Enhancing Neural Network Resilence against Adversarial Attacks based on FGSM Technique

Mohamed Ben Ammar
Department of Information Systems, Faculty of Computing and Information Technology, Northern Border University, Saudi Arabia
mohammed.ammar@nbu.edu.sa

Refka Ghodhbani
Department of Computer Science, Faculty of Computing and Information Technology, Northern Border University, Saudi Arabia
refka.ghodhbani@nbu.edu.sa (corresponding author)

Taoufik Saidani
Department of Computer Science, Faculty of Computing and Information Technology, Northern Border University, Saudi Arabia
taoufik.saidan@nbu.edu.sa

Received: 13 April 2024 | Revised: 23 April 2024 and 25 April 2024 | Accepted: 27 April 2024

 DOI: https://doi.org/10.48084/etasr.7379

ABSTRACT

The robustness and reliability of neural network architectures are put to the test by adversarial attacks, resulting in inaccurate findings and affecting the efficiency of applications operating on Internet of Things (IoT) devices. This study investigates the severe repercussions that might emerge from attacks on neural network topologies and their implications on embedded systems. In particular, this study investigates the degree to which a neural network trained in the MNIST dataset is susceptible to adversarial attack strategies such as FGSM. Experiments were conducted to evaluate the effectiveness of various attack strategies in compromising the accuracy and dependability of the network. This study also examines ways to improve the resilience of a neural network structure through the use of adversarial training methods, with particular emphasis on the APE-GAN approach. The identification of the vulnerabilities in neural networks and the development of efficient protection mechanisms can improve the security of embedded applications, especially those on IoT chips with limited resources.

Keywords-neural networks; adversarial attack; robustness

I. INTRODUCTION

Technological environments have undergone substantial changes in the last years due to the widespread use of neural networks in numerous applications, particularly in embedded systems such as Internet of Things (IoT) devices. Complex computational models have demonstrated remarkable performance in various domains, including natural language processing and image categorization. As a result, complex data analysis and decision-making are now more accurate and efficient. The widespread adoption of this technology has sparked a new wave of innovation and development that is propelling an expansion in numerous industries and businesses [1]. Deep learning, a kind of machine learning, has revolutionized computational methods by allowing models to learn from data and gain an organized understanding of the world. Deep learning methods improve data representation across successive layers by analyzing large datasets and identifying intricate patterns through the adoption of backpropagation techniques. With the advent of high-performance technology and Deep Neural Network (DNN) architectures, deep learning has made a significant progress in many areas. This advancement extends to novel domains, such as drug molecule analysis and brain circuit reconstruction, as well as to more traditional areas, namely speech recognition and image classification. Due to the extraordinary accuracy of these models, prominent players in the industry, involving Google, Alibaba, Intel, and Nvidia have contributed significantly to the advancement of AI-powered services [2].

Initiatives must be taken to limit the risks posed by adversarial attacks and address the vulnerabilities existing in neural network topologies to preserve the ongoing efficacy and reliability of such systems [3, 4]. This has led to an increase in the research efforts made in a variety of domains, aimed at clarifying the basic mechanisms underlying adversarial attacks, determining how resilient neural network architectures are

against such threats, and creating effective defensive strategies to protect against malicious modifications [5]. By investigating many facets of neural network vulnerabilities and offering innovative solutions to boost the resilience and security of neural networks, numerous studies have unveiled the diversity of the topic. However, concerns about the security and integrity of DNNs have grown in significance, as these systems leave the lab and find use in real-world applications. Adversaries can cause trained models to produce false output even if they can stealthily alter legitimate inputs, which are frequently invisible to human observers.

In [6], new insights were provided into how adversaries can target high-performing DNNs, underscoring the urgent need for robust security measures. Vulnerabilities have been discovered in Voice-Controllable Systems (VCS) and Automatic Speech Recognition (ASR) [7, 8], underscoring how commonplace this security risk is. In [2], attacks against driverless vehicles were disclosed. The former manipulated the traffic signals to mislead the learning algorithms. This made the concerns about the security of AI-driven systems in critical situations even more acute. Adversarial training of linear models has been supported by an extensive analysis and experiments conducted in related studies, such as in [10]. In [11], the generalization properties of adversarial situations were studied. The diverse approaches provided in [12] are indicative of the various efforts currently underway to address the evolving landscape of neural network safety. These strategies focus on applying distributed deep learning algorithms to ensure the privacy of the training data.

II. RELATED WORKS

Based on computer vision concepts, adversarial attacks are carefully designed to distort a classifier's decision-making process and significantly affect important performance measures, such as accuracy, loss, and precision [13]. There are several types of attacks, such as white-box attacks, black-box attacks, and techniques, like the Forward Gradient Descent (FGD), Momentum Iterative Method (MIM), and FGSM, among others [14]. Adversarial attacks are divided into two main categories: white-box and black-box attacks [15]. White-box attacks, which are characterized by the attacker's extensive knowledge of the architecture, variables, and gradients of the model under attack, provide a sophisticated way to use adversarial tactics [16]. Armed with this deep knowledge, attackers carefully plan perturbations to input data to take advantage of weaknesses in the model's decision-making process. Black-box attacks are restricted in their ability to access the internal workings of the target model. Attackers in these situations have no knowledge of the model's underlying structure or parameters and are only able to depend on its input-output behavior. In this case, attackers use black-box strategies, such as transferability [17], which involves adapting adversarial instances created for one model to another with similar behavior. The key distinction between white-box and black-box attacks is the amount of information that the attacker may access. White-box attacks use extensive internal knowledge, whereas black-box attacks operate within the boundaries of information access restrictions.

Numerous studies, involving [18], have defined several types of adversarial attacks. In white-box attacks, attackers are assumed to be well-versed with the target model's parameters, training protocols, training data, and architectural specifications, namely input structure, weight values, activation functions, and training methodology. These attacks can be divided into two other subcategories: targeted attacks and non-targeted attacks. Non-targeted attacks just attempt to cause misclassifications, regardless of the final class. In targeted attacks, the adversarial example is specifically designed to elicit a certain classification, for instance classifying all photographs as featuring one specific person. The generation of adversarial examples involves iterative access to the model to compute gradients. White-box attacks can be segmented into two main categories: gradient-based concepts and constrained concepts [19].

III. PROPOSED METHOD: ENHANCING CNN SECURITY AGAINST ADVERSARIAL ATTACKS

The proposed method was designed to improve the security of CNNs against adversarial attacks [20, 21]. Figure 1 shows the design of the proposed approach for a CNN, specifically designed for the MNIST dataset. This study also incorporates a cutting-edge augmentation strategy, using APE-GAN, indicating how it can be customized to withstand an adversarial attack.

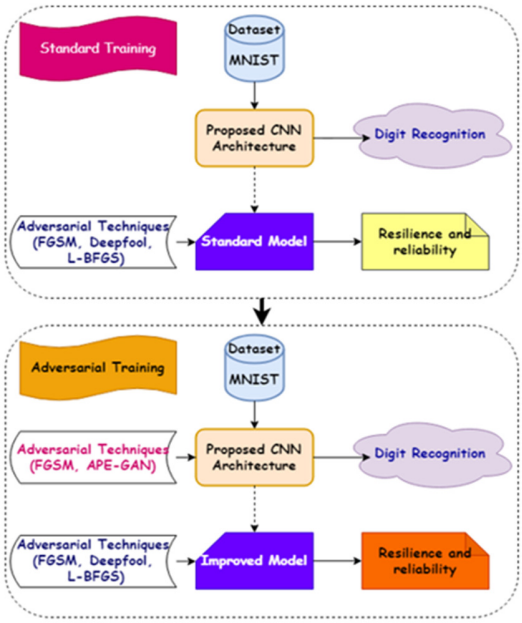
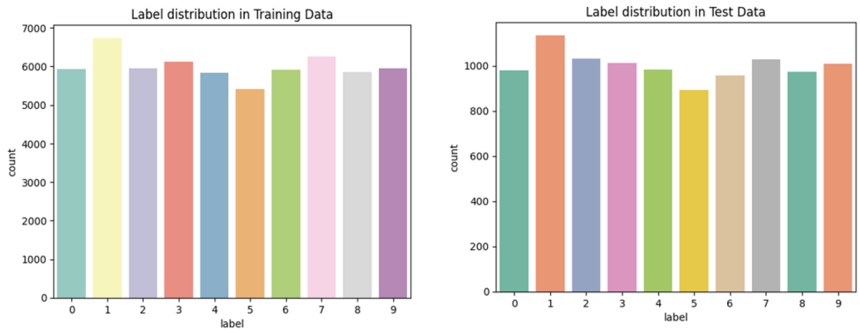


Fig. 1. Proposed method to enhance CNN resilience against adversarial attacks.

A. MNIST Dataset

The MNIST dataset has a prominent place in the area of neural networks, is well recognized for its role as a fundamental benchmark in image classification tasks, and is a fundamental tool for training and evaluating different algorithms [22]. It is made up of a series of 28×28 pixel grayscale images with handwritten digits. Due to its wide applicability and consistent structure, the MNIST dataset provides a reliable framework for assessing the resilience of

CNN models against adversarial attacks. By subjecting CNN architectures trained on the MNIST dataset to rigorous testing following adversarial techniques, important insights can be obtained into the effectiveness of protective mechanisms and techniques. The dataset was divided into 60,000 samples for the train set and 10,000 samples for the test set. Figure 2 presents the label distribution of the training and testing sets, respectively, while Figure 3 provides samples of the MNIST dataset.



(a) (b)

Fig. 2. Label distributions of the (a) training and (b) testing sets.

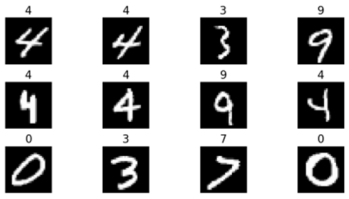


Fig. 3. MNIST dataset sample.

B. CNN Architecture

The CNN was designed employing a layering scheme to quickly analyze and extract characteristics from the input data. It consisted of three convolutional layers to extract complex spatial patterns and characteristics from the input data. Three batch normalization layers were carefully included to speed up and stabilize the training process. Three dropout layers were also included to reduce overfitting by randomly deactivating a subset of neurons during training. The output of the convolutional layers was then reshaped into a one-dimensional array by a flattened layer, which smoothed the transition to fully linked layers. This architecture offers a strong framework that is appropriate for tasks, such as image recognition and classification, thanks to its two thick layers that are responsible for categorization based on the learned features.

C. FGSM Technique

FSGM stands as a prominent technique in the realm of adversarial attacks, particularly in the context of neural networks. To achieve its goal of fooling the target model into providing inaccurate output, this approach works by introducing minor perturbations into the input data [10]. These perturbations are carefully constructed and strategically placed. The fundamental idea of the FSGM algorithm is the use of gradient information of the loss function in relation to the input data to ascertain the direction in which to perturb the input. Mathematically, the perturbed input x' can be computed as:

$$x' = x + \epsilon \cdot sign(\nabla_x J(x, y_{true}) \quad (1)$$

where x' represents the perturbed input data, x denotes the original input data, ϵ denotes the magnitude of the perturbation, and $\nabla_x J(x, y_{true})$ represents the gradient of the loss function J for the input x. Finally, y_{true} is the true label of the input data.

TABLE I. CNN ARCHITECTURE

Layer (type)	Output shape	Param. #
Conv2d	(None, 28, 28, 20)	520
Batch_normalization	(None, 28, 28, 20)	80
Dropout (Dropout)	(None, 28, 28, 20)	0
Conv2d_1 (Conv2D)	(None, 28, 28, 20)	6420
Batch_normalization_1	(None, 28, 28, 20)	80
Dropout_1 (Dropout)	(None, 28, 28, 20)	0
Conv2d_2 (Conv2D)	(None, 28, 28, 20)	6420
Batch_normalization_2	(None, 28, 28, 20)	80
Dropout_2 (Dropout)	(None, 28, 28, 20)	0
Flatten (Flatten)	(None, 156688)	0
Dense (Dense)	(None, 200)	3136200
Dense_1(Dense)	(None, 10)	2010
Total params: 3,151,810		
Trainable params: 3,151,690		
Non-trainable params: 120		

IV. RESULTS AND DISCUSSIONS

A. Standard Training Results of the Proposed CNN

The accuracy and validation accuracy curves presented in Figure 5 provide an insight into the model's overall performance and its ability to generalize on unseen data. The accuracy of the model reaches approximately 99.86%. The loss and validation loss curves offer a glimpse into the convergence behavior of the model during training, indicating the degree of error reduction over epochs reaching 0.0047. Table I portrays the evaluation metrics for precision, recall, and F1-score of the CNN, gauging its effectiveness in predicting digits. The categorized digits indicate average precision, average recall, and average F1-score of almost 100%. Figure 6 depicts the confusion matrix, providing a comprehensive overview of the model's performance in accurately predicting the digits. This

matrix displays the number of correct predictions (true positives) along the diagonal, whereas off-diagonal elements indicate misclassifications. The confusion matrix discloses which digits are frequently misclassified, providing an understanding into potential areas for improvement. The well-structured confusion matrix displays a strong diagonal pattern, indicating a high proportion of correct predictions across all digits and demonstrating the model's effectiveness in accurately identifying and classifying digits.

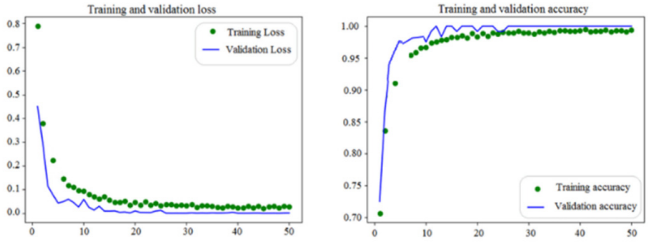


Fig. 4. Accuracy and loss curves.

TABLE II. CNN PERFORMANCE EVALUATION

Classes	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	422
1	1.00	0.99	1.00	473
2	1.00	0.99	1.00	409
3	1.00	1.00	1.00	426
4	0.99	1.00	1.00	429
5	1.00	1.00	1.00	382
6	1.00	1.00	1.00	412
7	0.99	1.00	1.00	469
8	0.99	1.00	1.00	384
9	1.00	0.99	1.00	394
Accuracy			1.00	4200
Macro avg	1.00	1.00	1.00	4200
Weighted avg	1.00	1.00	1.00	4200

B. Attack Results on the Standard CNN model

1) FGSM Attack Results

The effects of the FGSM attack on the performance of the CNN model were investigated by varying the epsilon parameter, which controls the magnitude of the injected error in the input images. Varying classification degrees were observed by systematically adjusting the epsilon and applying the FGSM attack to test images from the MNIST dataset. As the epsilon increases, the perturbations introduced into the input images become more pronounced, resulting in a higher rate of misclassifications. This analysis manifests how different levels of perturbation affect the model's ability to correctly classify digits. By examining the misclassified test images across different epsilon values, valuable insights are provided into the model's sensitivity to adversarial attacks. These insights offer recommendations for the development of robust defense strategies to mitigate such vulnerabilities. Figure 6 presents the attack results of the FGSM technique on the standard CNN model.

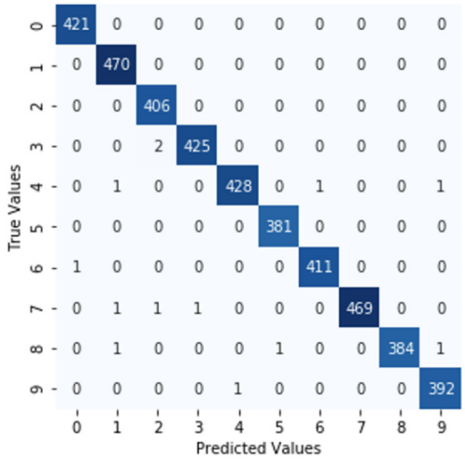


Fig. 5. Confusion matrix,

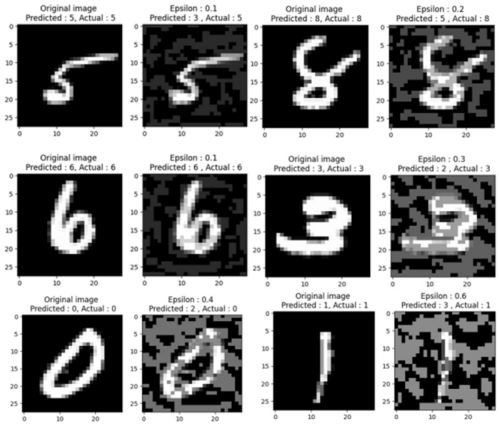


Fig. 6. FGSM attack.

C. FGSM-based Adversarial Training

The use of FGSM-based adversarial training resulted in a loss of 0.044 and an accuracy of 98.83%, underscoring the resilience of CNN models against adversarial attacks. Although these models excel in accuracy when handling clean data, their performance decreases notably when confronted with adversarial examples constructed through FGSM. This serves as a stark reminder of the necessity for rigorous robustness testing and the implementation of defenses to counter adversarial threats. Adversarial training, particularly employing the FGSM technique, presents a promising strategy to enhance model robustness. By incorporating both clean and adversarial examples during training, models can adapt and strengthen their defenses against such attacks.

TABLE III. ADVERSARIAL TRAINING PERFORMANCE

Test scenario	Standard CNN accuracy (%)	APE-GAN CNN accuracy (%)	Improvement (%)
Original Data	99.86	99.98	0.12
Adversarial attacks			
FGSM attack ($\varepsilon = 0.1$)	60.2	85.7	25.5
FGSM attack ($\varepsilon = 0.3$)	40.1	78.9	38.8
LBFGS attack	30.8	69.4	38.6

Adjusting the epsilon parameter of the FGSM technique, Figure 7 depicts the robustness of the adversarially trained CNN against FGSM attacks. This modification showcases the model's ability to withstand varying levels of adversarial perturbations, highlighting the effectiveness of the adversarial training approach in fortifying the model against equivalent attacks.

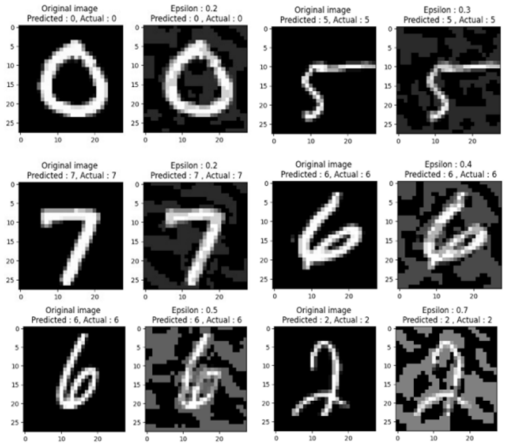


Fig. 7. FGSM attack on the adversarial trained CNN model.

V. CONCLUSION

This study stresses the efficacy of the APE-GAN in fortifying machine learning models against adversarial attacks. By comparing APE-GAN with traditional techniques, such as FGSM and other related approaches, significant improvements were achieved in both accuracy and robustness. These results underscore the promising potential of APE-GAN in addressing the vulnerabilities posed by adversarial attacks, thus advancing the field of adversarial machine learning. As the threat of adversarial attacks persists, the findings of this study emphasize the importance of adopting robust techniques, like APE-GAN, to enhance the security and reliability of machine learning systems. Looking ahead, further exploration of APE-GAN's scalability and applicability across various datasets and domains could offer valuable insights for developing more resilient machine-learning solutions in real-world settings. Among the next tasks for APE-GAN is the stabilization of the prediction via the use of majority voting from a collection of the samples created after burn-in, as well as the development of tools that estimate the gradient with more precision in high-dimensional spaces.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA, for funding this research work through project number NBU-FFR-2024-2461-01.

REFERENCES

[1] H. Qiu, T. Dong, T. Zhang, J. Lu, G. Memmi, and M. Qiu, "Adversarial Attacks Against Network Intrusion Detection in IoT Systems," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10327–10335, Jul. 2021, https://doi.org/10.1109/JIOT.2020.3048038.

[2] Y. Wang, Y. Tan, W. Zhang, Y. Zhao, and X. Kuang, "An adversarial attack on DNN-based black-box object detectors," *Journal of Network and Computer Applications*, vol. 161, Jul. 2020, Art. no. 102634, https://doi.org/10.1016/j.jnca.2020.102634.

[3] A. Aldahdooh, W. Hamidouche, S. A. Fezza, and O. Déforges, "Adversarial example detection for DNN models: a review and experimental comparison," *Artificial Intelligence Review*, vol. 55, no. 6, pp. 4403–4462, Aug. 2022, https://doi.org/10.1007/s10462-021-10125-w.

[4] L. Liu, Y. Guo, Y. Cheng, Y. Zhang, and J. Yang, "Generating Robust DNN With Resistance to Bit-Flip Based Adversarial Weight Attack," *IEEE Transactions on Computers*, vol. 72, no. 2, pp. 401–413, Oct. 2023, https://doi.org/10.1109/TC.2022.3211411.

[5] R. Duan, Y. Chen, D. Niu, Y. Yang, A. K. Qin, and Y. He, "AdvDrop: Adversarial Attack to DNNs by Dropping Information," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, Canada, Oct. 2021, pp. 7486–7495, https://doi.org/10.1109/ICCV48922.2021.00741.

[6] C. Szegedy *et al.*, "Intriguing properties of neural networks." arXiv, Feb. 19, 2014, https://doi.org/10.48550/arXiv.1312.6199.

[7] N. Carlini *et al.*, "Hidden voice commands," in *Proceedings of the 25th USENIX Conference on Security Symposium*, Austin, TX, USA, Aug. 2016, pp. 513–530.

[8] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "DolphinAttack: Inaudible Voice Commands," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas, TX, USA, Oct. 2017, pp. 103–117, https://doi.org/10.1145/3133956.3134052.

[9] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial Machine Learning at Scale," arXiv, Feb. 2017, https://doi.org/10.48550/arXiv.1611.01236.

[10] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples." arXiv, Mar. 20, 2015, https://doi.org/10.48550/arXiv.1412.6572.

[11] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples." arXiv, May 23, 2016, https://doi.org/10.48550/arXiv.1605.07277.

[12] M. Abadi *et al.*, "Deep Learning with Differential Privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Vienna, Austria, Oct. 2016, pp. 308–318, https://doi.org/10.1145/2976749.2978318.

[13] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li, "Adversarial Attacks on Deep-learning Models in Natural Language Processing: A Survey," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 3, pp. 1–41, Dec. 2020, https://doi.org/10.1145/3374217.

[14] R. R. Wiyatno, A. Xu, O. Dia, and A. de Berker, "Adversarial Examples in Modern Machine Learning: A Review." arXiv, Nov. 2019, https://doi.org/10.48550/arXiv.1911.05268.

[15] G. B. Ingle and M. V. Kulkarni, "Adversarial Deep Learning Attacks—A Review," in *Information and Communication Technology for Competitive Strategies (ICTCS 2020)*, Singapore, Jul. 2021, pp. 311–323, https://doi.org/10.1007/978-981-16-0882-7_26.

[16] Y. Lin, H. Zhao, Y. Tu, S. Mao, and Z. Dou, "Threats of Adversarial Attacks in DNN-Based Modulation Recognition," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, Toronto, Canada, Aug. 2020, pp. 2469–2478, https://doi.org/10.1109/INFOCOM41043.2020.9155389.

[17] H. Xu *et al.*, "Adversarial Attacks and Defenses in Images, Graphs and Text: A Review," *International Journal of Automation and Computing*, vol. 17, no. 2, pp. 151–178, Apr. 2020, https://doi.org/10.1007/s11633-019-1211-x.

[18] A. Aldahdooh, W. Hamidouche, S. A. Fezza, and O. Déforges, "Adversarial example detection for DNN models: a review and experimental comparison," *Artificial Intelligence Review*, vol. 55, no. 6, pp. 4403–4462, Aug. 2022, https://doi.org/10.1007/s10462-021-10125-w.

[19] S. Y. Khamaiseh, D. Bagagem, A. Al-Alaj, M. Mancino, and H. W. Alomari, "Adversarial Deep Learning: A Survey on Adversarial Attacks and Defense Mechanisms on Image Classification," *IEEE Access*, vol. 10, pp. 102266–102291, 2022, https://doi.org/10.1109/ACCESS.2022.3208131.

[20] U. Diaa, "A Deep Learning Model to Inspect Image Forgery on SURF Keypoints of SLIC Segmented Regions," *Engineering, Technology & Applied Science Research*, vol. 14, no. 1, pp. 12549–12555, Feb. 2024, https://doi.org/10.48084/etasr.6622.

[21] G. Alotibi, "A Cybersecurity Awareness Model for the Protection of Saudi Students from Social Media Attacks," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13787–13795, Apr. 2024, https://doi.org/10.48084/etasr.7123.

[22] A. Alotaibi and M. A. Rassam, "Enhancing the Sustainability of Deep-Learning-Based Network Intrusion Detection Classifiers against Adversarial Attacks," *Sustainability*, vol. 15, no. 12, pp. 1–25, 2023, https://doi.org/10.3390/su15129801.