In International Conference on Learning Representations.

[27] Graese, A., Rozsa, A., & Boult, T. E. (2016). Assessing threat of adversarial examples on deep neural networks. In Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on (pp. 69-74). IEEE.

[28] Guo, C., Rana, M., Cisse, M., & van der Maaten, L. (2018). Countering adversarial images using input transformations. In International Conference on Learning Representations.

[29] Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images. Tech. Rep., Citeseer.

[30] Kurakin, A., Goodfellow, I., & Bengio, S. (2017). Adversarial machine learning at scale. In International Conference on Learning Representations.

[31] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems (pp. 1097–1105).

[32] LeCun, Y. (1998). The MNIST database of handwritten digits. Retrieved from http://yann. lecun.com/exdb/mnist/.

[33] Tramer, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. In International Conference on Learning Representations. Retrieved from https://openreview.net/forum?id=rkZvSe-RZ.