applied sciences

MDPI

Article

Improving Adversarial Robustness via Distillation-Based Purification

Inhwa Koo [1], Dong-Kyu Chae [1,*] and Sang-Chul Lee [2,*]

[1] Department of Artificial Intelligence, Hanyang University, Seoul 04763, Republic of Korea; ihkoo@hanyang.ac.kr
[2] Division of Nanotechnology, Daegu Gyeongbuk Institute of Science & Technology (DGIST), Deagu 42988, Republic of Korea
* Correspondence: dongkyu@hanyang.ac.kr (D.-K.C.); sangchul.lee@dgist.ac.kr (S.-C.L.)

Abstract: Despite the impressive performance of deep neural networks on many different vision tasks, they have been known to be vulnerable to intentionally added noise to input images. To combat these adversarial examples (AEs), improving the adversarial robustness of models has emerged as an important research topic, and research has been conducted in various directions including adversarial training, image denoising, and adversarial purification. Among them, this paper focuses on adversarial purification, which is a kind of pre-processing that removes noise before AEs enter a classification model. The advantage of adversarial purification is that it can improve robustness without affecting the model's nature, while another defense techniques like adversarial training suffer from a decrease in model accuracy. Our proposed purification framework utilizes a Convolutional Autoencoder as a base model to capture the features of images and their spatial structure. We further aim to improve the adversarial robustness of our purification model by distilling the knowledge from teacher models. To this end, we train two Convolutional Autoencoders (teachers), one with adversarial training and the other with normal training. Then, through ensemble knowledge distillation, we transfer the ability of denoising and restoring of original images to the student model (purification model). Our extensive experiments confirm that our student model achieves high purification performance(i.e., how accurately a pre-trained classification model classifies purified images). The ablation study confirms the positive effect of our idea of ensemble knowledge distillation from two teachers on performance.

Keywords: adversarial robustness; adversarial attacks; adversarial purification; knowledge distillation; image classification; convolutional autoencoders

Citation: Koo, I.; Chae, D.-K.; Lee, S.-C. Improving Adversarial Robustness via Distillation-Based Purification. *Appl. Sci.* **2023**, *13*, 11313. https://doi.org/10.3390/app132011313

Academic Editor: Donato Cascio

Received: 22 September 2023
Revised: 12 October 2023
Accepted: 12 October 2023
Published: 15 October 2023

1. Introduction

Deep Neural Networks have achieved promising performances in many domains including computer vision [1,2], natural language processing [3,4] and medical image analysis [5,6]. However, there have been a lot of adversarial attacks that can fool the deep learning models [7]. Adversarial robustness is thus critical in real-world scenarios because deep learning models, when deployed in practical applications, can be vulnerable to such maliciously crafted inputs (i.e., adversarial examples) designed to deceive or mislead them. These adversarial attacks can have severe consequences, especially in safety-critical systems such as autonomous vehicles, medical diagnosis, or financial systems. Ensuring adversarial robustness is critical to maintaining the integrity, safety, and reliability of AI-driven systems in diverse real-world environments.

Consequently, there has been an active research effort to improve the adversarial robustness of recent neural models. In the field of computer vision, adversarial examples (AEs) are obtained by perturbing the original image to introduce small noises that are difficult to discern by human eyes. Adversarial attacks are designed to cause misclassification of the model by creating these AEs, and adversarial defenses are designed to make the

model more robust so that it can classify well even when these AEs are mixed into the input. There are many types of adversarial attacks, and one popular technique is to add noise to an image based on gradients (FSGM, PGD, etc.) [8–10]. Other methods include generating AEs that minimize a loss function over the input [11], changing only one of the most critical pixels in the image [12], and combining multiple methods of creating AEs [13]. The effectiveness of the attack usually depends on the value of the parameter ϵ, which controls the amount of noise added to images.

Adversarial defense strategies to combat these attacks have also been actively studied. Representative areas include adversarial training [8], which uses AEs together to train a model, and image denoising, which tries to remove noise from the input AEs [14,15]. Image denoising aims to restore the AEs as close as possible to the original image by removing the noise in the image, and among them, adversarial purification aims to remove the noise by assuming that the noise is definitely an adversarial purturbation caused by adversarial attacks. However, both methods have limitations in that their robustness performance decreases with different types of attacks, and their accuracy for normal inputs decreases.

In this paper, we propose a novel purification technique that can improve adversarial robustness. The main idea of the proposed method is to transfer the knowledge of two Convolutional Autoencoder [16] models (one with adversarially trained and the other with normally trained) to a student model through ensemble knowledge distillation [17]. Convolutional Autoencoder is an image-friendly structure that replaces the multi-layer perceptron (MLP) with Convolutional layers in the original MLP-based Autoencoder. It has shown good performance in image restoration and generation because it can capture the local features and spatial structures of images better than MLP. Using this structure as our base model, the knowledge of the adversarially trained teacher model (AT) and the normally trained teacher model (NT) is transferred to the student (purification) model by ensemble knowledge distillation, where the ability to remove the added noise is learned from the AT teacher and the ability to restore the features of the original image is learned from the NT teacher.

We measure the performance of the proposed purifier on a widely utilized benchmark dataset. Specifically, the purified images were fed into a pre-trained classification model to evaluate whether it can accurately predict the class; the better the purification ability, the higher the classification accuracy of the model. The experimental results show that the proposed purifier can indeed prevent accuracy degradation when classifying the original image, and is robust to both the attacks used in training and the attacks not used in training. An ablation study was conducted to verify the effectiveness of the teacher models used in knowledge distillation, and the results showed that the student model using both teacher models as proposed outperformed the other alternatives.

The rest part of this paper is organized as follows. In Section 2, we highlight existing methodologies regarding adversarial training and adversarial purification. In Section 3, we introduce our novel adversarial purification method. In Section 4, we report the experimental settings and results, demonstrating its efficacy and superiority. Finally, Section 5 concludes our study.

2. Related Work on Adversarial Defense

This chapter introduces two representative approaches in the context of adversarial defense: adversarial training and adversarial purification. Our work is in line with the latter category.

2.1. Adversarial Training

Adversarial Training involves training a model on adversarial examples (AEs) along with the normal training data. The idea is to expose the model to adversarial attacks during

training, so it can learn to resist them. Formally, adversarial attacks manipulate an original image x into the adversarial example x' using the following method:

$$x' = x + \delta \quad s.t. \ \|\delta\|_\infty \leq \epsilon \tag{1}$$

where δ indicates the adversarial noise to be injected. The strength of the attack is controlled by ensuring that the L_∞ norm of the noise does not exceed a hyper-parameter, ϵ. The noise introduces subtle changes to the original image that are imperceptible to the human eye [18]. Various adversarial attacks have been developed over the years. For example, Fast Gradient Sign Method (FGSM) [8] creates AEs by adding a perturbation in the direction of the gradient of the loss with respect to the input data. Projected Gradient Descent (PGD) [9] is an iterative version of FGSM, which applies the perturbation step multiple times, each time projecting the adversarial example back into a valid input space. A Carlini and Wagner (CW) [11] attack is a more sophisticated optimization-based approach that aims to find the smallest perturbation necessary to induce misclassification, often resulting in more subtle changes and thus challenging AEs than the aforementioned two methods. In addition, it minimizes the distance of the original image from the corresponding AE, making it more likely to be misclassified using a distance function, such as L_0, L_2, L_∞, as an objective.

AdvProp [19] enhanced robustness of the model by adversarial training using a mini-batch consisting solely of normal data as well as a supplementary minibatch consisting of PGD-generated AEs. The AEs in the supplementary minibatch have different underlying distributions than normal examples, which helps to mitigate the issue of distribution mismatch and makes it easier for the model to learn valuable features from both clean and adversarial domains. Robust Contrastive Learning (RoCL) [20] proposes a novel adversarial training approach without the need for labeled data. It uses instance-wise adversarial attacks and a contrastive learning framework to maximize the similarity between transformed examples and their adversarial perturbations. Ref. [21] explores adversarial training with imperfect supervision, specifically with complementary labels (CLs), and proposes a new learning strategy using gradually informative attacks to address the challenges of this setting. The authors aim to reduce the performance gap between adversarial training with ordinary labels and CLs (such as noisy or partial labels).

2.2. Adversarial Purification

Adversarial purification is a preprocessing technique that removes noise before the classification model receives input images, resulting in clean images. It does not require model modification or additional training, preserving the unique features and performance of each model. The concept of adversarial purification was first introduced by the authors of PixelDefend [22]. This method trains a PixelCNN [23] as a purifier by making small changes to input images to return AEs to the distribution of original dataset. However, because PixelDefend makes changes at the pixel level of images, which involves pixel-by-pixel operations which, in turn, increases computational overhead. The authors of [24] propose to improve the purification performance by training an Energy-Based Model (EBM) with a score function trained by Denoising Score-Matching (DSM).

Purification based on Generative Adversarial Nets (GAN) [25] has also been studied to purify AEs by training a generator to remove noise and a discriminator to distinguish the purified images produced by the generator from original images [26,27]. However, the training of GANs is inherently unstable, and there are vulnerabilities in the latent space that can be exploited by adversarial attacks to produce wrong images [28]. NRP [29] uses a similar idea to GANs to train a purifier. The purified image is passed through a "critic" network, which acts as a discriminator, and a feature extractor. The loss of the feature extractor is defined as the distance between the AEs and the original images. It is trained to minimize the loss of the critic network as well as to maximize the loss of the feature extractor, and noise is generated based on the loss of the feature extractor and added to the input image. SOAP [30] simultaneously performs the main task of classification and auxiliary tasks to train a purifier, where the auxiliary tasks include some widely-used tasks

in self-supervised learning, such as data reconstruction and rotation prediction. Other works used autoencoders and VAEs [31] to remove noise [32–34], and employed diffusion models to clean up AEs [35].

3. Method

The overview of the proposed framework for learning purifiers based on knowledge distillation is as follows. First, we train two Convolutional Autoencoder-based teacher models with the same structure. One is trained by adversarial training and the other by normal training using original images. The knowledge of the teacher models is then distilled in an ensemble fashion to the purifier (student model). After the knowledge is transferred, the purifier cleans the images affected by various adversarial attacks, and then classifies the purified images with a pre-trained classification model (ResNet56). This classification result is compared to the classification result of the corresponding original image. The closer the results match, the better the purification.

3.1. Base Model: Convolutional Autoencoders

In our work, the two teacher models and the student model in the knowledge distillation framework are based on the same Convolutional Autoencoder structure (see Figure 1). An Autoencoder is an encoder–decoder neural structure that compresses the input through the encoder and restores it to its original dimension through the decoder. The bottleneck layer between the encoder and decoder has a low-dimensional latent representation that retains important features of the original input. The decoder aims to produce an output that is as close as possible to the original input based on this latent. Autoencoders have been widely used for tasks such as data generation, super resolution, and data restoration. We believe that autoencoders are also well suited to the task of purification, which is the task of restoring images by removing noise from AEs.

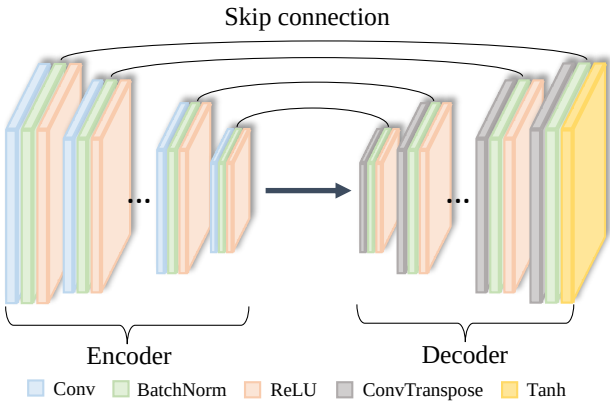

Figure 1. The structure of the Convolutional Autoencoder used in our work, inspired by [34]. The colors of the layers have the following meanings. Blue: Convolutional layer, green: batch normalization layers, orange: ReLU operations, gray: transposed Convolutional layers, yellow: Hyperbolic tangent operations.

Furthermore, Convolutional Autoencoders are specialized in dealing with image data. Instead of a fully connected network (FCN), a Convolutional layer with local connections is mainly utilized, which can better learn the spatial features of images. Here, the encoder consists of a series of Convolution, batch normalization, and Rectified Linear Unit (ReLU) layers in one block, for a total of 15 blocks. The Convolution layer extracts various features, colors, textures, etc., from images, while the batch normalization layer keeps the distributions within a batch consistent for stable learning. The ReLU activation function mitigates the problem of gradient vanishing. The decoder also consists of 15 blocks, each of which consists of a series of Convolutional Transpose, batch normalization, and ReLU operations. A tangent hyperbolic (Tanh) operation is added to the end of the last block. The latent

representation is upsampled by the Convolutional Transpose operation to decode as close to the input image as possible.

Our Convolutional Autoencoder is quite deep, with a total of 30 blocks. This has the advantage of learning a good quality of latent representation, but because of its depth, there is a risk that the gradient may vanish or explode during backpropagation. To avoid this, we make skip connections at the encoder and decoder to convey the gradient flow directly. This also has the effect of helping the decoder to reconstruct images by preventing the loss of information or details that are useful for reconstruction. As a result, the network structure of this study is similar to that of U-net [36].

3.2. Teacher Models

Next, we describe the training of two teacher models, as depicted in Figure 2. The teacher models are of the same Convolutional Autoencoder structure, but one is trained adversarially using the PGD attack (AT teacher model) and the other is trained using the original image (NT teacher model).

The objective function L_{AT} for training the AT teacher model consists of two loss terms, L_p and L_{adv}, as follows:

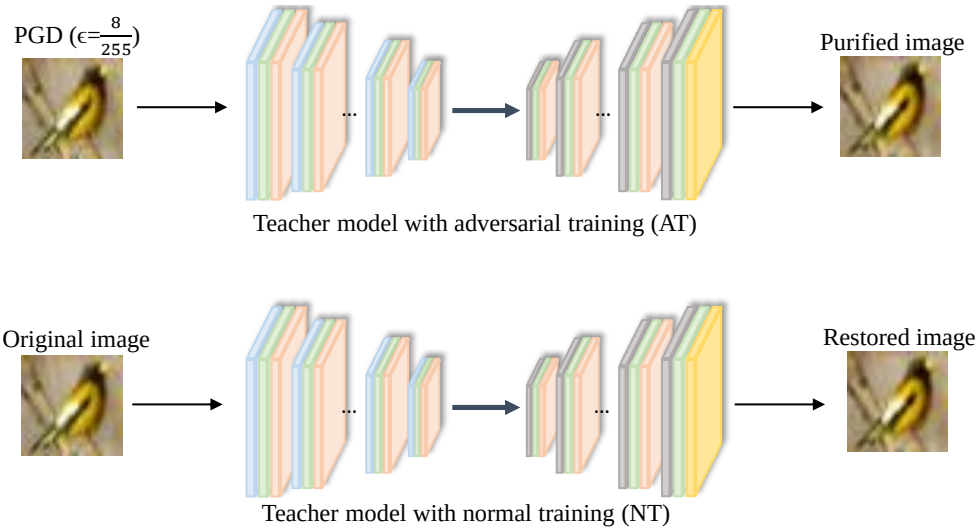

Figure 2. Training of the two teacher models, one with adversarial training (NT) using the PGD attack and the other with the normal training (NT) with original images. The ϵ value for PGD is set to $\frac{8}{255}$.

$$\begin{aligned} L_{AT} &= L_p + L_{adv} = MSE(f_{AT}(x'),\, x) - \log(\sigma(p_{out} - adv_{out})) \\ &= \frac{1}{n}\sum_{i=1}^{n}(f_{AT}(x')_i,\, x_i)^2 - \log\left(\frac{1}{1+e^{-(p_{out}-adv_{out})}}\right) \end{aligned} \tag{2}$$

Here, n is the number of pixels in an image. L_p computes the Mean Squared Error (MSE) of the original image x and the purified image $f_{AT}(x')$ (x' is the adversarial example). L_{adv} is the adversarial loss function, where p_{out} and adv_{out} denote the output of the classification model with the purified image $f_{AT}(x')$ and with the adversarial sample x', respectively. These two outputs should be maximized while minimizing the MSE term for training the AT teacher model.

Next, the NT teacher model is trained to minimize the difference between the restored image and the original image. For this, the loss function L_{NT} uses the mean square error as shown below:

$$\begin{aligned} L_{NT} &= MSE(f_{NT}((x),\, x) \\ &= \frac{1}{n}\sum_{i=1}^{n}(f_{NT}(x)_i - x_i)^2 \end{aligned} \tag{3}$$

where $f_{NT}(x)$ is the image restored by the NT teacher model.

As a result, the AT teacher model learns to remove noise by restoring the original images from the adversarial images, and the NT teacher model learns to restore original images by extracting the important features. The respective abilities of the two teachers are distilled to a purifier (student model).

3.3. Training Purifier via Knowledge Distillation

We next build a purifier model as a student to distill the knowledge of the two previously trained teacher models. The purifier also uses a Convolutional Autoencoder with the same structure as the teacher models. Overall, The purifier learns to effectively denoise the AEs, just like the AT teacher model. The purifier is trained based on the ensemble knowledge distillation framework: by distilling the restoration ability of the NT teacher model to the purifier, we hope that the denoised part of the purified image will be restored similar to the original image; by distilling the denoising ability of the AT teacher model to the purifier, we expect the purifier's denoising ability to be improved.

Figure 3 depicts our process of learning a purifier based on knowledge distillation. As we introduced, the AT teacher model is given AEs generated by the PGD attack to purify them, and the NT teacher model is given pure images to restore them. The purifier takes AEs (each denoted by x') as input and tries to remove the noise. Then, the difference between the purified image $f_s(x')$ and the original image x is defined as the reconstruction loss function L_s for our purifier, which is computed as follows:

$$\begin{aligned} L_s &= MSE\big((f_s(x'),\, x\big) \\ &= \frac{1}{n}\sum_{i=1}^{n}(f_s(x')_i,\, x_i)^2 \end{aligned} \tag{4}$$

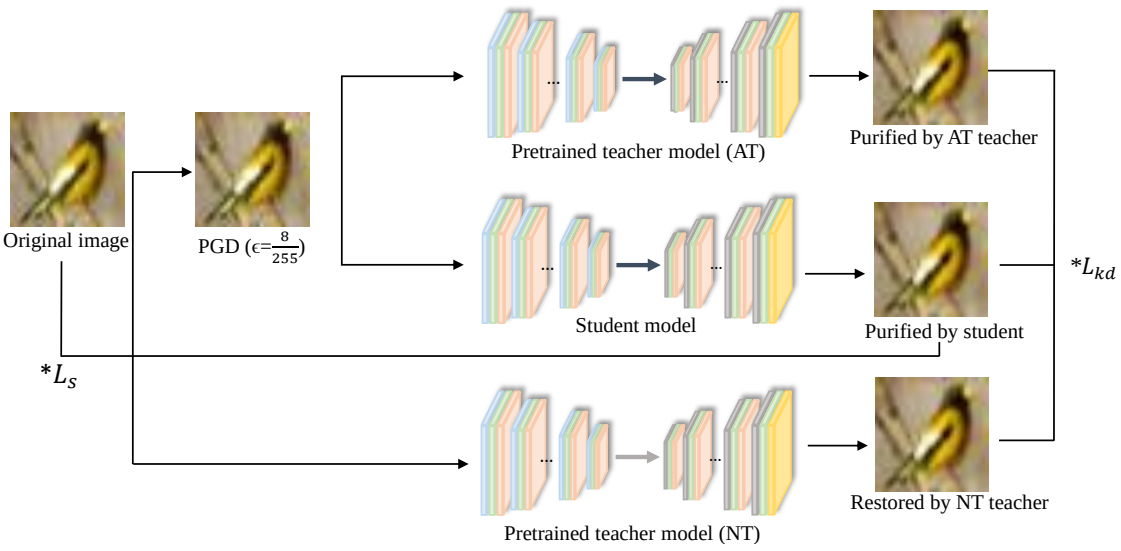

Figure 3. The proposed framework for learning a purification model based on knowledge distillation.

Another loss function of our purifier, the ensemble knowledge distillation loss function L_{kd}, consists of the Kullback–Leibler distance and the mean square error between the outputs of multiple teacher models f_{T_j} and student models f_s, as follows:

$$L_{kd} = \frac{1}{M}\sum_{j=1}^{M}\Big(KL\Big(g\Big(f_{T_j}(x')\Big),\, g(f_s(x'))\Big) + MSE\Big(f_{T_j}(x'),\, f_s(x')\Big)\Big) \tag{5}$$

where M is the number of teacher models (in our case, $M = 2$) and T_j is each teacher model. The Kullback–Leibler divergence function $KL()$ computes the difference between

the output distribution of the teacher model and that of the student model, where each probability distribution is computed by the Softmax function g.

As a result, the purifier learns the denoising ability and image restoration ability of the two teacher models, respectively, and is simultaneously optimized by the Kullback–Leibler divergence and the mean square error, which can reduce both the distribution difference between the student and teacher models and the output image difference. The final loss function L for training the purifier is configured as follows:

$$L = \beta \cdot L_s + \gamma \cdot L_{kd} \tag{6}$$

where β and γ control the importance of the reconstruction loss L_s and the knowledge distillation loss L_{kd}, respectively. For simplicity, we assume that the two loss terms have equal importance and set $\beta = \gamma = 0.5$.

3.4. Purification Process

Figure 4 depicts the overall purification process. After training with ensemble knowledge distillation, the purifier is able to cleanse the AEs generated by various attacks (in our experiments, we used a variety of attacks that the student model has not encountered, including FSGM, BIM, CW, and AutoAttack, in addition to the PGD attacks used in the training of the AT teacher model.) to output purified images (see Figure 5 for an example of images actually purified by our method). We feed the purified images into a pre-trained classification model (ResNet [37] is used) to classify them. If the classification result is the same as the classification result of the corresponding original image, we can say that the purification is successful.

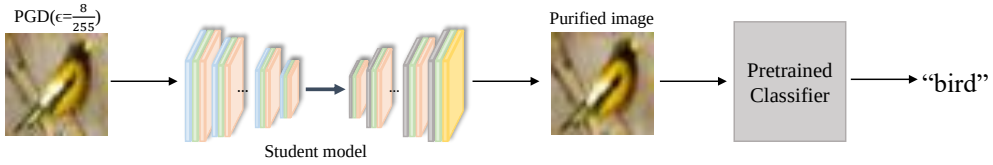

Figure 4. Purification process. A pre-trained ResNet56 [37] was used as a classification model.

(a) Original images.

(b) Adversarial images by PGD.

(c) Purified images.

Figure 5. From top to bottom, we show the original images, the adversarial examples generated by the PGD attack, and the purified images using our proposed method. We can see that the noise has been well removed from the purified images.

4. Evaluation

4.1. Settings

We used CIFAR-10 (https://www.cs.toronto.edu/~kriz/cifar.html, accessed on 5 September 2023), which is a collection of 60,000 32×32 color images (i.e., each image is a three-dimensional array of size $32 \times 32 \times 3$, where the third dimension represents the RGB color channels.) in 10 classes, with 6000 images per class. This dataset was collected by the well-known researcher Alex Krizhevsky [38]. There are 50,000 training images and 10,000 test images. The dataset is divided into five training batches and one test batch, each containing 10,000 images. The test batch contains exactly 1000 randomly selected images from each class. The 10 different classes represent airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. These classes are mutually exclusive, meaning an image can only belong to one class.

To evaluate the performance of our purifier, we first applied the aforementioned adversarial attacks to the all images in the test set. Then, we computed the classification accuracy as a metric, i.e., we fed the AEs generated by the adversarial attack into our purifier, and fed the purified image into a pre-trained classification model to predict the class of the image. We calculated the percentage of the predictions that matched the original image's predictions. Formally, let c_{target} be the number of images correctly predicted by the classification model, and T be the total number of (attacked) test set images. The accuracy is then calculated as follows:

$$Accuracy(\%) = \frac{c_{target}}{T} \times 100 \tag{7}$$

In our training, we used a batch size of 128, a learning rate of 0.01, and an Adaptive Moment Estimation (Adam) optimizer. The teacher model was trained for 100 epochs, while the student model was trained for 40 epochs. In the test scenarios, we used five different adversarial attacks: PGD, FGSM, BIM, CW, and AA. PGD, FGSM, and BIM generate noise using gradients which are then added to the input images. FGSM adds noise once, while PGD and BIM add noise iteratively to the images. Specifically, PGD generates noise based on the gradient from the adversarial sample produced in the previous iteration, while BIM consistently computes the gradient from the original input image. Consequently, when the iteration counts are equal, the magnitude of the noise produced by BIM is greater than that produced by PGD. The value of ϵ for each attack is set to $\frac{8}{255}$ by default. However, for PGD, since the value of $\frac{8}{255}$ was also used to train our purifier, we used an additional value of $\frac{16}{255}$, which was not used in the training. We name them PGD8 and PGD16, respectively. For PGD and BIM, we used α (step size) of $\frac{2}{255}$ and the iteration number of 20. For CW, we used L_2 as the distance function, 40 as the iteration number, and an Adam optimizer with a learning rate of 0.01.

4.2. Results and Analyses

First, to evaluate the superiority of our proposed purifier, we purified the AEs generated by the adversarial attacks described above, and then fed the purified images into a pre-trained classification model, ResNet56, to measure the classification accuracy. The accuracy of this classification model on CIFAR-10 is 89.46%. We employed NRP [29] and SOAP [30] as baseline purifiers for comparison.

Table 1 reports the experimental results. The proposed purifier generally performed satisfactorily against the gradient-based attacks PGD, FGSM, BIM, and AA. However, it performed slightly worse than SOAP against the PGD16 ($\epsilon = \frac{16}{255}$) and the BIM attacks, which add slightly stronger noise than PGD8 which was used for training. We also observed that our purifier did not perform well on samples subjected to CW attacks, which is likely due to the fact that CW is a different type of attack than the gradient-based attack used in the adversarial training of the AT teacher model.

Table 1. Comparison of purification performance against various adversarial attacks. Each bold number means the best performance against each adversarial attack.

	Ours	**NRP [29]**	**SOAP [30]**
Original	89.46	89.46	89.46
PGD8	**40.20**	35.07	39.14
PGD16	33.57	35.57	**35.65**
FGSM	**40.12**	39.79	37.43
BIM	38.37	31.96	**40.18**
CW	56.87	57.24	**79.22**
AA	**46.91**	12.34	41.03

Next, we performed an ablation study. Table 2 reports the results. The last row of the table is the proposed method (training two teacher models, AT and NT, and distilling their knowledge to our purifier model), and the two rows above it are versions of distilling the knowledge of only one teacher model, AT or NT, to the purifier, respectively. Finally, the first row is a purification method using only adversarial training without knowledge distillation. Through this experiment, we aim to gain insight into whether the performance of the purifier (student) model changes depending on the number of teacher models used and their learning strategies.

Table 2. Resultsof our ablation study. ✓ indicates that the corresponding component is adopted. Each bold number means the best performance against each adversarial attack.

AT	**NT**	**KD**	**PGD8**	**PGD16**	**FGSM**	**BIM**	**CW**	**AA**
✓	-	-	34.16	28.01	38.07	33.29	56.54	43.20
✓	-	✓	35.46	28.70	38.42	34.92	56.32	42.77
-	✓	✓	35.77	29.47	36.93	34.88	**57.90**	40.50
✓	✓	✓	**40.20**	**33.57**	**40.12**	**38.37**	56.87	**46.91**

The experimental results show that the proposed method generally performs best, and that using only one of the two teacher models or no knowledge distillation leads to lower performance. In particular, distilling the knowledge of the NT teacher model resulted in good performance, which suggests that the knowledge of image restoration is helpful in the purification task. However, the NT teacher model's knowledge alone was not sufficient to improve adversarial robustness of the student model, and we found that ensemble knowledge distillation from both AT and NT teachers was most effective.

5. Conclusions

In this paper, we proposed a novel adversarial purification framework for improving the robustness of deep neural networks against adversarial attacks. Our approach utilizes a convolutional autoencoder to capture image features and spatial structure, and a student model is trained on the purified images using knowledge distillation from two teacher models. Experimental results demonstrate that our proposed method can effectively remove adversarial noise from input images and improve model robustness against both white-box and black-box attacks. We also find that the number of teacher models used for knowledge distillation and the way they are trained affects the performance of the student model, and that using two different teacher models improves the performance of the purifier the most. Our approach also outperforms existing state-of-the-art methods in terms of accuracy and robustness. Future work will focus on exploring the effectiveness of our approach on other types of neural networks and datasets.

Author Contributions: Conceptualization, I.K. and D.-K.C.; methodology, I.K.; software, I.K.; validation, I.K. and D.-K.C.; formal analysis, I.K.; investigation, I.K.; resources, I.K.; data curation, I.K.; writing—original draft preparation, I.K. and D.-K.C.; writing—review and editing, D.-K.C. and

S.-C.L.; visualization, I.K.; supervision, D.-K.C. and S.-C.L.; project administration, D.-K.C. and S.-C.L.; funding acquisition, S.-C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the DGIST R&D program of the Ministry of Science and ICT of KOREA (23-IT-10-03 and 23-DPIC-08) and the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-01373, Artificial Intelligence Graduate School Program (Hanyang University)).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: CIFAR10 can be found here: https://www.cs.toronto.edu/~kriz/cifar.html, accessed on 5 September 2023.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Singh, A.; Chopra, A.; Garza, E.; Zhang, E.; Vepakomma, P.; Sharma, V.; Raskar, R. Disco: Dynamic and invariant sensitive channel obfuscation for deep neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12125–12135.
2. Feng, Y.; Feng, Y.; You, H.; Zhao, X.; Gao, Y. Meshnet: Mesh neural network for 3d shape representation. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8279–8286.
3. Clark, K.; Luong, M.; Le, Q.V.; Manning, C.D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In Proceedings of the 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, 26–30 April 2020. Available online: OpenReview.net (accessed on 5 September 2023).
4. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
5. Amodeo, I.; De Nunzio, G.; Raffaeli, G.; Borzani, I.; Griggio, A.; Conte, L.; Macchini, F.; Condò, V.; Persico, N.; Fabietti, I.; et al. A maChine and deep Learning Approach to predict pulmoNary hyperteNsIon in newbornS with congenital diaphragmatic Hernia (CLANNISH): Protocol for a retrospective study. *PLoS ONE* **2021**, *16*, e0259724. [CrossRef] [PubMed]
6. Taormina, V.; Raso, G.; Gentile, V.; Abbene, L.; Buttacavoli, A.; Bonsignore, G.; Valenti, C.; Messina, P.; Scardina, G.A.; Cascio, D. Automated Stabilization, Enhancement and Capillaries Segmentation in Videocapillaroscopy. *Sensors* **2023**, *23*, 7674. [CrossRef] [PubMed]
7. Dong, J.; Wang, Y.; Lai, J.H.; Xie, X. Improving adversarially robust few-shot image classification with generalizable representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 9025–9034.
8. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
9. Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; Vladu, A. Towards Deep Learning Models Resistant to Adversarial Attacks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
10. Kurakin, A.; Goodfellow, I.J.; Bengio, S. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*; Chapman and Hall/CRC: New York, NY, USA, 2018; pp. 99–112.
11. Carlini, N.; Wagner, D. Towards evaluating the robustness of neural networks. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (sp), San Jose, CA, USA, 22–26 May 2017; pp. 39–57.
12. Su, J.; Vargas, D.V.; Sakurai, K. One pixel attack for fooling deep neural networks. *IEEE Trans. Evol. Comput.* **2019**, *23*, 828–841. [CrossRef]
13. 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, PMLR, Virtual Event, 13–18 July 2020; pp. 2206–2216.
14. Silva, S.H.; Najafirad, P. Opportunities and challenges in deep learning adversarial robustness: A survey. *arXiv* **2020**, arXiv:2007.00753.
15. Liang, H.; He, E.; Zhao, Y.; Jia, Z.; Li, H. Adversarial attack and defense: A survey. *Electronics* **2022**, *11*, 1283. [CrossRef]
16. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In Proceedings of the Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, 14–17 June 2011; Springer: Berlin, Germany, 2011; pp. 52–59.
17. Hinton, G.; Vinyals, O.; Dean, J. Distilling the Knowledge in a Neural Network. In Proceedings of the NIPS Deep Learning and Representation Learning Workshop, Montreal, QC, Canada, 7–12 December 2015.
18. Kurakin, A.; Goodfellow, I.J.; Bengio, S. Adversarial Machine Learning at Scale. In Proceedings of the International Conference on Learning Representations, San Juan, PR, USA, 2–4 May 2016.

19. Xie, C.; Tan, M.; Gong, B.; Wang, J.; Yuille, A.L.; Le, Q.V. Adversarial examples improve image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 18–22 June 2020; pp. 819–828.
20. Kim, M.; Tack, J.; Hwang, S.J. Adversarial self-supervised contrastive learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 2983–2994.
21. Zhou, J.; Zhu, J.; Zhang, J.; Liu, T.; Niu, G.; Han, B.; Sugiyama, M. Adversarial Training with Complementary Labels: On the Benefit of Gradually Informative Attacks. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 23621–23633.
22. Song, Y.; Kim, T.; Nowozin, S.; Ermon, S.; Kushman, N. PixelDefend: Leveraging Generative Models to Understand and Defend against Adversarial Examples. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 May–3 April 2018.
23. Van Den Oord, A.; Kalchbrenner, N.; Kavukcuoglu, K. Pixel recurrent neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, New York City, NY, USA, 19–24 June 2016; pp. 1747–1756.
24. Yoon, J.; Hwang, S.J.; Lee, J. Adversarial purification with score-based generative models. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual Event, 18–24 July 2021; pp. 12062–12072.
25. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144. [CrossRef]
26. Samangouei, P.; Kabkab, M.; Chellappa, R. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
27. Jin, G.; Shen, S.; Zhang, D.; Dai, F.; Zhang, Y. Ape-gan: Adversarial perturbation elimination with gan. In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, Brighton, UK, 12–17 May 2019; pp. 3842–3846.
28. Kos, J.; Fischer, I.; Song, D. Adversarial examples for generative models. In Proceedings of the 2018 IEEE Security and Privacy Workshops (spw), IEEE, San Francisco, CA, USA, 24 May 2018; pp. 36–42.
29. Naseer, M.; Khan, S.; Hayat, M.; Khan, F.S.; Porikli, F. A self-supervised approach for adversarial robustness. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 262–271.
30. Shi, C.; Holtz, C.; Mishne, G. Online Adversarial Purification based on Self-supervised Learning. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 30 April 2020.
31. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. In Proceedings of the International Conference on Learning Representations, Banff, AB, Canada, 14–16 April 2014.
32. Vincent, P.; Larochelle, H.; Bengio, Y.; Manzagol, P.A. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; pp. 1096–1103.
33. Hwang, U.; Park, J.; Jang, H.; Yoon, S.; Cho, N.I. Puvae: A variational autoencoder to purify adversarial examples. *IEEE Access* **2019**, *7*, 126582–126593. [CrossRef]
34. Kalaria, D.R.; Hazra, A.; Chakrabarti, P.P. Towards Adversarial Purification using Denoising AutoEncoders. In Proceedings of the NeurIPS ML Safety Workshop, Virtual Event, 9 December 2022.
35. Nie, W.; Guo, B.; Huang, Y.; Xiao, C.; Vahdat, A.; Anandkumar, A. Diffusion Models for Adversarial Purification. In Proceedings of the International Conference on Machine Learning, PMLR, Baltimore, MD, USA, 17–23 June 2022; pp. 16805–16827.
36. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; Proceedings, Part III 18; Springer: Berlin, Germany, 2015; pp. 234–241.
37. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
38. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]