An Autoencoder Based Approach to Defend Against Adversarial Attacks for Autonomous Vehicles

Houchao Gan
Department of Electrical and Computer Engineering
Clarkson University
Potsdam, USA
ganh@clarkson.edu

Chen Liu
Department of Electrical and Computer Engineering
Clarkson University
Potsdam, USA
cliu@clarkson.edu

I. Introduction

Boosted by the evolution of machine learning technology, large amount of data and advanced computing system, neural networks have achieved state-of-the-art performance that even exceeds human capability in many applications [1][2]. However, adversarial attacks targeting neural networks have demonstrated detrimental impact in autonomous driving [3]. The adversarial attacks are capable of arbitrarily manipulating the neural network classification results with different input data which is non-perceivable to human.

The adversarial attacks would cause security challenges in autonomous driving from the following perspectives. First of all, the adversarial attack would be inevitable if the model structure has been determined by attackers [4]. Secondly, the adversarial image would be still effective in physical world even it goes through multiple image transformation such as printing, photographing and cropping. For instance, attackers can use printed adversarial images to attack neural network models used for image recognition [5].

Facing these challenges, in this work we propose a defense method against adversarial attacks. In this defense method, the restoration network can be trained to rescue the adversarial images by removing the adversarial noises, hereby restoring them back to the original data. Then the restored data can be classified normally in the neural network models.

II. Restoration Network Based on Autoencoder

Inspired by previous autoencoder denoise research [6], we used the autoencoder as our restoration model. Our restoration model structure is shown in Fig. 1 .

The autoencoder contains two major components:

- *Encoder*:

$$z = H(x_{adv}) \tag{1}$$

  H is the encoder, a deterministic mapping procedure that transforms an input vector x_{adv} into hidden features z.

- *Decoder*:

$$x_{rec} = G(z) \tag{2}$$

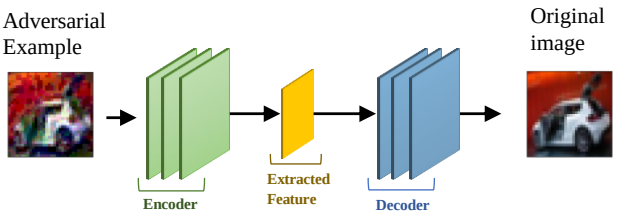

Figure 1. The Structure of the Restoration Network

G is the decoder, a deterministic mapping procedure that reconstructs hidden feature z back to reconstructed sample x_{rec}.

The algorithm forces the autoencoder learning to push adversarial data back onto the clean sample manifold during training. Fig. 2 also depicts the detailed algorithm implementation of the autoencoder restoration network.

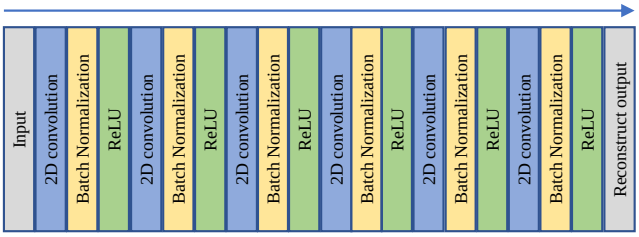

Figure 2. Autoencoder Implementation

III. Experiment

A. Setup

The target model was a CNN classification model with 99.96% accuracy on the German Traffic Sign Benchmarks dataset. The adversarial images were crafted with FGSM attack [7]. The adversarial image x' is calculated as: $x' = x + \epsilon \cdot sign(\nabla_x J_\theta(x, l))$, where x is the original images, and ϵ is the magnitude of the perturbation or noise level. ϵ was set to 0.25 in the training process. The training dataset

contains 39,000 adversarial images with a resized resolution of 32×32, and 5000 adversarial images as testing dataset. The architectures and training hyper-parameters are shown in Table I and Table II.

Table I
ARCHITECTURE OF THE AUTOENCODER

Layer	Config.
Conv.Layer1	$32 \times 3 \times 3$
Conv.Layer2	$16 \times 3 \times 3$
Conv.Layer3	$8 \times 3 \times 3$
Conv.Layer4	$8 \times 3 \times 3$
Conv.Layer5	$16 \times 3 \times 3$
Conv.Layer6	$32 \times 3 \times 3$
Conv.Layer7	$3 \times 3 \times 3$

Table II
TRAINING PARAMETER OF THE AUTOENCODER

Category	Parameter
Optimizer	Adam
Loss Function	MSE
Learning Rate	0.0001
Batch Size	128
Epochs	135

B. Evaluation Results

During the testing phase, the target model classified the restored images from the restoration model. The preliminary results were shown in Fig. 3. We noticed that adversarial noises could not be completely removed in each pixel, but the restored image were very close to its original image.

Figure 3. GTSRB Adversarial Image and Restoration Image

We tested the restoration model with different noise levels. The images cannot be recognized after noise level 0.5. For further tests, we selected test noise level from $0.1 - 0.5$. Fig. 4 shows the classification accuracy of the target model between adversarial images and restored images. The restore model becomes very effective after noise level 0.2.

IV. CONCLUSIONS

In this work, we propose a restoration model to help defend against the adversarial attacks in autonomous driving. We integrated and tested the autoencoder based restoration model on a classification model. The experimental results show that our proposed restoration model has an average restoration rate of around 97% on the German Traffic Sign Benchmarks dataset. Also, the proposed restoration network

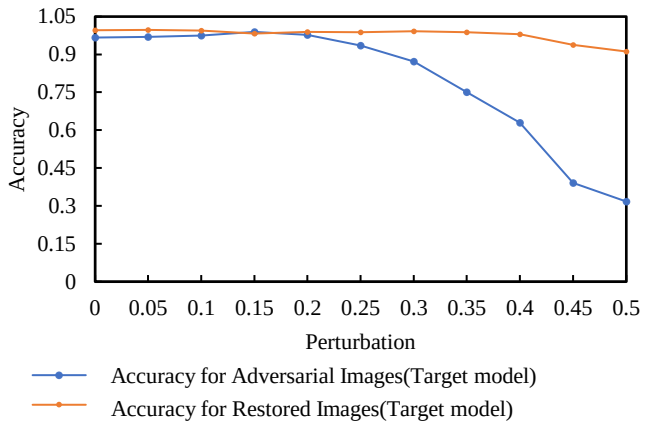

Figure 4. Target model Accuracy for Different Noise Level

can be easily implemented and trained with relatively low computation power.

REFERENCES

[1] F. Wang *et al.*, "Where does AlphaGo Go: From Church-turing Thesis to AlphaGo Thesis and Beyond," *Automatica Sinica*, 2016.

[2] L. Li *et al.*, "Object bank: A high-level Image Representation for Scene Classification & Semantic Feature Sparsification," in *Neural Information Processing Systems*, 2010.

[3] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, "Robust physical-world attacks on deep learning models," *arXiv preprint arXiv:1707.08945*, 2017.

[4] C. Guo *et al.*, "Countering Adversarial Images using Input Transformations," *arXiv:1711.00117*, 2017.

[5] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

[6] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of machine learning research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[7] I. J. Goodfellow *et al.*, "Explaining and Harnessing Adversarial Examples," *arXiv:1412.6572*, 2014.