electronics

Article

Robustness and Transferability of Adversarial Attacks on Different Image Classification Neural Networks

Kamilya Smagulova [1,†], Lina Bacha [2,†], Mohammed E. Fouda [3,*], Rouwaida Kanj [2] and Ahmed Eltawil [1]

[1] Division of Computer, Electrical and Mathematical Sciences and Engineering (CEMSE), King Abdullah University of Science and Technology, Thuwal 23955, Saudi Arabia
[2] Department of Electrical and Computer Engineering, American University of Beirut, Beirut 1107 2020, Lebanon
[3] Rain Neuromorphics, Inc., San Francisco, CA 94110, USA
* Correspondence: foudam@uci.edu
† These authors contributed equally to this work.

Abstract: Recent works demonstrated that imperceptible perturbations to input data, known as adversarial examples, can mislead neural networks' output. Moreover, the same adversarial sample can be transferable and used to fool different neural models. Such vulnerabilities impede the use of neural networks in mission-critical tasks. To the best of our knowledge, this is the first paper that evaluates the robustness of emerging CNN- and transformer-inspired image classifier models such as SpinalNet and Compact Convolutional Transformer (CCT) against popular white- and black-box adversarial attacks imported from the Adversarial Robustness Toolbox (ART). In addition, the adversarial transferability of the generated samples across given models was studied. The tests were carried out on the CIFAR-10 dataset, and the obtained results show that the level of susceptibility of SpinalNet against the same attacks is similar to that of the traditional VGG model, whereas CCT demonstrates better generalization and robustness. The results of this work can be used as a reference for further studies, such as the development of new attacks and defense mechanisms.

Keywords: adversarial attacks; robustness; transferability; CCT; VGG; SpinalNet; ART toolbox

check for updates

Citation: Smagulova, K.; Bacha, L.; Fouda, M.E.; Kanj, R.; Eltawil, A. Robustness and Transferability of Adversarial Attacks on Different Image Classification Neural Networks. *Electronics* **2024**, *13*, 592. https://doi.org/10.3390/electronics13030592

Academic Editor: Ping-Feng Pai

Received: 6 June 2023
Revised: 16 January 2024
Accepted: 24 January 2024
Published: 31 January 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

A growing amount of data and the efficiency of deep neural networks (DNNs) for complex learning problems make deep learning-based solutions ubiquitous in a wide range of domains. The popular neural architectures and their applications include convolutional neural networks for image classification [1], long short-term memory for speech recognition [2], and transformer neural networks for machine translation [3]. However, in [4], the authors reported two counter-intuitive properties that can be used to deceive the neural network. The process of intentional generation and the addition of such feature perturbations designed to mislead the neural model is now known as an adversarial attack. Later, more works demonstrated that different neural models exhibit different levels of adversarial vulnerability and robustness, which are typically measured by achieved accuracy after the applied attack [5–7]. In real-world applications, adversarial attacks and defenses of DNNs in medical imaging showed that even a limited amount of perturbation could significantly impact the result of diagnosis [8]. Moreover, attacks can be crafted in real-time, as in the case of autonomous driving systems [9]. As a result, a new area of adversarial attack defense has emerged, which can be implemented via model and/or data optimization and using additional networks [10]. The applied perturbations to the input can be measured by different ℓ_p-norms and used to determine the strength of the attack [11]. Susceptibility to ℓ_p-norm bounded attacks implies low robustness capabilities of a model against realistic threats [12]. Moreover, recent works demonstrated that an adversarial example designed for the easy-to-attack model can attack the hard-to-attack model.

Currently, the research area of adversarial attacks and defense is highly active, and there is a growing number of studies. According to the proposed taxonomy [13], adversarial attacks can be injected at the training or testing phases. Perturbations added to the input data to generate adversarial examples can be specific or universal. In the first case, different perturbations are added to each input, resulting in different perturbation patterns. In the latter case, added perturbations are the same [10]. Based on the number of iterations required to generate an adversarial example, attacks can be either one-shot/one-step or iterative [10]. In addition, the types of adversarial attacks can be tentatively classified based on the degree of knowledge, adversarial scenarios, and goals [13]. If the attacker has complete knowledge of the neural model parameters, then the attack is known as a white-box attack. In contrast, if the attacker has zero knowledge, it is known as a black-box attack [14]. In the case of a limited degree of knowledge, the attacks belong to the gray-box class. Depending on the adversarial goals, attacks can be targeted, untargeted, or aim to reduce the degree of confidence of neural networks. Untargeted attacks try to misguide the model to predict any of the incorrect classes, whereas targeted attacks aim to misguide the model to produce specific incorrect classes [15]. Moreover, an adversarial example can be designed on different training datasets [16]. This is known as the transferability. Therefore, along with the robustness to adversarial attacks, some works such as [17] tested the concept of transferability of the models and searched for ways to protect systems and to create a defense against attacks.

Studying adversarial robustness and transferability is crucial for several reasons. Firstly, model trustfulness and security are important for deploying transparent mission- and business-critical applications. Secondly, exploring the generalizability and transferability of the model is one of the major steps toward developing a defense mechanism. This work explores the robustness of CNN- and transformer-based classifier neural model architectures against white-box and black-box evasion adversarial attacks and their transferability. The studied neural models include the popular Very Deep Convolutional Network (VGG) [18], somatosensory system-inspired SpinalNet [19], and the newly developed CCT-$7/3 \times 1$ [20]. The performance of the VGG model, including its vulnerability, is widely studied and often serves as a foundational baseline for many neural networks [21]. SpinalNet is promising for multi-scale feature extraction, whereas CCT relies on hierarchical feature extraction and decomposition. To the best of our knowledge, the robustness and transferability of these models have not been studied before. Their robustness was examined against four white-box attacks: Carlini and Wagner ℓ_2, Carlini and Wagner ℓ_{inf}, the Fast Gradient Sign Method (FGSM), and AutoAttack, and three black-box attacks: SquareAttack, HopSkipJump, and PixelAttack. The attacks used were primarily chosen based on their popularity in testing adversarial robustness and, secondly, on the most promising results in early testing stages using pre-trained VGG models extracted from the torchvision module in PyTorch. Additionally, the transferability rate of the adversarial examples between the given models was evaluated to determine whether or not the tested models can be used to create a defense system against the adversary. It should be noted that adversarial defense mechanisms will not be covered in this work.

The contributions of this paper are summarized in the following points:

- We evaluated the robustness of SpinalNet and CCT models against popular targeted and non-targeted attacks;
- We assessed the attack transferability of the non-targeted attacks between VGG, SpinalNet, and CCT.

The paper is organized as follows: Section 2 gives a brief introduction to the generation of attacks and background on attacks used to craft adversarial examples, including four white-box and three black-box attacks extracted from the Adversarial Robustness Toolbox (ART) [22] which is a Python library developed to evaluate and defend machine learning models. Related works are summarized in Section 3. Section 4 provides a background on the neural network architectures utilized to evaluate robustness against adversarial attacks. Section 5 covers the robustness and transferability tests setup. Finally, Section 6 provides results and analysis on the robustness and transferability of the given neural models.

2. Background

2.1. Generation of the Evasion Attacks

Applying an attack during the testing phase helps to evaluate the model's vulnerability, whereas, during the training phase, it helps to improve its robustness and assess attack transferability. Most of the proposed attacks during the testing phase adopted evasion scenarios [13] and were created by adding a malicious input sample to fool the neural network. Given some input data and the corresponding classes $C_1, \ldots, C_k$ with the decision boundaries specified by discriminants $g_i(\cdot)$ where $i = 1, \ldots, k$; the input x belongs to the input class C_i. An adversarial attack is an intentional malicious attempt to replace original input x with an adversarial example x' which leads to the high possibility of misclassification of $C_i(x)$ as a target class $C_t(x')$ [16]. The adversarial examples are input with human-imperceptible perturbation such that the perturbed input $x' = x + \delta$, where δ is the additive adversarial attack [23,24]. The set of possible attacks satisfies the following boundary condition:

$$\underset{j \neq t}{\text{maximize}} \quad \{g_j(x')\} - g_t(x') \leq 0 \tag{1}$$

where $g_t(x')$ is a discriminant value of the target class C_t and $g_j(x')$ is a discriminant of other classes in the classifier.

A similarity between the original input x and modified input x' is quantified using a distance metric $\ell_p = \|x - x'\|_p$, also known as a norm. Each norm refers to a different mode of functioning used by the attacks to alter the original data and create the corresponding adversarial examples [24].

- The ℓ_0 norm is used to indicate the number of features altered to create the added perturbation on the images;
- The ℓ_1 norm indicates the sum of the added perturbations to the crafted examples;
- The ℓ_2 norm specifies the Euclidean distance of the added perturbation to the original image;
- The ℓ_{inf} norm is used to stipulate the maximum added perturbation.

The process of finding attack x' requires solving the optimization problem, which aims to minimize the magnitude of perturbation δ or to minimize the distance between x' and x. Currently, the following most commonly used attacks can be distinguished [23]:

- Minimum norm attack: ensures the minimum magnitude of perturbation;
- Maximum allowable attack: the magnitude of the attack is upper bounded by $\|x - x'\|_p \leq \epsilon$;
- Regularization-based attack: tries to simultaneously minimize two objectives such as $\|x - x'\|_p$ and $max_{j \neq t}\{g_j(x')\} - g_t(x')$.

The attack success rate (ASR) is the metric that shows the proportion of adversarial examples that fooled the model [13]. In some cases, an adversarial example generated to mislead one classifier model can also be transferred to deceive another model. It should be noted the attacks crafted under white-box attack conditions can be transferred to a black-box setting [25]. Sections 2.2 and 2.3 below provide a background on attacks applied in this work.

2.2. White-Box Attacks

2.2.1. Carlini and Wagner

Carlini and Wagner (C&W) proposed three new effective attack algorithms customized to three distance metrics ℓ_0, ℓ_2, and ℓ_{inf} [16]. The initial formulation of finding an adversarial instance can be described as follows:

$$\text{minimize } \mathcal{D}(x, x + \delta) \tag{2}$$

such that $x + \delta \in [0,1]^n$ and $C(x + \delta) = C(x') = C_t$ where C_t is the target label, δ is the additive adversarial attack , $\mathcal{D}$ is some distance metric [16]. Since the formulation in

Equation (2) was difficult to solve, it was updated with an additional constant c with the objective function f:

$$\text{minimize } \mathcal{D}(x, x+\delta) + c \cdot f(x+\delta) \tag{3}$$

such that $x + \delta \in [0,1]^n$ and $c > 0$ [16]. Besides different norms, objective function has a set of constraints Ω and a decision boundary between classes C_i and C_t.

For the targeted Carlini and Wagner attack, the objective function can be expressed as follows:

$$\ell(c, x') = \|x' - x\|_2^2 + c \cdot \ell(x') \tag{4}$$

Our work utilizes two of the attacks based on ℓ_2 and ℓ_{inf}. In a targeted attack, the C&W ℓ_2 attack aims to find the smallest value of c such that $\ell(x') = 0$ with the smallest adversarial perturbation $\|x' - x\|_2^2$. In ART, the value of c is searched using a binary search. The untargeted attack has the only constraint on classes $C(x') \neq C(x)$.

Unlike C&W ℓ_2, the C&W ℓ_{inf} attack works iteratively and operates on batches instead of individual images [26]. This attack aims to find x' such that $\ell(x') = 0$ while $\|x - x'\|_\infty \leq \epsilon$ for a given $\epsilon > 0$ in the ℓ_{inf} norm constraint [22]. It should be noted that the implementation of this attack in ART [22] differs from the originally presented version in [16]. For targeted and untargeted cases, C&W ℓ_{inf} has the same constraints as C&W ℓ_2.

2.2.2. Fast Gradient Sign Method

The Fast Gradient Sign Method (FGSM) adds perturbation with respect to the loss function in a single step and, therefore, is easily implemented [24,27]. The FGSM considers the sign of the gradients of the loss function with respect to the original input image. In the case of the targeted the FGSM attack and ℓ_{inf}, a new adversarial example is created using the perturbation in the following Equation (5) [22]:

$$\psi(x, y) = -\epsilon.\text{sign}\Big(\nabla_x \mathcal{L}(x, y)\Big) \tag{5}$$

where x is the input, y is the target, ϵ is the attack strength [22], and the $\mathcal{L}$ is the loss function. The FGSM attack is mainly chosen for its high efficiency as it only requires one gradient evaluation and can be applied to a batch of inputs instead of singular inputs. For our work, the attack was used in the untargeted default setting in which the original input x is transformed in a way that increases the loss of the classifier when continuing to label x as $C(x)$.

2.2.3. AutoAttack

AutoAttack is an ensemble of parameter-free attacks with low computational cost, which makes it suitable for a minimal test of the robustness of a model [28]. It combines the White-Box Auto Projected Gradient Descent Attack (AutoPGD or APGD), the DeepFool, and the Black-Box SquareAttack attacks. Moreover, it has fixed hyperparameters and is fully automatic [28].

The Projected Gradient Descent (PGD) attack is one of the most popular methods to test the robustness of a model. However, in some cases, it fails due to two potential reasons: (i) a step size and (ii) a cross-entropy loss function. AutoPGD is a gradient-based with an alternative loss function which was designed to overcome limitations of PGD [28]. In particular, in AutoPGD, each iteration N_{iter} is split into exploration and exploitation phases. The aim of the first phase is to identify a set of adequate initial points, and the goal of the next phase is to maximize the accumulated knowledge. Progressive reduction in the step size takes place during the transition between the phases. The DeepFool attack is an untargeted fast but low-accuracy attack [29]. It aims to iteratively search for decision boundaries between classes in the ℓ_2 norm [22]. DeepFool has been adapted for binary and multiclass classifiers.

The description of SquareAttack is provided in Section 2.3.1 below. Moreover, unlike DeepFool and APGD attacks, our work includes an independent evaluation of SquareAttack apart from AutoAttack.

2.3. Black-Box Attacks

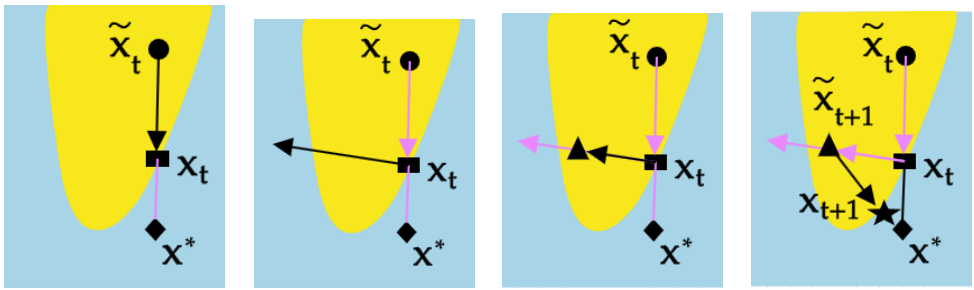
2.3.1. SquareAttack

The SquareAttack attack is a ℓ_2 and ℓ_{inf} score-based attack, and, for a given input, it has access only to the score predicted by a classifier. It performs a random search to select the area to alter. SquareAttack does not exploit any gradient approximation, which makes it safe against gradient masking. The height of the square delimiting the area to alter is the closest positive integer to $\sqrt{p \times w^2}$ with p being the percentage of elements of the original image to modify and w the width of the image of size $w \times w$.

The algorithm of the attack was developed in a way that ensures that the changes made to the original sample are localized. This means that, at each step, only a small fraction of the contiguous square-shaped pixels are modified. The first step in the algorithm is to choose the height of the square to be altered. Then, a random update is computed and added to the previously computed iterate. If the loss function resulting from that addition is better than all previous computed losses, the change is accepted and applied. Otherwise, it is discarded [30].

2.3.2. HopSkipJump

The HopSkipJump attack is a recently developed attack based on a new estimate of the gradient direction using binary information gathered at the decision boundary [31]. It aims to minimize the distance to the target image while staying at the adversarial side of the boundary. The functioning of this attack is best described by an illustration shown in Figure 1 extracted from [31]



(**a**) (**b**) (**c**) (**d**)

Figure 1. Figure showing the intuitive approach to thinking about HopSkipJump: (**a**) a binary search of the boundary; (**b**) the update $\tilde{x_t} \rightarrow x_t$; (**c**) a geometric progression followed by the update $x_t \rightarrow \tilde{x}_{t+1}$; (**d**) a binary search and the update $\tilde{x}_{t+1} \rightarrow x_{t+1}$.

The first step shown in Figure 1 is the binary search to locate the boundary and update the binary point. Then, an estimate of the gradient is obtained at the boundary point, and geometric progression occurs to locate the closest point to the boundary but still in the adversarial region. The identified adversarial point is saved, and then a binary search is performed again to find the second boundary point, and so on [31].

2.3.3. PixelAttack

The PixelAttack is a low-cost and straightforward attack. Instead of small-spread perturbations, it perturbs one pixel:

$$\begin{aligned} \underset{e^{x*}}{\text{maximize}} \quad & f_{adv}(x + e(x)) \\ \text{subject to} \quad & \|e(x)\|_0 \leq d \end{aligned} \tag{6}$$

where d is a small number and equal to 1 for ℓ_0 one-pixel attack. PixelAttack generates adversarial perturbations based on either Differential Evolution (DE) or Covariance Matrix Adaptation (CMA-ES). The corresponding perturbation is then applied to a single pixel in the image to compose the adversarial example [32]. In our research, PixelAttack was applied using the default option set in the attack using CMA-ES.

3. Related Work

Generally, there is no common opinion on how adversarial examples are generated [10]. One group of academic circles believes that a low probability of adversarial examples already exists in real data. Some works [27,33,34] employ the ℓ_0-ℓ_1-ℓ_2 norm to improve image quality and classification accuracy. The other group assumes that the vulnerability of neural networks comes with linear features of the model due to the use of linear activation functions [10]. In addition, early works studied the role of each layer in handling adversarial examples and showed that neural networks could be fooled by compromising certain layers [35,36]. In particular, the most susceptible layers in VGG16 against attacks generated using DLFuzz were *Block*4_*conv*1 and *Block*5_*conv*1 [37].

Attributes of the popular attacks against CNN-based image classification models and their reported strengths were earlier summarized in [38]. According to the presented results, the weakest attack is one-pixel attack, which changes only one pixel of an image [32]. On the contrary, Carlini and Wagner attacks restricted by their ℓ_p-norm [16] are among the strongest. Interestingly, attention blocks added to image classification CNN models increase robustness against adversarial examples [39]. Recent works demonstrated that attention-based transformer models, and Vision Transformer (ViT) in particular, show better performance in computer vision tasks than state-of-the-art CNN models [40]. Moreover, ViT and its hybrid models are more robust than CNN models when different pre-processing methods are applied [21]. Table 1 shows the success rate of attacks applied on different ViT configurations and CNN models without pre-processing steps.

Table 1. The attack success rate (ASR) of the adversarial examples (AEs) for 1000 images from ImageNet-1k [21].

Type of Attack	ASR (%) Top-1 Error			
	FGSM	**C&W ℓ_2**	**C&W ℓ_{inf}**	**PGD ℓ_2**
ViT-S-16	47.80	99.70	100.00	90.00
Vit-B-16	50.00	98.10	95.80	94.10
ViT-L-16	39.80	95.90	94.80	88.40
ResNet50	43.80	99.50	100.00	90.70
VGG-16	89.00	99.90	100.00	98.90

A comprehensive exploration of the robustness and transferability of different CNN architectures was recently published in [41]. This work considered architectures such as VGG-16, VGG-19, ResNet-50, ResNet-152v2, MobileNet v2, EfficientNetB0, and DenseNet201. The results showed that attacks designed for deeper CNNs are effective in fooling CNNs with less depth. For instance, adversarial examples developed using the surrogate DenseNet201 model achieved transferability accuracy on other networks above 90%, whereas an attack designed using VGG-16 achieved transferability accuracy between 15 and 72% on the same models. In [42], the authors tried to understand the transferability between CNNs and ViTs based on the inductive bias phenomenon, which is an ability to generalize to unseen data.

4. Neural Network Models

In this work, the robustness and transferability of various adversarial attacks presented in Section 2 were studied for three types of image classifiers. Since CNN (Figure 2a) and transformer models dominate image classification tasks, the first model type is a

widely spread VGG architecture. The second model of interest is a lightweight compact transformer-based CCT. Initially designed for natural language processing, transformers have rapidly evolved and been used in computer vision [20]. The integration of transformers into the computer vision field started with Vision Transformers (ViTs). However, the need for an excessive amount of data to train ViTs led to the development of their compact version, CCTs, with added sequence pooling and convolutional blocks (Figure 2b) [20]. The third model is SpinalNet, a novel neural network architecture with gradual input that aims to reduce computational overhead during both training and inference [19]. Inspired by the body's somatosensory system, the architecture of SpinalNet consists of three main rows: the input row, the intermediate row, and the output row, as shown in Figure 2c [19]. At the input row, input data are split and repeatedly passed to the intermediate row, which comprises several hidden layers with non-linear activation functions. The number of neurons in those layers is kept small to reduce computations. The output row then sums up the weighted outputs generated by the hidden layers of the intermediate row. The result can be split and passed to the next hidden states if required. The effectiveness of SpinalNet was verified on different configurations of CNN, VGG, ResNet, and DenseNet neural networks for regression and classification problems.

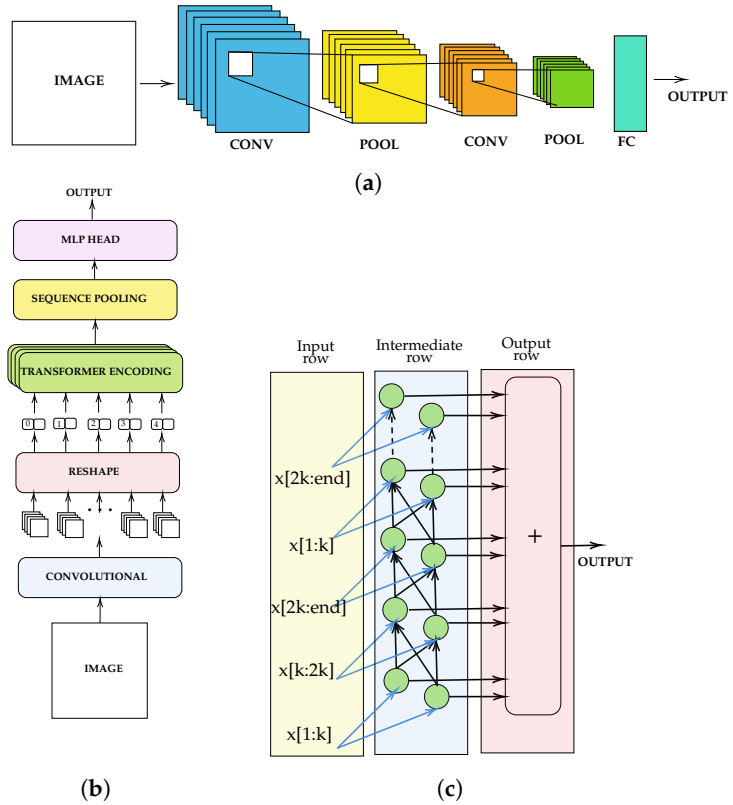


Figure 2. (**a**) CNN architecture, (**b**) SpinalNet architecture, and (**c**) CCT architecture.

5. DNN Model Robustness and Transferability Testing Setup

Analyses of the robustness and transferability of the selected models were conducted using four white-box attacks and three black-box attacks presented in Section 2. The vulnerability of the neural network models and success rate of attacks were studied during the testing phase using evasion or poisoning attacks as summarized in Figure 3. The evaluation was carried out in the PyTorch framework with attacks imported from the Adversarial Robustness Toolbox (ART). The models were evaluated on both untargeted and targeted attacks. The adversarial hyperparameters of each attack were chosen based on the default settings of the ART. Target classes in the case of targeted attacks were chosen randomly.

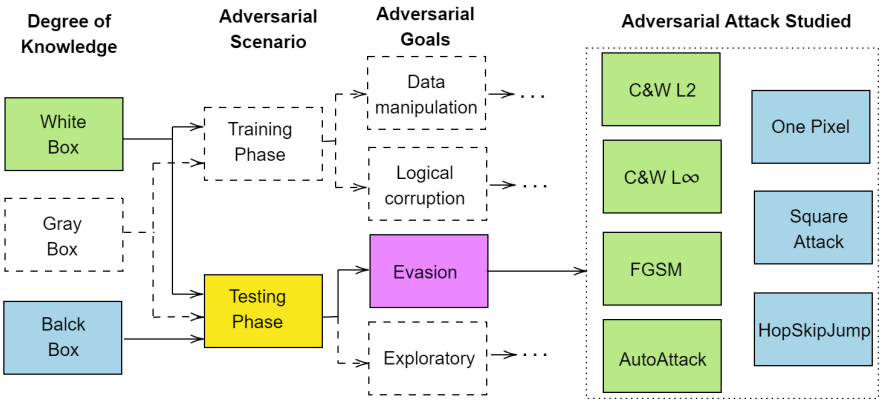


Figure 3. Illustration of the taxonomy of the studied adversarial attacks (adapted from [13]): three black-box (in blue) and four white-box (in green) evasion attacks.

5.1. Data and Models

This work considers three topologies of neural models—SpinalNet, VGG, and CCT. The VGG group comprises VGG-11, VGG-13, VGG-16, and VGG-19. The SpinalNet group includes neural network configurations such as SpinalNet-11, SpinalNet-13, SpinalNet-16, and SpinalNet-19 that are based on VGG models. And the last one is the CCT-7/3 $\times$ 1 configuration.

The classifier models were tested on the widely used CIFAR-10 dataset. In the case of SpinalNet and VGG models, the input shape was 3 $\times$ 32 $\times$ 32, and the dataset values were clipped between (0, 1). In the case of the CCT-7/3 $\times$ 1 model, the CIFAR-10 dataset images had to be normalized before being fed to the model. For normalization, a mean was specified as (0.4914, 0.4822, 0.4465), and the standard deviation as (0.2471, 0.2435, 0.2616). This was performed to acquire the same accuracy mentioned in [20].

5.2. Setup

5.2.1. Testing Robustness

Training and testing of robustness of SpinalNet models and their VGG counterparts on the non-normalized CIFAR-10 dataset were carried out using the codes provided in [43]. The supervised training was performed for over 200 epochs with a learning rate equal to 0.0001. The loss function used to train is the "CrossEntropyLoss" function, and the utilized optimizer is the Adam optimizer in all classifiers. Evaluation of the CCT-7/3 $\times$ 1 model was conducted on the pre-trained model using 5000 epochs [20]. The utilized loss function was the "HingeEmbeddingLoss" function.

5.2.2. Transferability

In this work, the transferability of non-targeted attacks between SpinalNet models and their VGG counterparts and that of both SpinalNet and VGG with CCT-7/3$\times$1 were evaluated (Figure 4). Since crafting adversarial examples is a time-consuming process, transferability tests were conducted on the attacks generated during robustness tests. However, since the data used to train and test the VGG models and their SpinalNet counterparts were not normalized, unlike those used for CCT-7/3 $\times$ 1, we had to normalize the examples crafted on VGG and SpinalNet before transferring them to CCT-7/3 $\times$ 1 using the same values of mean and standard deviation mentioned above and denormalize those crafted on CCT-7/3 $\times$ 1 before transferring them to the VGG and SpinalNet models.

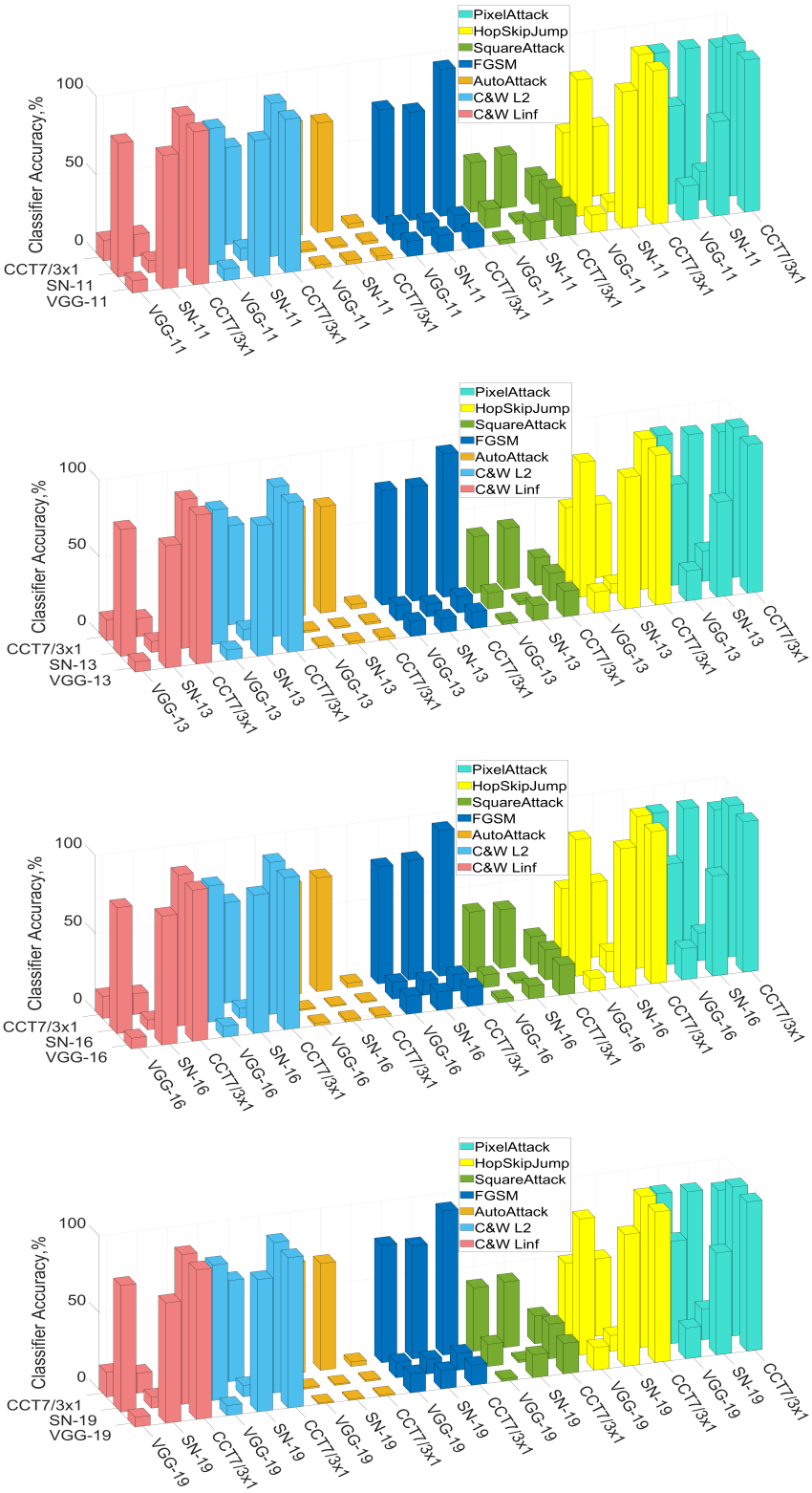


Figure 4. The robustness accuracy and transferability of the following neural network architectures: (**a**) VGG-11, SpinalNet-11, and CCT-7/3 × 1; (**b**) VGG-13, SpinalNet-13, and CCT-7/3 × 1; (**c**) VGG-16, SpinalNet-16, and CCT-7/3 × 1; (**d**) VGG-19, SpinalNet-19, and CCT-7/3×1.

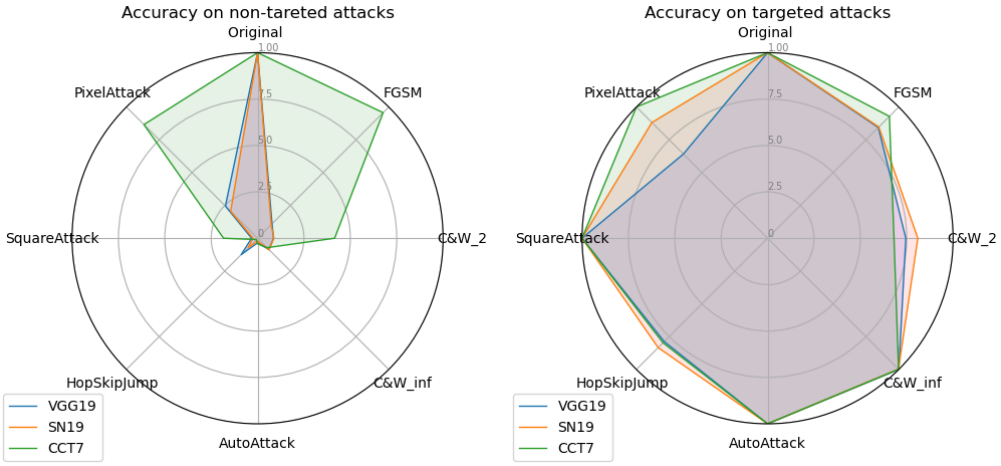
6. Results and Discussion

6.1. Robustness

Evaluation of the robustness of VGG, SpinalNet, and CCT on the same attacks showed that they all are vulnerable to adversarial examples. In the case of non-targeted attacks (more details in Appendix A.1), in a few cases, SpinalNet models appeared to be slightly more robust than their VGG counterparts. But overall, SpinalNet and VGG classifier models respond similarly to all applied attacks. This means that the splits in the SpinalNet architecture do not provide advantages over the fully connected layers in VGG against perturbations, and both models were fooled at early layers and interfered with the performance of subsequent layers. The best performance of SpinalNet and VGG models was observed against PixelAttack (around 20%). The largest accuracy drops were observed for the AutoAttack and the SquareAttack (always below 3%), and the latter, as mentioned earlier, is part of the AutoAttack. This means that CNN features such as multichanneling, weight sharing, downsampling, and locality do not resist attacks in vision tasks.

When comparing our results with those obtained for Vision Transformers (ViTs) in [17], it can be concluded that SpinalNet models show better promise for robustness against adversarial attacks than ViTs. Nevertheless, our work also demonstrates that in most cases, the CCT-7/3×1 model, which also has a convolutional component, appeared to be more robust than SpinalNet and VGG. CCTs are especially good at detecting relatively simple adversarial examples such as the FGSM (93.77%) and PixelAttack (84.89%). For C&W ℓ_{inf} and AutoAttack, the wrapping classifier's accuracy for the CCT model is similar to that of SpinalNet and VGG (below 8% and 3%, respectively). Surprisingly, C&W ℓ_2 attack demonstrated weakness against the CCT model with an accuracy drop of up to 40.7%, whereas in the case of SpinalNet and VGG, its performance was comparable to C&W ℓ_{inf}. In [44], the authors show that despite close adversarial accuracy, the distribution of attacks may differ, and this can explain the sensitivity of CCT. The only attack that appeared to mislead CCT-7/3×1 even more than SpinalNet and VGG is HopSkipJump, which shows the advantage of the binary search in misleading transformers. In fact, with the HopSkipJump attack, when an adversarial example is located at the boundary between adversarial and non-adversarial samples, the accuracy of CCT-7/3×1 dropped from 98.00% to 0.99%. This means CCT can be confused by the closeness of adversarial examples to the decision boundaries of HopSkipJump, which is also the case of the DeepFool attack as part of AutoAttack. Eventually, AutoAttack proved that it generalizes well with any datasets and models.

Figure 5 compares the performance of targeted and untargeted attacks. Targeted attacks were generated for random targets with no masking at the same settings used for non-targeted attacks. As expected, the success rate of untargeted attacks is higher than in the case of targeted attacks since they are more generalizable and do not aim to mislead the network to a specific class. We believe non-targeted attacks better represent the vulnerability of the model. Generally, white-box attacks have more potential than black-box attacks since they possess information on the dataset and model parameters. Nevertheless, recently developed SquareAttack and HopSkipJump black-box attacks demonstrated results comparable with white-box attacks and even better. In addition, the success of AutoAttack shows that an ensemble of features of several adversarial examples strengthens the applied attack. The performance of the models is summarized in Table 2, where the strength of the attack increases with the number of check marks "✓" (between min = 1 and max = 6).



(**a**) (**b**)

Figure 5. Performance of VGG19, SpinalNet-19 (SN-19), and CCT-7×1 neural models on original and attacked data in the following cases (normalized to the original dataset): (**a**) untargeted attacks; (**b**) targeted attacks.

Table 2. Summary of the applied untargeted attacks and model robustness (e.g., neural network classification accuracy: ✓✓✓✓✓✓ 0–5%; ✓✓✓✓✓ 5–10%; ✓✓✓✓ 10–20%; ✓✓✓ 20–40%; ✓✓ 40–80%; ✓ 80–100%.).

Attack Method	Norm	Adversarial Knowledge	Search Type	Crafting Adversarial Example	Strength of the Untargeted Attack		
					SpinalNet	VGG	CCT
FGSM	ℓ_{inf}	White-Box	single step gradient-based guided differentiable search	applies the sign function on gradients	✓✓✓✓✓	✓✓✓✓✓	✓
C&W	ℓ_2	White-box	iterative gradient-based guided differentiable search	attempts to find a small perturbation δ while keeping the ℓ_2 distance between **x** and **x′** minimal	✓✓✓✓✓	✓✓✓✓✓	✓✓
	ℓ_{inf}	White-box	iterative gradient-based guided differentiable search	unlike ℓ_2, operates on batches	✓✓✓✓✓	✓✓✓✓✓	✓✓✓✓✓
SquareAttack	ℓ_{inf}	Black-box	score-based iterative non-guided random search	every iteration selects square-shaped updates so that perturbation is located at the boundary of the feasible set	✓✓✓✓✓✓	✓✓✓✓✓✓	✓✓✓✓
AutoPGD	n/a	White-box	iterative gradient-based guided differentiable search	size-free step with exploration (search for good initial point) and exploitation (knowledge accumulation) phases	n/a	n/a	n/a
DeepFool	n/a	White-box	iterative guided differentiable search	iteratively searches for adversarial example at the decision boundary between classes	n/a	n/a	n/a
AutoAttack	ℓ_{inf}	White-box	iterative without any hyperparameter tuning	ensemble of parameter-free attacks	✓✓✓✓✓✓	✓✓✓✓✓✓	✓✓✓✓✓✓
PixelAttack	ℓ_0	Black-box	iterative differential evolution	only one pixel is changed	✓✓✓✓	✓✓✓✓	✓
HopSkipJump	ℓ_2	Black-box	iterative gradient-based binary search	minimizes distance to the target image while staying at the adversarial side of the boundary	✓✓✓✓✓	✓✓✓✓✓	✓✓✓✓✓✓

6.2. Transferability

Transferable adversarial attacks show the similar features different models share with each other. Regarding behavior, SpinalNet and VGG have similar transferability in both directions (more details in Appendix A.2), which can be explained by the same architecture at early layers. Good transferability was observed in the cases of AutoAttack, the FGSM, and SquareAttack. In the cases of the FGSM and the SquareAttack, accuracy dropped to values ranging from around 8% to around 14.8%, respectively. The AutoAttack could deceive the models and result in values similar to those obtained during the robustness evaluation ranging between 0.7 and 2.5%.

In the case of CCT-7/3×1, transferability was not bidirectional, meaning that some examples were transferable from one model to another but not the other way around. In the case of PixelAttack, the adversarial examples were not transferable either way between VGG and CCT or between SpinalNet and CCT. This introduces the simplest form of perturbations that are not enough to mislead the neural architectures such as CNN and Neural Transformer. In the cases of C&W ℓ_2 and HopSkipJump, the adversarial examples were somewhat transferable from CCT to both SpinalNet and VGG and even more in the case of C&W ℓ_{inf} but not the other way around. In the case of AutoAttack, however, the adversarial examples crafted using the CCT-7/3×1 were partially transferable to SpinalNet and VGG and almost fully transferable the other way around. The SquareAttack examples, on the other hand, were partially transferable both ways. Finally, an interesting result was observed in the case of the FGSM. As mentioned earlier, the CCT-7/3×1 demonstrated the best robustness result against the FGSM attack. However, when examples crafted using the said attack were transferred from surrogate CNN-based models, the CCT-7/3×1 model appeared to be fooled, and the accuracy dropped to between 10 and 13% for all tested cases. The difference in the architectures of CNN and transformer-based neural networks can explain such behavior. But here, we should remember that CCT also has convolutional layers. For instance, CNNs are known for their shift-invariance with input layers designed to extract simple features, and the layers closer to output typically extract high-frequency features. On the contrary, shift-equivariant attention blocks of transformers are responsible for lower-order features [45]. In addition, neural transformers are powerful tools in processing a large volume of spatiotemporal data [46]. Therefore, adversarial perturbations designed for certain architecture have different levels of transferability. If models share similar capturing features, attacks have low transferability.

The results show that despite high effectiveness on certain models independently, some of the attacks have little or no transferability to other models, as in the case of AutoAttack. The potential direction in strengthening attacks and their transferability is the generation of adversarial examples against multiple models at once, as in the case of the Self-Attention Gradient Attack (SAGA) [17]. Another way to enhance transferability is the introduction of the Intermediate Level Attack (ILA) [47]. Overall, similar architectures such as VGG and SpinalNet show similar performance in robustness and transferability. Additionally, in most cases, the transferability is low between SpinalNet and VGG models and not always bidirectional between SpinalNet/VGG and CCT due to differences in capturing input features. This showed that the combination of the tested neural network models that react differently at low- and high-frequency features of data could increase the robustness of the developed system against crafted adversarial examples. However, the order and way the models would be cascaded depend on the type of attack used and perturbation added and could cause a serious security breach. Crafting new attacks that could deceive all these neural architectures will be considered in future work.

7. Conclusions

In this work, the evaluation of robustness and transferability of emerging image classifiers such as SpinalNet and CCT-7 were compared against the well-studied VGG. The study has shown that all models are susceptible to attacks, but the non-targeted attacks have a higher success rate. In addition, due to the same architecture at early layers, VGG

and the SpinalNet architecture exhibit similar performance against targeted and untargeted attacks. Applying the same evasion attacks demonstrated that the convolution-based transformer model CCT-7/3×1 has superior robustness against newly generated and transferable attacks developed based on the CNN architecture. All models are generally more susceptible to ensemble attacks such as SquareAttack and AutoAttack. The evaluation study of cascaded neural models is planned as future work.

Author Contributions: Methodology, M.E.F.; Validation, M.E.F.; Investigation, L.B. and K.S.; Writing—review & editing, K.S., L.B. and M.E.F.; Supervision, R.K. and A.E. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially supported by King Abdullah University of Science and Technology CRG program under grant number: URF/1/4704-01-01.

Data Availability Statement: The data presented in this study are available in this article.

Conflicts of Interest: Author Mohammed E. Fouda was employed by the company Rain Neuromorphics, Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A

Appendix A.1

Table A1. Accuracy of the VGG, SpinalNet (SN), and CCT-7/3×1 (CCT 7) classifier models after applying untargeted attacks.

Attack	Accuracy (%)								
	VGG-11	**VGG-13**	**VGG-16**	**VGG-19**	**SN-11**	**SN-13**	**SN-16**	**SN-19**	**CCT 7**
Baseline	87.96	89.90	89.87	90.25	88.23	88.52	90.71	88.91	98.00
Carlini ℓ_{inf}	7.62	6.57	6.86	6.37	7.69	7.13	6.35	7.21	7.06
Carlini ℓ_2	7.58	6.60	7.01	6.52	7.76	7.35	6.39	7.31	40.70
AutoAttack	2.21	1.77	1.67	0.76	1.44	1.23	0.81	0.82	2.93
FGSM	9.70	9.77	11.57	12.37	9.13	8.37	9.48	9.53	93.77
SquareAttack	2.83	2.14	2.65	1.71	2.19	2.28	1.97	1.60	17.98
HopSkipJump	11.02	13.48	8.00	14.52	6.33	6.16	13.19	10.22	0.99
PixelAttack	21.57	19.62	19.72	19.94	18.10	19.84	17.81	19.89	84.89

Table A2. Accuracy of the VGG-19, SpinalNet-19 (SN-19), and CCT-7/3×1 (CCT 7) classifier models after applying untargeted and targeted attacks.

Attack	Accuracy (%)					
	Untargeted			**Targeted**		
	VGG-19	**SN-19**	**CCT 7**	**VGG-19**	**SN-19**	**CCT 7**
Baseline	90.25	88.91	98.00	90.25	88.91	98.00
Carlini ℓ_{inf}	6.37	7.21	7.06	67.25	73.09	61.50
Carlini ℓ_2	6.52	7.31	40.70	90.25	90.03	90.10
AutoAttack	0.76	0.82	2.93	90.00	89.25	89.55
FGSM	12.37	9.53	93.77	76.18	76.52	83.80
SquareAttack	1.71	1.60	17.98	90.20	90.37	93.13
HopSkipJump	14.52	10.22	0.99	71.08	72.25	71.92
PixelAttack	19.94	19.89	84.89	57.90	79.58	90.01

Appendix A.2

Table A3. Transferability results between SpinalNet (SN) and VGG.

Attack	Accuracy (%)							
	SN-11 ↔ VGG-11		**SN-13 ↔ VGG-13**		**SN-16 ↔ VGG-16**		**SN-19 ↔ VGG-19**	
Carlini ℓ_{inf} (%)	85.19	85.39	79.95	82.96	82.99	81.11	78.25	82.20
Carlini ℓ_2 (%)	86.98	86.98	85.47	87.92	89.10	87.72	85.99	88.16
AutoAttack (%)	2.53	1.45	1.77	1.24	1.68	0.81	0.77	1.09
FGSM (%)	10.59	10.35	9.23	9.99	11.83	10.61	11.94	10.02
SquareAttack (%)	11.65	12.07	9.85	10.54	8.08	8.07	14.82	14.38
HopSkipJump (%)	87.02	86.99	85.99	88.10	89.32	87.90	85.70	88.46
PixelAttack (%)	60.19	62.26	62.10	65.83	64.42	64.59	66.25	66.52

Table A4. Transferability results between VGG and CCT-7/3×1 (CCT 7).

Attack	Accuracy (%)							
	VGG-11 ↔ CCT 7		**VGG-13 ↔ CCT 7**		**VGG-16 ↔ CCT 7**		**VGG-19 ↔ CCT 7**	
Carlini ℓ_{inf}	13.08	97.78	13.70	97.48	14.08	97.24	15.88	97.18
Carlini ℓ_2	60.86	97.90	64.47	97.84	62.94	97.85	65.80	97.78
AutoAttack	72.82	2.77	70.86	2.17	71.87	1.74	72.12	0.93
FGSM	73.13	10.57	74.97	10.72	75.37	12.56	76.24	12.97
SquareAttack	31.72	19.16	37.08	16.64	38.77	19.21	41.23	19.83
HopSkipJump	43.04	97.90	48.20	97.91	46.43	97.82	49.72	97.85
PixelAttack	86.17	97.06	87.80	97.10	87.66	96.88	87.69	96.95

Table A5. Transferability results between SpinalNet and CCT-7/3×1 (CCT 7).

	SN-11 ↔ CCT 7		**SN-13 ↔ CCT 7**		**SN-16 ↔ CCT 7**		**SN-19 ↔ CCT 7**	
Carlini ℓ_{inf} (%)	14.14	97.73	11.81	97.40	13.32	97.22	12.78	97.40
Carlini ℓ_2 (%)	62.02	97.90	64.91	97.93	64.59	97.85	65.73	97.88
AutoAttack (%)	69.95	1.83	69.59	1.46	73.14	0.91	69.45	0.99
FGSM (%)	69.00	10.57	74.83	10.64	77.01	10.71	73.41	10.98
SquareAttack (%)	33.78	20.37	39.95	18.10	37.84	19.23	42.54	22.45
HopSkipJump (%)	44.31	97.90	47.76	97.92	48.02	97.83	50.20	97.93
PixelAttack (%)	86.34	97.20	85.77	96.87	88.14	97.28	86.47	96.88

References

1. Sultana, F.; Sufian, A.; Dutta, P. Advancements in image classification using convolutional neural network. In Proceedings of the 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 22–23 November 2018; pp. 122–129.
2. Han, S.; Kang, J.; Mao, H.; Hu, Y.; Li, X.; Li, Y.; Xie, D.; Luo, H.; Yao, S.; Wang, Y.; et al. Ese: Efficient speech recognition engine with sparse lstm on fpga. In Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Monterey, CA, USA, 22–24 February 2017; pp. 75–84.
3. Bhandare, A.; Sripathi, V.; Karkada, D.; Menon, V.; Choi, S.; Datta, K.; Saletore, V. Efficient 8-bit quantization of transformer neural machine language translation model. *arXiv* **2019**, arXiv:1906.00532.
4. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.

5. Moosavi-Dezfooli, S.M.; Fawzi, A.; Fawzi, O.; Frossard, P. Universal adversarial perturbations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1765–1773.
6. Michel, A.; Jha, S.K.; Ewetz, R. A survey on the vulnerability of deep neural networks against adversarial attacks. *Prog. Artif. Intell.* **2022**, *11*, 131–141. [CrossRef]
7. Lin, X.; Zhou, C.; Wu, J.; Yang, H.; Wang, H.; Cao, Y.; Wang, B. Exploratory adversarial attacks on graph neural networks for semi-supervised node classification. *Pattern Recognit.* **2023**, *133*, 109042. [CrossRef]
8. Kaviani, S.; Han, K.J.; Sohn, I. Adversarial attacks and defenses on AI in medical imaging informatics: A survey. *Expert Syst. Appl.* **2022**, *198*, 116815. [CrossRef]
9. Wu, H.; Yunas, S.; Rowlands, S.; Ruan, W.; Wahlström, J. Adversarial driving: Attacking end-to-end autonomous driving. In Proceedings of the 2023 IEEE Intelligent Vehicles Symposium (IV), Anchorage, AK, USA, 4–7 June 2023; pp. 1–7.
10. Liang, H.; He, E.; Zhao, Y.; Jia, Z.; Li, H. Adversarial attack and defense: A survey. *Electronics* **2022**, *11*, 1283. [CrossRef]
11. Shafahi, A.; Huang, W.R.; Studer, C.; Feizi, S.; Goldstein, T. Are adversarial examples inevitable? *arXiv* **2018**, arXiv:1809.02104.
12. Carlini, N.; Athalye, A.; Papernot, N.; Brendel, W.; Rauber, J.; Tsipras, D.; Goodfellow, I.; Madry, A.; Kurakin, A. On evaluating adversarial robustness. *arXiv* **2019**, arXiv:1902.06705.
13. Khamaiseh, S.Y.; Bagagem, D.; Al-Alaj, A.; Mancino, M.; Alomari, H.W. Adversarial deep learning: A survey on adversarial attacks and defense mechanisms on image classification. *IEEE Access* **2022**, *10*, 102266–102291. [CrossRef]
14. Huang, S.; Jiang, H.; Yu, S. Mitigating Adversarial Attack for Compute-in-Memory Accelerator Utilizing On-chip Finetune. In Proceedings of the 2021 IEEE 10th Non-Volatile Memory Systems and Applications Symposium (NVMSA), Beijing, China, 18–20 August 2021; pp. 1–6.
15. Rathore, P.; Basak, A.; Nistala, S.H.; Runkana, V. Untargeted, targeted and universal adversarial attacks and defenses on time series. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
16. Carlini, N.; Wagner, D. Towards evaluating the robustness of neural networks. In Proceedings of the 2017 IEEE Symposium on Security and Privacy, San Jose, CA, USA, 22–26 May 2017; pp. 39–57.
17. Mahmood, K.; Mahmood, R.; Van Dijk, M. On the robustness of vision transformers to adversarial examples. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 7838–7847.
18. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
19. Kabir, H.; Abdar, M.; Jalali, S.M.J.; Khosravi, A.; Atiya, A.F.; Nahavandi, S.; Srinivasan, D. Spinalnet: Deep neural network with gradual input. *arXiv* **2020**, arXiv:2007.03347.
20. Hassani, A.; Walton, S.; Shah, N.; Abuduweili, A.; Li, J.; Shi, H. Escaping the big data paradigm with compact transformers. *arXiv* **2021**, arXiv:2104.05704.
21. Aldahdooh, A.; Hamidouche, W.; Deforges, O. Reveal of vision transformers robustness against adversarial attacks. *arXiv* **2021**, arXiv:2106.0373 .
22. Nicolae, M.I.; Sinn, M.; Tran, M.N.; Buesser, B.; Rawat, A.; Wistuba, M.; Zantedeschi, V.; Baracaldo, N.; Chen, B.; Ludwig, H.; et al. Adversarial Robustness Toolbox v1. 0.0. *arXiv* **2018**, arXiv:1807.01069.
23. Wu, F., Gazo, R., Haviarova, E. and Benes, B. Efficient project gradient descent for ensemble adversarial attack. *arXiv* **2019** arXiv:1906.03333.
24. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* **2014**, arXiv:1412.6572.
25. Gil, Y.; Chai, Y.; Gorodissky, O.; Berant, J. White-to-black: Efficient distillation of black-box adversarial attacks. *arXiv* **2019**, arXiv:1904.02405.
26. Mani, N. On Adversarial Attacks on Deep Learning Models. Master's Thesis, San Jose State University, San Jose, CA, USA, 2019.
27. Liu, X.; Wang, H.; Zhang, Y.; Wu, F.; Hu, S. Towards efficient data-centric robust machine learning with noise-based augmentation. *arXiv* **2022**, arXiv:2203.03810.
28. Croce, F.; Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In Proceedings of the International Conference on Machine Learning, (ICML), Vienna, Austria, 13–18 July 2020; pp. 2206–2216.
29. Croce, F.; Hein, M. Minimally distorted adversarial examples with a fast adaptive boundary attack. In Proceedings of the International Conference on Machine Learning, (ICML), Vienna, Austria, 13–18 July 2020; pp. 2196–2205.
30. Andriushchenko, M.; Croce, F.; Flammarion, N.; Hein, M. Square attack: A query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision (ECCV)*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 484–501.
31. Chen, J.; Jordan, M.I.; Wainwright, M.J. Hopskipjumpattack: A query-efficient decision-based attack. In Proceedings of the 2020 Ieee Symposium on Security and Privacy (sp), San Francisco, CA, USA, 18–21 May 2020; pp. 1277–1294.
32. Su, J.; Vargas, D.V.; Sakurai, K. One pixel attack for fooling deep neural networks. *IEEE Trans. Evol. Comput.* **2019**, *23*, 828–841. [CrossRef]
33. Liu, Y.; Yan, Z.; Tan, J.; Li, Y. Multi-purpose oriented single nighttime image haze removal based on unified variational retinex model. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 1643–1657. [CrossRef]
34. Liu, Y.; Yan, Z.; Ye, T.; Wu, A.; Li, Y. Single nighttime image dehazing based on unified variational decomposition model and multi-scale contrast enhancement. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105373. [CrossRef]
35. Bakiskan, C.; Cekic, M.; Madhow, U. Early layers are more important for adversarial robustness. In Proceedings of the ICLR 2022 Workshop on New Frontiers in Adversarial Machine Learning, (ADVML Frontiers @ICML), Baltimore, MD, USA, 17–23 July 2022.

36. Siddiqui, S.A.; Breuel, T. Identifying Layers Susceptible to Adversarial Attacks. *arXiv* **2021**, arXiv:2107.04827.
37. Renkhoff, J.; Tan, W.; Velasquez, A.; Wang, W.Y.; Liu, Y.; Wang, J.; Niu, S.; Fazlic, L.B.; Dartmann, G.; Song, H. Exploring adversarial attacks on neural networks: An explainable approach. In Proceedings of the 2022 IEEE International Performance, Computing, and Communications Conference (IPCCC), Austin, TX, USA, 11–13 November 2022; pp. 41–42.
38. Akhtar, N.; Mian, A. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access* **2018**, *6*, 14410–14430. [CrossRef]
39. Agrawal, P.; Punn, N.S.; Sonbhadra, S.K.; Agarwal, S. Impact of attention on adversarial robustness of image classification models. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 3013–3019.
40. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
41. Álvarez, E.; Álvarez, R.; Cazorla, M. Exploring Transferability on Adversarial Attacks. *IEEE Access* **2023**, *11*, 105545–105556. [CrossRef]
42. Chen, Z.; Xu, C.; Lv, H.; Liu, S.; Ji, Y. Understanding and improving adversarial transferability of vision transformers and convolutional neural networks. *Inf. Sci.* **2023**, *648*, 119474. [CrossRef]
43. Spinalnet: Deep neural network with gradual input. *IEEE Trans. on Artif. Intell.* **2022**, *4*, 1165–1177
44. Kotyan, S.; Vargas, D.V. Adversarial Robustness Assessment: Why both L_0 and L_∞ Attacks Are Necessary. *arXiv* **2019**, arXiv:1906.06026.
45. Benz, P.; Ham, S.; Zhang, C.; Karjauv, A.; Kweon, I.S. Adversarial robustness comparison of vision transformer and mlp-mixer to cnns. *arXiv* **2021**, arXiv:2110.02797.
46. Yan, C.; Chen, Y.; Wan, Y.; Wang, P. Modeling low-and high-order feature interactions with FM and self-attention network. *Appl. Intell.* **2021**, *51*, 3189–3201. [CrossRef]
47. Huang, Q.; Katsman, I.; He, H.; Gu, Z.; Belongie, S.; Lim, S.N. Enhancing adversarial example transferability with an intermediate level attack. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 4733–4742.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.