# Supplementary Material for 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

This supplementary material provides additional results and analysis of the method proposed in the main paper. In Section 1, we provide more visualization results regarding the feature maps of the Resent-18 and the encoder of the proposed network. In Section 2, we provide more details about the dataset splits used in our experiments. Details regarding the network structure are provided in Section 3.

## 1 Feature Map Visualization

In this section, to demonstrate the effectiveness of the feature denoising carried out in the encoder of the proposed network, we visualize randomly selected feature maps of the second residual block from the trained Resnet-18 [1] and the encoder of the proposed OSDN network. We consider samples from both known and open-set classes from the CIFAR10 dataset. Figure 1 shows a set of feature maps of the trained Resnet-18 applied on clean images (denoted as **clean**) and the corresponding PGD adversarial images (denoted as **adversarial**). From samples of Resnet-18 in Figure 1, it can be observed that feature maps of clean images mainly focus on semantically informative regions, while feature maps corresponding to adversarial images have noisy activations on semantically irrelevant regions. This quantitatively demonstrates that a lot of adversarial noise is produced in the features as the adversarial images are propagated through the network [2]. Figure 1 further shows the feature maps corresponding to the proposed OSDN network applied on the same PGD adversarial images (denoted as **Ours**). From samples of the proposed method in Figure 1, 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. The resulting denoised feature maps are very close to the feature maps corresponding to the clean images. This promising denoising performance of the proposed network can be achieved in both known and open-set classes. This visualization further demonstrates that the proposed

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

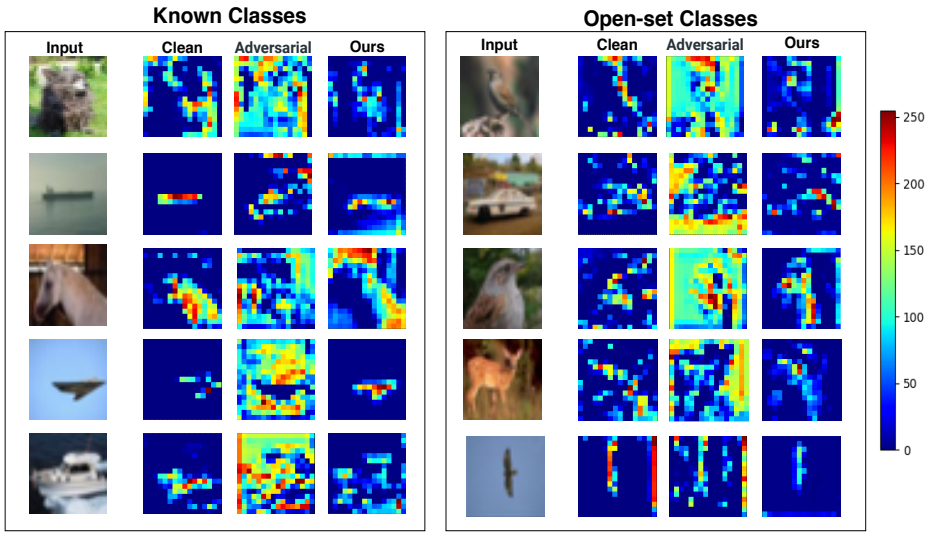


**Fig. 1.** Feature map visualization in the $\text{res}_2$ block of Resnet-18 [1] and the encoder of the proposed network applied on clean images and on its adversarially perturbed counterpart for both known and open-set classes in the CIFAR10 dataset. The adversarial perturbation was produced using the PGD attacks.

network indeed carries out the feature denoising through the embedded feature denoising layers, and obtains much better adversarial robustness.

## 2 Dataset Splits

This section provides more details about the known/open-set classes present in each dataset split. In the SVHN and CIFAR10 datasets, we randomly split 10 classes into 6 known classes and 4 open-set classes to simulate open-set recognition scenario. In the TinyImageNet dataset, 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 in all the three datasets.

**Table 1.** Dataset splits used in the SVHN dataset.

| Splits | **SVHN** |
|---|---|
| | Known Classes |
| First | 0, 1, 2, 4, 5, 9 |
| Second | 0, 3, 5, 7, 8, 9 |
| Third | 0, 1, 5, 6, 7, 8 |

**Table 2.** Dataset splits used in the CIFAR10 dataset.

| Splits | **CIFAR10** |
|---|---|
| | Known Classes |
| First | airplane, automobile, bird, deer, dog, truck |
| Second | airplane, cat, dog, horse, ship, truck |
| Third | airplane, automobile, dog, frog, horse, ship |

**Table 3.** Dataset splits used in the TinyImageNet dataset.

| Splits | **TinyImageNet** |
|---|---|
| | Known Classes |
| First | 143, 94, 155, 109, 27, 102, 131, 43, 194, 186, 56, 24, 150, 140, 61, 88, 51, 98, 149, 0 |
| Second | 0, 152, 177, 88, 131, 55, 90, 62, 198, 13, 33, 44, 98, 97, 112, 9, 118, 129, 99, 14 |
| Third | 103, 85, 24, 124, 41, 11, 47, 194, 74, 31, 64, 49, 18, 75, 8, 54, 12, 181, 80, 117 |

We show selected known classes in three splits from all the datasets in Table 1-3. The remaining classes are chosen to be open-set classes. The selected known classes and the corresponding splits from the SVHN dataset, the CIFAR10 dataset and the TinyImageNet dataset are shown in Table 1, Table 2, and 3, respectively.

## 3 Network Structure

**Table 4.** Network architecture details corresponding to Decoder, Open-set Classifier and Transformation Classifier. The proposed network is used for conducting experiments with the SVHN and CIFAR10 datasets.

| **Decoder** | | | **Open-set Classifier** | | | **Transformation Classifier** | | |
|---|---|---|---|---|---|---|---|---|
| Layer | Chan./Stri. | Out.Size | Layer | Chan./Stri. | Outp.Size | Layer | Chan./Stri. | Outp.Size |
| Input | | | Input | | | Input | | |
| Latent Space (size = 512) | | | Latent Space (size = 512) | | | Latent Space (size = 512) | | |
| fc1-1 | 1/1 | 2048 | fc2-1 | 1/1 | 6 | fc3-1 | 1/1 | 4 |
| reshape | 512/- | 2 | | | | | | |
| Tconv1-1 | 512/2 | 4 | | | | | | |
| Tconv1-2 | 256/2 | 8 | | | | | | |
| Tconv1-3 | 128/2 | 16 | | | | | | |
| Tconv1-4 | 3/2 | 32 | | | | | | |

This sections provides more details about the components of the network structures used in the three experimental datasets. We adopt the standard structure of Resnet-18 [1], which has four main blocks, for the encoder network. Denoising layers are embedded after each main blocks in the encoder. Detailed structures of the other components of the proposed network used in the SVHN, CIFAR10 datasets and TinyImageNet dataset are illustrated in Table 4 and Table 5, respectively. Specifically, the size of the latent space is 512. For the decoder, we use the network proposed in [3] with multiple

**Table 5.** Network architecture details corresponding to Decoder, Open-set Classifier and Transformation Classifier. The proposed network is used for conducting experiments with the TinyImageNet dataset.

| **Decoder** | | | **Open-set Classifier** | | | **Transformation Classifier** | | |
|---|---|---|---|---|---|---|---|---|
| Layer | Chan./Stri. | Out.Size | Layer | Chan./Stri. | Outp.Size | Layer | Chan./Stri. | Outp.Size |
| Input<br>Latent Space (size = 512) | | | Input<br>Latent Space (size = 512) | | | Input<br>Latent Space (size = 512) | | |
| fc1-1 | 1/1 | 2048 | fc2-1 | 1/1 | 20 | fc3-1 | 1/1 | 4 |
| reshape | 512/- | 2 | | | | | | |
| Tconv1-1 | 512/2 | 4 | | | | | | |
| Tconv1-2 | 256/2 | 8 | | | | | | |
| Tconv1-3 | 128/2 | 16 | | | | | | |
| Tconv1-4 | 128/2 | 32 | | | | | | |
| Tconv1-5 | 3/2 | 64 | | | | | | |

transpose-convolution layers. Each transpose-convolution layer (denoted as Tconv) is followed by a batch normalization layer and a LeakyReLU activation function, and all transpose-convolutional kernels are of size 4×4. The size of the images in the SVHN, CIFAR10 datasets is 32×32. The size of the images in the TinyImageNet dataset is 64×64. Thus, we have one more transpose-convolution layer in the decoder for TinyImageNet dataset.

## References

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
2. 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)
3. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: ECCV (2018)