Open-set Adversarial Defense

Rui Shao[1[0000−0003−0090−9604]], Pramuditha Perera[2[0000−0003−2821−6367]⋆], Pong C. Yuen[1[0000−0002−9343−2202]], and Vishal M. Patel[3[0000−0002−5239−692X]]

[1]Department of Computer Science, Hong Kong Baptist University, Hong Kong
[2]AWS AI Labs, USA
[3]Department of Electrical and Computer Engineering, Johns Hopkins University, USA
{ruishao, pcyuen}@comp.hkbu.edu.hk, pramudi@amazon.com, vpatel36@jhu.edu

Abstract. Open-set recognition and adversarial defense study two key aspects of deep learning that are vital for real-world deployment. The objective of open-set recognition is to identify samples from open-set classes during testing, while adversarial defense aims to defend the network against images with imperceptible adversarial perturbations. In this paper, we show that open-set recognition systems are vulnerable to adversarial attacks. Furthermore, we show that adversarial defense mechanisms trained on known classes do not generalize well to open-set samples. Motivated by this observation, we emphasize the need of an Open-Set Adversarial Defense (OSAD) mechanism. This paper proposes an Open-Set Defense Network (OSDN) as a solution to the OSAD problem. The proposed network uses an encoder with feature-denoising layers coupled with a classifier to learn a noise-free latent feature representation. Two techniques are employed to obtain an informative latent feature space with the objective of improving open-set performance. First, a decoder is used to ensure that clean images can be reconstructed from the obtained latent features. Then, self-supervision is used to ensure that the latent features are informative enough to carry out an auxiliary task. We introduce a testing protocol to evaluate OSAD performance and show the effectiveness of the proposed method in multiple object classification datasets. The implementation code of the proposed method is available at: https://github.com/rshaojimmy/ECCV2020-OSAD.

Keywords: Adversarial Defense, Open-set Recognition

1 Introduction

A significant improvement has been achieved in the image classification task since the advent of deep convolutional neural networks (CNNs) [14]. The promising performance in classification has contributed to many real-world computer vision applications [41,37,40,38,39,36,42,45,47,20,2,46]. However,

⋆ This work was conducted prior to joining AWS AI Labs when the author was affiliated with Johns Hopkins University.

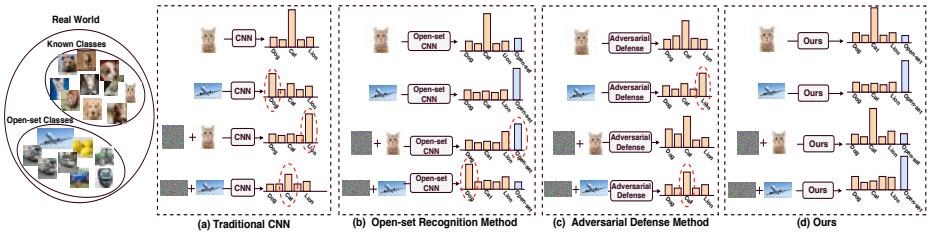

Fig. 1. Challenges in classification. (a) Conventional CNN classifiers fail in the presence of both open-set and adversarial images. (b) Open-set recognition methods can successfully identify open-set samples, but fail on adversarial samples. (c) Adversarial defense methods are unable to identify open-set samples. (d) Proposed method can identify open-set images and it is robust to adversarial images.

Table 1. Importance of an Open-set Adversarial Defense (OSAD) mechanism.

	Clean Images	Adversarial Images	
	Original Network	Original Network	Proposed Method
Closed Set Accuracy	96.0	31.8	88.2
Open-set Detection (AUC-ROC)	81.2	51.5	79.1

there exist several limitations of conventional CNNs that have an impact in real-world applications. In particular, open-set recognition [3,10,25,30,48,32,31,33,50] and adversarial attacks [12,24,5,19,44] have received a lot of interest in the computer vision community in the last few years.

A classifier is conventionally trained assuming that classes encountered during testing will be identical to classes observed during training. But in a real-world scenario, a trained classifier is likely to encounter open-set samples from classes unseen during training. When this is the case, the classifier will erroneously associate a known-set class identity to an open-set sample. Consider a CNN trained on animals classes. Given an input that is from an animal class (such as a cat), the network is able to produce the correct prediction as shown in Figure 1(a-First Row). However, when the network is presented with a non-animal image, such as an Airplane image, the classifier wrongly classifies it as one of the known classes as shown in Figure 1(a-Second Row). On the other hand, it is a well known fact that adding carefully designed imperceptible perturbations to clean images can alter model prediction in a classifier [12]. These types of *adversarial attacks* are easy to deploy and may be encountered in real-world applications [9]. In Figure 1(a-Third Row) and Figure 1(a-Fourth Row), we show how such adversarial attacks can affect model prediction for known and open-set images, respectively.

Computer vision community has developed several open-set recognition algorithms [3,10,25,30,48] to combat against the former challenge. These algorithms convert the c-class classification problem into a $c+1$ class problem

by considering open-set classes as an additional class. These algorithms provide correct classification decisions for both known and open-set classes as shown in Figure 1(b-First and Second rows). However, in the presence of adversarial attacks, these models fail to produce correct predictions as illustrated in Figure 1(b-Third and Fourth rows). On the other hand, there exist several defense strategies [19,44,22,17] that are proposed to counter the latter challenge. These defense mechanisms are designed with the assumption of closed-set testing. Therefore, although they work well when this assumption holds (Figure 1(c-First and third rows)), they fail to generalize well in the presence of open-set samples as shown in Figure 1(c-Second and Fourth rows).

Based on this discussion, it is evident that existing solutions in the open-set recognition paradigm does not necessarily complement well with adversarial defense and vise versa. This observation motivates us to introduce a new research problem – Open-Set Adversarial Defense (OSAD), where the objective is to simultaneously detect open-set samples and classify known classes in the presence of adversarial noise. In order to demonstrate the significance of the proposed problem, we conducted an experiment on the CIFAR10 dataset by considering only 6 classes to be known to the classifier. In Table 1 we tabulate both open-set detection performance (expressed in terms of area under the curve of the ROC curve) and closed-set classification accuracy for this experiment. When the network is presented with clean images, it produces a performance better than 0.8 in both open-set detection and closed set classification. However, as evident from Table 1, when images are attacked, open-set detection performance drops along with the closed set accuracy by a significant margin. It should be noted that, open-set detection performance in this case is close to random guessing (0.5).

This paper proposes an Open-Set Defense Network (OSDN) that learns a noise-free, informative latent feature space with the aim of detecting open-set samples while being robust to adversarial images. We use an autoencoder network with a classifier branch attached to its latent space as the backbone of our solution. The encoder network is equipped with feature-denoising layers [44] with the aim of removing adversarial noise. We use self-supervision and decoder reconstruction processes to make sure that the learned feature space is informative enough to detect open-set samples. The reconstruction process uses the decoder to generate noise-free images based on the obtained latent features. Self-supervision is carried out by forcing the network to perform an auxiliary classification task based on the obtained features. The proposed method is able to provide robustness against adversarial attacks in terms of classification as well as open-set detection as shown in Table 1. Main contributions of our paper can be summarized as follows:

1. This paper proposes a new research problem named Open-Set Adversarial Defense (OSAD) where adversarial attacks are studied under an open-set setting.

2. We propose an Open-Set Defense Network (OSDN) that learns a latent feature space that is robust to adversarial attacks and informative to identify open-set samples.

3. A test protocol is defined to the OSAD problem. Extensive experiments are conducted on three publicly available image classification datasets to demonstrate the effectiveness of the proposed method.

2 Related Work

Adversarial Attack and Defense Methods. Szegedy *et al.* [43] reported that carefully crafted imperceptible perturbations can be used to fool a CNN to make incorrect predictions. Since then, various adversarial attacks have been proposed in the literature. Fast Gradient Sign Method (FGSM) [12] was proposed to consider the sign of a gradient update from the classifier to generate adversarial images. Basic Iteration Method (BIM) [19] and Projected Gradient Descent (PGD) [24] extended FGSM to stronger attacks using iterative gradient descent. Different from the above gradient-based adversarial attacks, Carlini and Wagner [5] proposed the C&W attack to generate adversarial samples by taking a direct optimization approach. Adversarial training [24] is one of the most widely-used adversarial defense mechanisms. It provides defense against adversarial attacks by training the network on adversarially perturbed images generated on-the-fly based on model's current parameters. Several recent works have proposed denoising-based operations to further improve adversarial training. Pixel denoising [22] was proposed to exploit the high-level features to guide the denoising process. The most influential local parts to conduct the pixel-level denoising is found in [13] based on class activation map responses. Adversarial noise removal is carried out in the feature-level using denoising filters in [44]. Effectiveness of this process is demonstrated using a selection of different filters.

Open-set Recognition. Possibility for open-set samples to generate very high probability scores in a closed-set classifier was first brought to attention in [35]. It was later shown that deep learning models are also affected by the same phenomena [3]. Authors in [3] proposed a statistical solution, called OpenMax, for this problem. They converted the c-class classification problem into a $c + 1$ problem by considering the extra class to be the open-set class. They apportioned logits of known classes to the open-set class considering spatial positioning of a query sample in an intermediate feature space. This formulation was later adopted by [10] and [25] by using a generative model to produce logits of the open-set class. Authors in [48] argued that a generative feature contain information that can benefit open-set recognition. On these grounds they considered a concatenation of a generative feature and a classifier feature when building the OpenMax layer. A generative approach was used in [30] where a class conditioned decoder was used to detect open-set samples. Works of both [30] and [48] show that incorporating generative features can benefit open-set recognition. Note that open-set recognition is more challenging than the novelty detection [27,28,31,34,29] which only aims to determine whether an observed image during inference belongs to one of the known classes.

Self-Supervision. Self-supervision is an unsupervised machine learning technique where the data itself is used to provide supervision. Recent works in self-supervision introduced several techniques to improve the performance in classification and detection tasks. For example, in [7], given an anchor image patch, self-supervision was carried out by asking the network to predict the relative position of a second image patch. In [8], a multi-task prediction framework extended this formulation, forcing the network to predict a combination of relative order and pixel color. In [11], the image was randomly rotated by a factor of 90 degrees and the network was forced to predict the angle of the transformed image.

3 Background

Adversarial Attacks. Consider a trained network parameterized by parameters θ. Given a data and label pair $(\mathbf{x}, \mathbf{y})$, an adversarial image $\mathbf{x}_{adv}$, can be produced using $\mathbf{x}_{adv} = \mathbf{x} + \delta$, where δ can be determined by a given white-box attack based on the models parameters. In this paper, we consider two types of adversarial attacks.

The first attack considered is the Fast Gradient Signed Method (FGSM) [12] where the adversarial images are formed as follows,

$$\mathbf{x}_{adv} = \text{Proj}_{\chi}(\mathbf{x} + \epsilon sign(\bigtriangledown_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{y}; \theta))), \tag{1}$$

where $\mathcal{L}(\cdot)$ is a classification loss. Proj_{χ} denotes the projection of its element to a valid pixel value range, and ϵ denotes the size of l_∞-ball. The second attack considered is Projective Gradient Descent (PGD) attacks [24]. Adversarial images are generated in this method as follows,

$$\mathbf{x}_{adv}^{(t+1)} = \text{Proj}_{\zeta \cap \chi}(\mathbf{x}_{adv}^{(t)} + \epsilon_{step} sign(\bigtriangledown_{\mathbf{x}} \mathcal{L}(\mathbf{x}_{adv}^{(t)}, \mathbf{y}; \theta))), \tag{2}$$

where $\text{Proj}_{\zeta \cap \chi}(\cdot)$ denotes the projection of its element to l_∞-ball ζ and a valid pixel value range, and ϵ_{step} denotes a step size smaller than ϵ. We use the adversarial samples of the final step T: $\mathbf{x}_{adv} = \mathbf{x}_{adv}^{(T)}$.

OpenMax Classifier. A SoftMax classifier trained for a c-class problem typically has c probability predictions. OpenMax is an extension where the probability scores of $c + 1$ classes are produced. The probability of the final class corresponds to the open-set class. Given c known classes $\mathcal{K} = \{C_1, C_2, ..., C_c\}$, OpenMax is designed to identify open-set samples by calibrating the final hidden layer of a classifier as follows:

$$\hat{\boldsymbol{l}}_i = \begin{cases} \boldsymbol{l}_i \boldsymbol{w}_i & (i \leq c) \\ \sum_{i=1}^{c} \boldsymbol{l}_i (1 - \boldsymbol{w}_i) & (i = c + 1) \end{cases}, \text{OpenMax}_i(\mathbf{x}) = \text{SoftMax}_i(\hat{\boldsymbol{l}}) \tag{3}$$

where $\boldsymbol{l}$ denotes the logit vector obtained prior to the SoftMax operation in a classifier, $\boldsymbol{w}_i$ represents the belief that $\mathbf{x}_{adv}$ belongs to the known class C_i.

Here, the class C_{N+1} corresponds to the open-set class. Belief $\boldsymbol{w}_i$ is calculated considering the distance of a given sample to it's class mean μ in an intermediate feature space. During training, distance of all training image samples from a given class to its corresponding class mean μ is evaluated to form a matched score distribution. Then, a Weibull distribution is fitted to the tail of the matched distribution. If the feature representation of the input in the same feature space is $\boldsymbol{v(x)}$, $\boldsymbol{w}_i$ is calculated as

$$\boldsymbol{w}_i = 1 - \max\Big(0, \frac{\sigma - \text{rank}(i)}{\sigma}\Big) e^{\Big(-\Big(\frac{|\boldsymbol{v(x)} - \mu_i|_2}{\eta_i}\Big)^{m_i}\Big)}, \tag{4}$$

where m_i, η_i are parameters of the Weibull distribution that corresponding to class C_i. σ is hyperparameter and rank(i) is the index in the logits sorted in the descending order.

4 Proposed Method

The proposed network consists of four CNNs: encoder, decoder, open-set classifier and transformation classifier. In Figure 2, we illustrate the network structure of the proposed method and denote computation flow. The encoder network consists of several feature-denoising layers [44] between the convolutional layers. Open-set classifier has no structural difference from a regular classifier. However, an OpenMax layer is added to the end of the classifier during inference. We denote this by indicating an OpenMax layer in Figure 2.

Given an input image, first the network generates an adversarial image based on the current network parameters. This image is passed through the encoder network to obtain the latent feature. This feature is passed through the open-set classifier via path (1) to evaluate the cross entropy loss $\mathcal{L}_{cls}$. Then, the image corresponding to the obtained latent feature is generated by passing the feature through the decoder following path (2). The decoded image is used to calculate its difference to the corresponding clean image based on the reconstruction loss $\mathcal{L}_{rec}$. Finally, the input image is subjected to a geometric transform. An adversarial image corresponding to the transformed image is obtained. This image is passed through path (3) to arrive at the transformation classifier. Output of the classifier is used to calculate the cross entropy loss $\mathcal{L}_{ssd}$ considering the transform applied to the image. The network is trained end-to-end using the following loss function

$$\mathcal{L}_{OSDN} = \mathcal{L}_{cls} + \mathcal{L}_{rec} + \mathcal{L}_{ssd}. \tag{5}$$

In the following subsections, we describe various components and computation involved in all three paths in detail.

Noise-free Feature Encoding. The proposed network uses an encoder network to produce noise-free features. Then, the open-set classifier operating

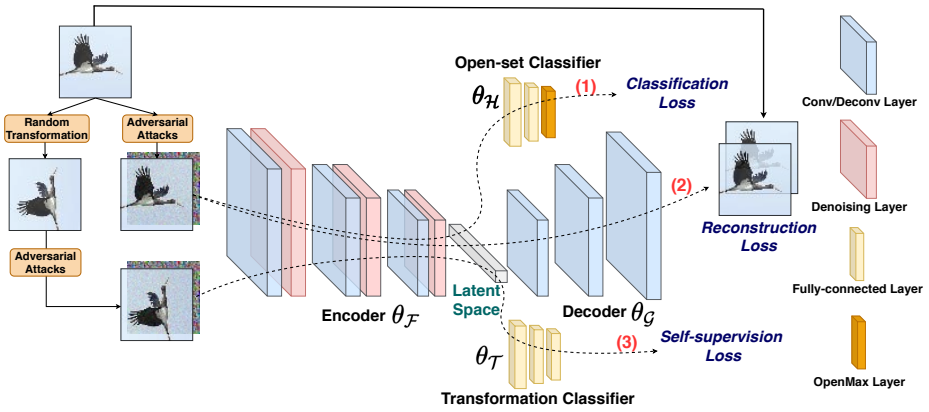

Fig. 2. Network structure of the proposed Open-Set Defense Network (OSDN). It consists of four components: encoder, decoder, open-set classifier and transformation classifier.

on the learned feature is used to perform classification. During training, there is no structural difference in the open-set classifier from a standard classifier. Inspired by [44], we embed denoising layers after the convolutional layer blocks in the encoder so that feature denoising can be explicitly carried out on adversarial samples. We adopt the Gaussian (softmax) based non-local means filter [4] as the denoising layer. Given an input feature map m, non-local means [4] takes a weighted mean of features in the spatial region $\mathcal{R}$ to compute a denoised feature map g as follows

$$g_i = \frac{1}{\mathcal{N}(m)} \sum_{\forall j \sim \mathcal{R}} f(m_i, m_j) \cdot m_j, \tag{6}$$

where $f(m_i, m_j)$ is a feature-dependent weighting function. For the Gaussian (softmax) based version, $f(m_i, m_j) = e^{\alpha(m_i)^T \beta(m_j)/\sqrt{d}}$. α and β are two 1×1 convolutional layers as embedding functions and d corresponds to the number of channels. $\mathcal{N}(m)$ is a normalization function and $\mathcal{N}(m) = \sum_{\forall j \sim \mathcal{R}} f(m_i, m_j)$.

Formally, we denote the encoder embedded with denoising layers as $\mathcal{F}$ parameterized by $\theta_{\mathcal{F}}$, and the classifier as $\mathcal{H}$ parameterized by $\theta_{\mathcal{H}}$. Given the labeled clean data $(\mathbf{x}, \mathbf{y}) \sim \mathcal{I}$ from the data distribution of known classes, we can generate the corresponding adversarial images $\mathbf{x}_{adv}$ on-the-fly using either FGSM or PGD attacks based on the current parameters $\theta_{\mathcal{F}}$, $\theta_{\mathcal{H}}$ using the true label $\mathbf{y}$. Obtained adversarial image $\mathbf{x}_{adv}$ is passed through encoder and classifier (via path (1)) to arrive at the cross-entropy loss defined as

$$\mathcal{L}_{cls} = \min_{\theta_{\mathcal{F}}, \theta_{\mathcal{H}}} \mathbb{E}_{(\mathbf{x}_{adv}, \mathbf{y}) \sim \mathcal{I}} \mathcal{L}_{CE}(\mathbf{x}_{adv}, \mathbf{y}; \theta_{\mathcal{F}}, \theta_{\mathcal{H}}). \tag{7}$$

By minimizing the above adversarial training loss, the trained encoder embedded with the denoising layers is able to learn a noise-free latent feature

space. During inference, an OpenMax layer is added on top of the classifier. With this formulation, open-set classifier operating on the noise-free latent feature learns to predict the correct class, for both known and open-set samples, even when the input is contaminated with adversarial noise.

Clean Image Generation. In this section, we introduce the image generation branch proposed in our method. The objective of the image generation branch is to generate noise-free images from adversarial images by taking advantage of the decoder network. This is motivated by two factors.

First, autoencoders are widely used in the literature for image denoising applications. By forcing the autoencoder network to produce noise-free images, we are providing additional supervision to remove noise in the latent feature space. Secondly, it is a well known fact that open-set recognition becomes more effective in the presence of more descriptive features [48]. When a classifier is trained, it models the boundary of each class. Therefore, a feature produced by a classification network only contains information that is necessary to model class boundaries. However, when the network is asked to generate noise-free images based on the latent representations, it ends up with learning generative features. As a result, features become more descriptive than in the case of a pure classifier. In fact, such generative features are used in [48] and [30] to boost the open-set recognition performance. Therefore, we argue that adding the decoder as an image generation branch can mutually benefit both open-set recognition and adversarial defense.

We pass adversarial images through path (2) as illustrated in Figure 2 to generate the decoded images. The decoder network denoted as $\mathcal{G}$ parameterized by $\theta_{\mathcal{G}}$ and the encoder network $\mathcal{F}$ are jointly optimized to minimize the distance between the decoded images and the corresponding clean images using the following loss

$$\mathcal{L}_{rec} = \min_{\theta_{\mathcal{F}}, \theta_{\mathcal{G}}} \mathbb{E}_{(\mathbf{x}, \mathbf{x}_{adv}) \sim \mathcal{I}} \|\mathbf{x} - \mathcal{G}(\mathcal{F}(\mathbf{x}_{adv}))\|_2^2. \tag{8}$$

Self-supervised Denoising. Finally, we propose to use self-supervision as a means to further increase the informativeness and robustness of the latent feature space. Self-supervision is a machine learning technique that is used to learn representations in the absence of labeled data. In our work we adopt rotation-based self-supervision proposed in [11]. In [11], first, a random rotation from a finite set of possible rotations is applied to an image. Then, a classifier is trained on top of a latent feature vector to automatically recognize the applied rotation.

In our approach, similar to [11], we first generate a random number $r \in \{0, 1, 2, 3\}$ as the rotation ground-truth and transform the input clean image $\mathbf{x}$ by rotating it with $90°r$ degrees. Then, based on the rotated clean image, we generate a rotated adversarial image $\mathbf{x}_{adv}^{\mathcal{T}}$ on-the-fly using either FGSM or PGD attack based on the current network parameters and rotation ground-truth r, which is passed through the transformation classifier to generate the cross-entropy loss with respect to the ground-truth r. We denote the transformation classifier as $\mathcal{T}$ parameterized by $\theta_{\mathcal{T}}$ and formulate the adversarial training loss

function for self-supervised denoising as follows

$$\mathcal{L}_{ssd} = \min_{\theta_{\mathcal{F}},\theta_{\mathcal{T}}} \mathbb{E}_{\mathbf{x}^{\mathcal{T}}_{adv}\sim\mathcal{I}} \mathcal{L}_{CE}(\mathbf{x}^{\mathcal{T}}_{adv}, r; \theta_{\mathcal{F}}, \theta_{\mathcal{T}}). \tag{9}$$

There are multiple reasons why we use self-supervision in our method. When a classifier learns to differentiate between different rotations, it learns to pay attention to object structures and orientations of known classes. As a result, when self-supervision is carried out in addition to classification, the underlying feature space learns to represent additional information that was not considered in the case of a pure classifier. Therefore, self-supervision enhances the informativeness of the latent feature space which would directly benefit the open-set recognition process. On the other hand, since we use adversarial images for self-supervision, this operation directly contributes towards learning the denoising operator in the encoder. It should also be noted that recent work [16] has found that self-supervised learning contributes towards robustness against adversarial samples. Therefore, we believe that addition of self-supervision benefits both open-set detection and adversarial defense processes.

Implementation Details. We adopt the structure of Resnet-18 [14], which has four main blocks, for the encoder network. Denoising layers are embedded after each main blocks in the encoder. For the decoder, we use the decoder network proposed in [25] with three transpose-convolution layers for conducting experiments with the SVHN and CIFAR10 dataset. Four transpose-convolution layers are used for conducting experiments with the TinyImageNet dataset. For both open-set classifier and transformation classifier, we use a single fully connected layers. We use the Adam optimizer [18] for the optimization with a learning rate of 1e-3. We carried out model selection considering the trained model that has produced the best closed-set accuracy on the validation set. We use the iteration = 5 for the PGD attacks and $\epsilon = 0.3$ for the FGSM attacks in both adversarial training and testing.

5 Experimental Results

In order to assess the effectiveness of the proposed method, we carry out experiments on three multiple-class classification datasets. In this section, we first describe datasets, baseline methods and the protocol used in our experiments. We evaluate our method and baselines in the task of open-set recognition under adversarial attacks. To further validate the effectiveness of our method, additional experiments in the task of out-of-distribution detection under adversarial attacks are conducted. We conclude the section by presenting an ablation study and various visualizations with a brief analysis of the results.

5.1 Datasets

The evaluation of our method and other state-of-the-arts are conducted on three standard images classification datasets for open-set recognition:

SVHN and CIFAR10. Both CIFAR10 [1] and SVHN [26] are classification datasets with 10 classes with images of size 32×32. Street-View House Number dataset (SVHN) contains house number signs extracted from Google Street View. CIFAR10 contains images from four vehicle classes and six animal classes. We randomly split 10 classes into 6 known classes and 4 open-set classes to simulate open-set recognition scenario. We consider three randomly selected splits for testing[1].

TinyImageNet. TinyImageNet contains a sub-set of 200 classes selected from the ImageNet dataset [6] with image size of 64×64. 20 classes are randomly selected to be known and the remaining 180 classes are chosen to be open-set classes. We consider three randomly chosen splits for evaluation.

5.2 Baseline Methods.

We consider the following two recently proposed adversarial defense methods as baselines: **Adversarial Training** [24] and **Feature Denoising** [44]. We add an OpenMax layer in the last hidden layer during testing for both baselines to facilitate a fair comparison in open-set recognition. Moreover, to evaluate the performance of a classifier without a defense mechanism, we train a Resnet-18 network on clean images obtained from known classes and add an OpenMax layer during testing. We test this network using clean images for inference and we denote this test case by **clean**. Furthermore, we test this model with adversarial images, which is denoted as **adv on clean**.

5.3 Quantitative Results

Table 2. Adversarial Defense: Closed-set accuracy.

Method	**SVHN**		**CIFAR-10**		**TinyImageNet**	
	FGSM	PGD	FGSM	PGD	FGSM	PGD
clean	96.0±0.6	96.0±0.6	93.1±1.8	93.1±1.8	56.8±3.6	56.8±3.6
adv on clean	41.6±3.2	39.3±1.8	31.8±4.5	13.0±4.0	11.2±2.6	4.4±0.8
adversarial training	88.5±2.7	75.8±2.5	87.3±1.1	72.4±4.6	66.6±1.2	40.3±2.3
feature denoising	86.9±3.7	75.5±2.6	87.4±2.3	72.5±4.5	64.5±1.3	39.3±3.0
ours	**89.3**±0.7	**77.9**±1.6	**88.2**±2.9	**74.2**±4.3	**75.1**±7.9	**41.6**±2.2

Open-set Recognition. In conventional open-set recognition, the model is required to perform two tasks. First, it should be able to detect open-set samples effectively. Secondly, it should be able to perform correct classification on closed set samples. In order to evaluate the open-set defense performance, we take these two factors into account. In particular, following previous open-set works [25], we use area under the ROC curve (AUC-ROC) to evaluate the performance on identifying open-set samples under adversarial attacks. In order to evaluate the

[1] Details about known classes present in each split can be found in supplementary materials.

closed-set accuracy, we calculate prediction accuracy by only considering known-set samples in the test set. In our experiments, both known and open-set samples were subjected to adversarial attacks prior to testing. During our experiments we consider FGSM and PGD attacks to attack the model. We generated adversarial samples from known classes using the ground-truth labels, while we generated the adversarial samples from open-set classes based on model's prediction.

We tabulate the obtained performance for closed-set accuracy and open-set detection in Tables 2 and 3, respectively. We note that, networks trained on clean images produce very high recognition performance for clean images under both scenarios. However, when the adversarial noise is present, both open-set detection and closed-set classification performance drops significantly. This validates that current adversarial attacks can easily fool an open-set recognition method such as OpenMax, and thus OSAD is a critical research problem. Both baseline defense mechanisms considered are able to improve the recognition on both known and open-set samples. It can be observed from Tables 2 and 3, that the proposed method obtains the best open-set detection performance and closed-set accuracy compared to all considered baselines across all three datasets. In particular, the proposed method has achieved about 7% improvement in open-set detection on the SVHN dataset compared to the other baselines. On other datasets, this improvement varies between $1-5\%$. The proposed method is also able to perform better in terms of closed-set accuracy compared to the baselines consistently across datasets.

It is interesting to note that methods involving adversarial training perform better than the baseline of clean image classification under FGSM attacks on the TinyImageNet dataset. This is because only 20 classes from the TinyImageNet dataset are selected for training and each class has only 500 images. When a small dataset is used to train a model with large number of parameters, it is easier for the network to overfit to the training set. Such network observes variety of data in the presence of adversarial training. Therefore model reaches a more generalizable optimization solution during training.

Table 3. Open Set Classification: Area under the ROC curve.

Method	**SVHN**		**CIFAR-10**		**TinyImageNet**	
	FGSM	PGD	FGSM	PGD	FGSM	PGD
clean	91.3±2.4	91.3±2.4	81.2±2.9	81.2±2.9	59.5±0.8	59.5±0.8
adv on clean	56.4±1.2	54.1±2.9	51.5±2.8	45.5±0.5	47.9±2.7	48.6±1.3
adversarial training	61.4±8.0	65.2±4.0	75.2±1.2	68.7±3.2	65.1±8.1	56.5±0.9
feature denoising	64.5±14.7	64.9±4.2	76.9±3.7	69.8±2.4	65.3±5.1	56.1±1.6
ours	**71.4**±4.2	**71.6**±2.6	**79.1**±1.0	**70.6**±1.7	**70.8**±5.1	**58.2**±1.9

Out-of-distribution detection. In this sub-section, we evaluate the performance of the proposed method on the out-of-distribution detection (OOD) [15] problem on CIFAR10 using the protocol described in [48]. We considered all classes in CIFAR10 as known-classes and consider test images from ImageNet and LSUN dataset [49] (both cropped and resized) as out-of-distribution images [21]. We tested the performance of adversarial

images by creating adversarial images using the PGD attacks for both known and OOD data. We generated adversarial samples from the known classes using the ground-truth labels, while we generated adversarial samples from the OOD class based on model's prediction. We evaluated the performance of the model on adversarial samples based on macro-averaged F1 score. We used OpenMax layer with threshold 0.95 when assigning open-set labels to the query images, In Table 4, we tabulate the OOD detection performance across all four cases considered for both baselines as well as the proposed method. As evident from Table 4, the proposed method outperforms baseline methods in all test cases in the ODD task. This experiment further verifies the effectiveness of our method to identify samples from open-set classes even under the adversarial attacks.

Table 4. Performance of out-of-distribution object detection on the CIFAR10 dataset.

Detector	ImageNet-Crop	ImageNet-Resize	LSUN-Crop	LSUN-Resize
clean	78.9	76.2	82.1	78.7
adv on clean	4.7	4.4	7.3	3.8
adversarial training	35.2	34.5	35.0	34.7
feature denoising	43.2	41.0	43.5	41.2
ours	**46.5**	**44.8**	**47.1**	**44.2**

5.4 Ablation Study

The proposed network consists of four CNN components. In this sub-section we investigate the impact of each network component to the overall performance of the system. To validate the effectiveness of various parts integrated in our proposed network, this section conducts the ablation study in our network using the CIFAR10 dataset for the task of open-set recognition. Considered cases and the corresponding results obtained for each case are tabulated Table 5 (C-accuracy means closed-set accuracy). From Table 5, it can be seen that compared to normal adversarial training with an encoder, embedding the denoising layers helps to improve the open-set classification performance. Moreover, as evident from Table 5, adding a denoising layer to perform feature denoising and adding self-supervision both have resulted in improved performance. The proposed method that integrates all these components has the best performance, which shows that added components complement each other to produce better performance for both adversarial defense and open-set recognition.

5.5 Qualitative Results

In this section, we visualize the results of the denoising operation and obtained features in a 2D plane to qualitatively analyze the performance of the proposed method. For this purpose, we first consider the SVHN dataset. Figure 3 shows a set of clean images, corresponding PGD attacks adversarial images and images

Table 5. Results corresponding to the ablation study.

Method	CIFAR-10	
	AUC-ROC	C-accuracy
clean	83.7	92.7
adv on clean	45.9	8.6
Encoder	66.1	69.9
Encoder+Denoising Layer	68.5	70.4
Encoder+Decoder	67.3	68.8
Encoder+Decoder+Denoising Layer	68.2	70.6
Encoder+Decoder+self-supervised Denoising	68.5	71.9
ours	**69.6**	**72.8**

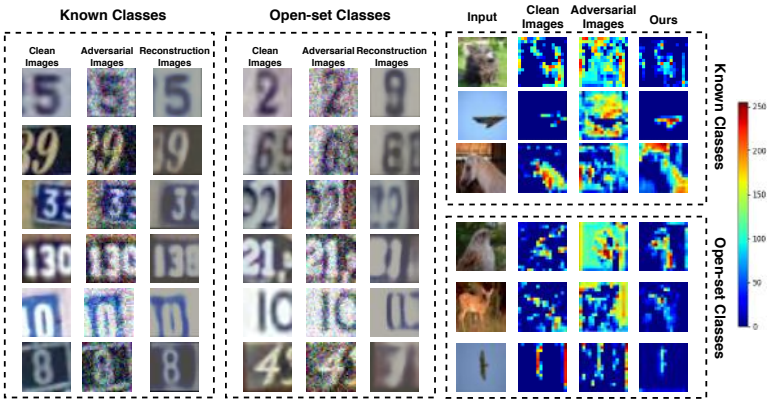

Fig. 3. Visualization of input clean images, corresponding adversarial images, and the reconstructed images from the decoder.

Fig. 4. Feature map visualization in the res_2 block of Resnet-18 and the encoder of proposed network.

obtained when the latent feature is decoded under the proposed method. We have indicated known and open-set sample visualizations in two columns. From the image samples shown in Figure 3, it can be observed that image noise has been removed in both open-set and known-class images. However, the reconstruction quality is superior for the known-class samples compared to the open-set images. Reconstructions of open-set samples look blurry and structurally different. For example, the image of digit 2 shown in the first row, looks similar to the digit 9 once reconstructed.

In Figure 5 we visualize latent features obtained in the proposed method along with two other baselines using tSNE visualization [23] . As shown in Figure 5, most of open-set features lie away from the known-set feature distribution based on our method. This is why the proposed method is able to obtain good open-set detection performance. On the other hand, it can be observed from Figure 5 that there is more overlap between the two types of features in all baseline methods. When open-set features lie away from the data manifold of known set classes, the reconstruction obtained through the decoder network tends to be poor. Therefore, the tSNE plot justifies why the

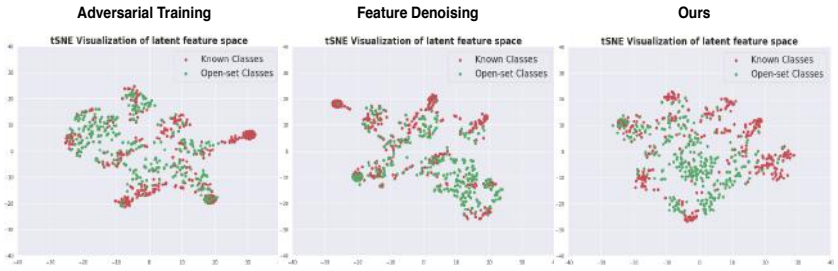

Fig. 5. tSNE visualization of the latent feature space corresponding to our method and two baselines.

reconstruction of our method was poor for open-set samples in Figure 3. As such, Figure 3 and Figure 5 mutually verify the effectiveness of our method for defending adversarial samples and identifying open-set samples simultaneously.

Moreover, we visualize randomly selected feature maps of the second residual block from the trained Resnet-18 [14] and the encoder of the proposed OSDN network applied on clean images and on its adversarially perturbed counterpart in the CIFAR10 dataset. From Figure 4, it can be observed that compared to Resnet-18, the proposed network is able to reduce adversarial noise significantly in feature maps of adversarial images in both known and open-set classes. This further demonstrates that the proposed network indeed carries out the feature denoising through the embedded feature denoising layers.

6 Conclusion

In this paper, we studied a novel research problem – Open-set Adversarial Defense (OSAD). We first showed that existing adversarial defense mechanisms do not generalize well to open-set samples. Furthermore, we showed that even open-set classifiers can be easily attacked using the existing attack mechanisms. We proposed an Open-Set Defense Network (OSDN) with the objective of producing a model that can detect open-set samples while being robust against adversarial attacks. The proposed network consisted of feature denoising operation, self-supervision operation and a denoised image generation function. We demonstrated the effectiveness of the proposed method under both open-set and adversarial attack settings on three publicly available classification datasets. Finally, we showed that proposed method can be deployed for out-of-distribution detection task as well.

Acknowledgments

This work is partially supported by Research Grants Council (RGC/HKBU12200518), Hong Kong. Vishal M. Patel was supported by the DARPA GARD Program HR001119S0026-GARD-FP-052.

References

1. Alex Krizhevsky, V.N., Hinton, G.: Cifar-10(canadian institute for advanced research)
2. Baweja, Y., Oza, P., Perera, P., Patel, V.M.: Anomaly detection-based unknown face pre- sentation attack detection. In: IJCB (2020)
3. Bendale, A., Boult, T.E.: Towards open set deep networks. In: CVPR (2016)
4. Buades, A., Coll, B., Morel, J.M.: A non-local algorithm for image denoising. In: CVPR (2005)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: SP (2017)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
7. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: ICCV (2015)
8. Doersch, C., Zisserman, A.: Multi-task self-supervised visual learning. In: ICCV (2017)
9. Evtimov, I., Eykholt, K., Fernandes, E., Kohno, T., Li, B., Prakash, A., Rahmati, A., Song, D.: Robust physical-world attacks on deep learning models. In: CVPR (2018)
10. Ge, Z., Demyanov, S., Chen, Z., Garnavi, R.: Generative openmax for multi-class open set classification. BMVC (2017)
11. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. ICLR (2018)
12. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. ICLR (2014)
13. Gupta, P., Rahtu, E.: Ciidefence: Defeating adversarial attacks by fusing class-specific image inpainting and image denoising. In: CVPR (2019)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
15. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. ICLR (2017)
16. Hendrycks, D., Mazeika, M., Kadavath, S., Song, D.: Using self-supervised learning can improve model robustness and uncertainty. In: NIPS (2019)
17. Jang, Y., Zhao, T., Hong, S., Lee, H.: Adversarial defense via learning to generate diverse attacks. In: ICCV (2019)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. ICLR (2017)
20. Lan, X., Ye, M., Shao, R., Zhong, B., Yuen, P.C., Zhou, H.: Learning modality-consistency feature templates: A robust rgb-infrared tracking system. In: IEEE Trans. Industrial Electronics, 66(12), 9887–9897 (2019)
21. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. ICLR (2018)
22. Liao, F., Liang, M., Dong, Y., Pang, T., Hu, X., Zhu, J.: Defense against adversarial attacks using high-level representation guided denoiser. In: CVPR (2018)
23. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)

24. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. ICLR (2018)
25. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: ECCV (2018)
26. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011)
27. Oza, P., Nguyen, H.V., Patel, V.M.: Multiple class novelty detection under data distribution shift. In: ECCV (2020)
28. Oza, P., Patel, V.M.: Utilizing patch-level activity patterns for multiple class novelty detection. In: ECCV (2020)
29. Oza, P., Patel, V.M.: One-class convolutional neural network. IEEE Signal Processing Letters **26**(2), 277–281 (2018)
30. Oza, P., Patel, V.M.: C2ae: Class conditioned auto-encoder for open-set recognition. In: CVPR (2019)
31. Perera, P., Patel, V.M.: Deep transfer learning for multiple class novelty detection. In: CVPR (2019)
32. Perera, P., Morariu, V.I., Jain, R., Manjunatha, V., Wigington, C., Ordonez, V., Patel, V.M.: Generative-discriminative feature representations for open-set recognition. In: CVPR (2020)
33. Perera, P., Nallapati, R., Xiang, B.: OCGAN: One-class novelty detection using gans with constrained latent representations. In: CVPR (2019)
34. Perera, P., Patel, V.M.: Learning deep features for one-class classification. IEEE Transactions on Image Processing **28**(11), 5450–5463 (2019)
35. Scheirer, W.J., Rocha, A., Sapkota, A., Boult, T.E.: Towards open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI) **35** (July 2013)
36. Shao, R., Lan, X.: Adversarial auto-encoder for unsupervised deep domain adaptation. In: IET Image Processing (2019)
37. Shao, R., Lan, X., Li, J., Yuen, P.C.: Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In: CVPR (2019)
38. Shao, R., Lan, X., Yuen, P.C.: Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3D mask face anti-spoofing. In: IJCB (2017)
39. Shao, R., Lan, X., Yuen, P.C.: Feature constrained by pixel: Hierarchical adversarial deep domain adaptation. In: ACM MM (2018)
40. Shao, R., Lan, X., Yuen, P.C.: Joint discriminative learning of deep dynamic textures for 3D mask face anti-spoofing. In: IEEE Trans. Inf. Forens. Security, 14(4): 923-938 (2019)
41. Shao, R., Lan, X., Yuen, P.C.: Regularized fine-grained meta face anti-spoofing. In: AAAI (2020)
42. Shao, R., Perera, P., Yuen, P.C., Patel, V.M.: Federated face anti-spoofing. arXiv preprint arXiv:2005.14638 (2020)
43. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. ICLR (2014)
44. Xie, C., Wu, Y., Maaten, L.v.d., Yuille, A.L., He, K.: Feature denoising for improving adversarial robustness. In: CVPR (2019)
45. Ye, M., Shen, J., Lin, G., Xiang, T., Shao, L., Hoi, S.C.H.: Deep learning for person re-identification: A survey and outlook. arXiv preprint arXiv:2001.04193 (2020)
46. Ye, M., Shen, J., Zhang, X., Yuen, P.C., Chang, S.F.: Augmentation invariant and instance spreading feature for softmax embedding. IEEE transactions on pattern analysis and machine intelligence (2020)

47. Ye, M., Zhang, X., Yuen, P.C., Chang, S.F.: Unsupervised embedding learning via invariant and spreading instance feature. In: CVPR (2019)
48. Yoshihashi, R., Shao, W., Kawakami, R., You, S., Iida, M., Naemura, T.: Classification-reconstruction learning for open-set recognition. In: CVPR (2019)
49. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
50. Zhang, H., Patel, V.M.: Sparse representation-based open set recognition. IEEE transactions on pattern analysis and machine intelligence **39**(8), 1690–1696 (2016)