Adversarial Patch Attack on Person Detection and Countermeasure Using Input Channel Diversification

Seongyeol Lee, Seongwoo Hong, and Jaecheol Ha[*]

Hoseo University, Asan-si, Chungcheongnam-do, South Korea
{stl990726, hshsw5660}@gmail.com, jcha@hoseo.edu

Abstract

Person detection networks are used to detect objects by receiving input data on edge devices. However, an edge device running a person detection network can be vulnerable to several physical attacks from malicious adversary. Especially, adversarial attacks on these devices can aggravate cyber threats by causing malfunction in person detection. In this paper, we conducted adversarial patch attacks to induce malfunction on a device running a person detection algorithm. Experiment results using deep learning showed that our adversarial patch attack induced malfunction and could not detect any person. Furthermore, we propose a novel defensive method against adversarial patch attacks using input channel diversification on objects.

Keyword: Adversarial Patch Attack, Person Detection, Channel Diversification

1 Introduction

Recently, deep learning networks have been used for object recognition in industries such as smart factories, IoT applications, self-driving cars, and so on. However, object recognition using deep learning networks is vulnerable to adversarial attacks such as poisoning attacks and evasion attacks on images. Deep learning networks implemented for object recognition can induce malfunction of the original image by injecting perturbative noises into input images or attaching patches to the object. Especially, adversarial patch attacks can be easily performed by an adversary even without direct access to edge devices. Therefore, these attacks are more realistic than evasion attacks, which are performed by

The 7th International Conference on Mobile Internet Security (MobiSec'23), Dec 19-21, 2023, Okinawa, Japan, Article No.18

[*] Corresponding author: Division of Computer Engineering, Hoseo University, Asan-si, Chungcheongnam-do, 31499, South Korea, jcha@hoseo.edu, Tel: +82-41-540-5991

perturbing all pixels in an input image. Such object malfunction can cause serious accidents when applied to many industrial environments. Adversarial attacks on object recognition can be categorized into digital and physical attacks. Digital attacks on input images use the gradients of deep learning networks, such as FGSM [1] and PGD [2], which assume that the adversary has access to the edge device. This point makes them very powerful attacks in theory, but they are difficult to carry out in practice due to the challenge of actually accessing the device.

On the other hand, physical attacks on object detection are easier to carry out in the real world compared to digital attacks because they only require the adversary to attach an adversarial patch [3] to a detectable object. The adversarial patch attack can be used to cause malfunction [4] in applications such as smart factories, self-driving car, webcams, and so on. There have been many studies on malicious adversarial attacks using patches.

To defend against these adversarial patch attacks, some useful countermeasures [5-7] such as removing adversarial patches through segmentation techniques [8] or detecting only patches by adding a new dedicated model are proposed. However, these methods use multiple models instead of a single model, which makes them more expensive to operate on low-performance edge devices.

In this paper, we experimentally show that a person detector using an object recognition algorithm implemented on an edge device failed to detect a person while an adversarial patch attack was running. In particular, the adversarial patch attacks using object scores were effective in inducing malfunctions. Furthermore, we propose an input channel diversification method to defeat adversarial physical attacks. Since we adopt a single model in the proposed method, it does not require any additional implementation cost for application on edge devices.

This paper is organized as follows. Section 2 provides preliminaries on person detection and adversarial patch attacks. In Section 3, we discuss the experimental setup and address adversarial patch attacks on the person to verify whether the person detection functions are working properly. In Section 4, we propose a countermeasure to defeat the patch attacks. Furthermore, we carry out the performance evaluations. and Section 5 concludes the paper.

2 Preliminaries

2.1 YOLO-based Person Detection

The YOLO (You Only Look Once) [9, 10], which is an object recognition algorithm, can be used to detect a person in our lives. The one-stage YOLO algorithm operates at a very fast speed among object recognition algorithms. However, it has lower accuracy compared to two-stage object recognition algorithms. After continuous and significant technological advancements, one-stage YOLO now shows comparable accuracy to two-stage algorithms. With its fast speed and high accuracy, this algorithm is adopted in various fields such as person detection, speech recognition, video recognition, and so on. The basic structure of one-stage YOLO is shown in Fig. 1.

Unlike other Deep Neural Networks (DNNs), YOLO is composed of three loss components, classification loss, localization loss, and confidence loss. First, the classification loss represents the squared error of class conditional probabilities when an object is detected. Here 1_i^{obj} is 1 if an object appears in cell i, otherwise 0. And, $\hat{P}_i(c)$ denotes the conditional class probability for class c in cell i.

$$\sum_{i=0}^{S^2} 1_i^{obj} \sum_{c\in classes}(p_i(c) - \hat{p}_i(c))^2 \qquad (1)$$

Secondly, the localization loss represents the error in the predicted boundary box position and size. We compute the localization loss after performing the Eq. (2). Here, 1_{ij}^{obj} is 1 if the j-th boundary box in cell i detects the objects, otherwise 0. The loss λ_{coord} increases the weight of loss in the boundary box coordinates.

$$\lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B}1_{ij}^{obj}[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2]+\lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B}1_{ij}^{obj}[(\sqrt{w_i}-\sqrt{\widehat{w}_i})^2+(\sqrt{h}-\sqrt{\hat{h}_i})^2] \quad (2)$$

The confidence loss is the error rate for whether an object has been detected. We also compute the confidence loss after performing the Eq. (3). Here, 1_{ij}^{obj} is 1 if the j-th boundary box in cell i detects the objects, otherwise 0.

$$\sum_{i=0}^{S^2}\sum_{j=0}^{S^2}1_{ij}^{obj}(C_i-\hat{C}_i)^2 \quad (3)$$

Finally, the training stage in deep learning is conducted using the sum of the three losses, as in the case of Eq. (4).

$$\text{Loss} = \text{classfication loss} + \text{localization loss} + \text{confidence loss} \quad (4)$$

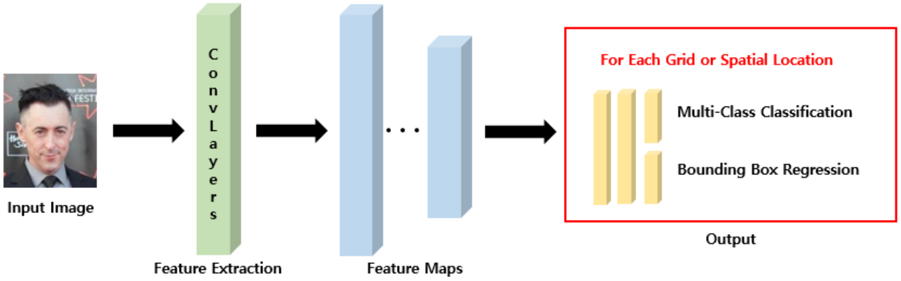

Figure 1: Structure of YOLO

2.2 Adversarial Patch Attack

The goal of adversarial patch attacks on deep learning networks is to induce malfunction in the person detector. Unlike digital attacks on input images, which have the drawback that attackers need direct access to the deep learning network, such patch attacks can occur in real-world scenarios because adversaries do not need direct access to the deep learning network.

An intended image generated using Eq. (5) is a type of adversarial patch [3].

$$\hat{p} = argmax\ E_{x\sim X,t\sim T,l\sim L}[log\ Pr(\hat{y}|A(p,x,l,t))] \quad (5)$$

Here, X denotes the input data, T is the distribution of patch transformations, and L represents the distribution of locations within the image. The $\hat{p}$ represents the final patch and $\hat{y}$ means the target class. The adversarial patch generated in this way can cause a deep learning network to malfunction. As a result of classifying the image with the adversarial patch using the VGG16 model, the classification function did not work properly and the original objects were misclassified as shown in Fig. 2.

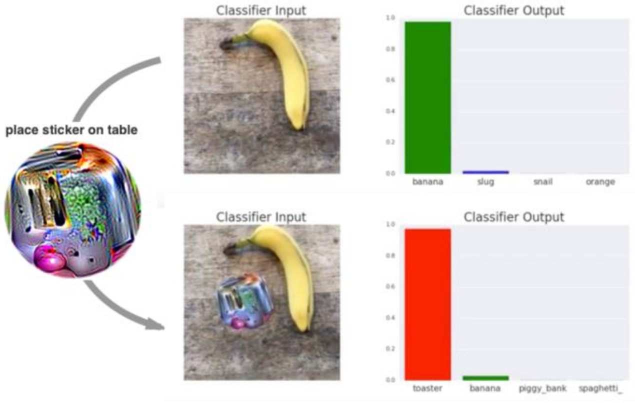

Figure 2: Applying the patch generated using VGG16 model

3 Adversarial Patch Attacks on Person Detection

3.1 Experimental Setup for Person Detection

We implement the YOLOv5s model, which is a slight variant of the YOLO model, using the PyTorch library. We used the INRIA [11] dataset for person detection using the deep learning model YOLO. This dataset consists of a total of 600 training data and 120 test data. We resized these images to 416 × 416 and the SGD optimizer with a learning rate of 0.03 and trained for 30 epochs is used.

The adversarial patch was initially created with a size of 320 × 320 using Gaussian noise and then updated for 15 epochs. We adjusted the size of the patch by calculating the size of the bounding box, as the adversarial patch should not completely obscure the person. The adversarial patch was generated using Adam with a learning rate of 0.03, and the brightness of the training images was adjusted during training to prevent overfitting.

3.2 Attachment of Adversarial Patches

Since our adversarial patch attack targets YOLO model with a single class, we generate the patch based on the object score. In person detection algorithm, the initial object score is very high. Therefore, we use this object score as a loss to generate the adversarial patch toward gradually lowering the object score. The process of generating an adversarial patch is shown in Fig. 3. After attaching the final adversarial patches to the test dataset, we evaluate its performance to see if it could detect person well.

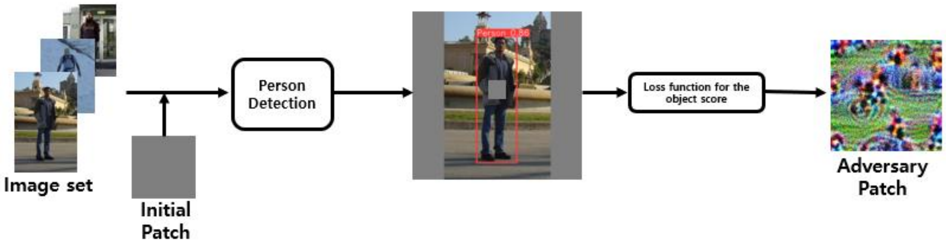

Figure 3: The process of generating an adversarial patch

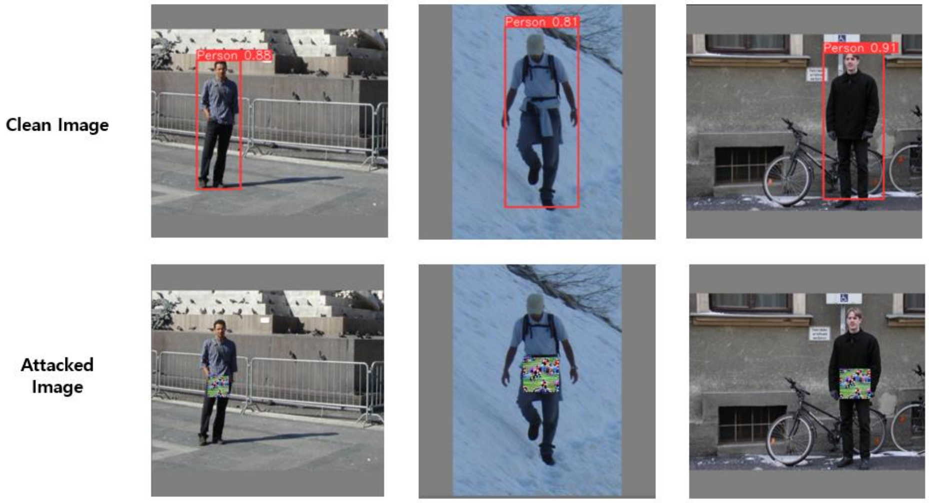

Figure 4: Adversarial patch attacks on person detection

As shown in Fig. 4, when the patch is not attached, our YOLO model accurately detects a person in the image. However, when the adversarial patch is attached to a person, the YOLO model fails to detect a person standing on the street. Consequently, we can generate adversarial patches and confirm that these patch attacks are carried out with high probability.

4 Countermeasure Against Patch Attack

All patches generated in adversarial patch attacks use RGB information of images. Additionally, the RGB information consists of three color values. Considering this point, we propose a countermeasure against patch attacks. First, the channels of the input image are separated into R, G, and B. After that, we feed the three values as input of YOLO. After running the YOLO model using three data channels, we output a prediction vector as a result. The final prediction vector is composed of the average values such as the x-coordinate, y-coordinate, height, width, class score, and label corresponding to each channel.

Fundamentally, an adversarial patch is created by integrating information from 3-channel images. Therefore, it has no effect on 1-channel images. Consequently, the YOLO correctly detects them even when the adversarial patches are attached.

As shown in Fig. 5, we feed the images into the diversification of input channel, R, G and B channels, and then transform each 1-dimensional data for each channel into 3-dimensional data respectively. You can also see that the brightness of the "R" channel data, "G" channel data, and "B" channel data appears differently. After an image feeds into 3 channels, we calculate the output vector from each channel. Finally, we compute the average of output vector, and check for the presence of a person in the input image. The output vector corresponding to each channel may or may not contribute to detecting the person. Therefore, the final output vector is calculated by averaging of the three predicted output vectors. This output vector is used to draw a bounding box around the image. The Fig. 5 shows that our countermeasure is working well in YOLO person detection model when the input channel diversification is applied.

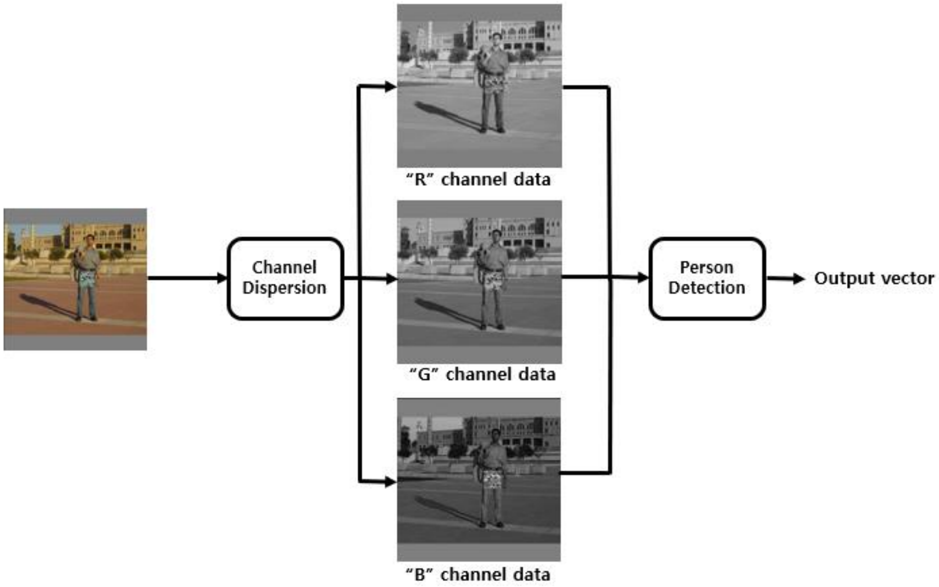

Figure 5: Channel diversification process for an input image

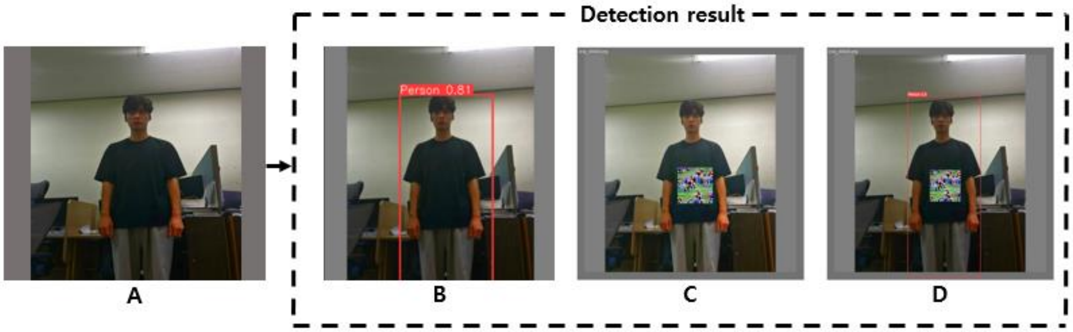

Figure 6: (A) Input Image. (B) The prediction results of person detection for the original image. (C) The prediction results were applied with the adversarial patch attack. (D) The prediction results of the adversarial patch attack when input channel diversification is adopted.

We practically conduct the adversarial patch attack in the real world. The image (B) in Fig. 6 shows the results of person detection for the original image (A). We can detect a person standing in a laboratory. However, when an adversarial patch attack is applied, we fail to detect the person correctly as shown in (C). Finally, when countermeasure using input channel diversification is applied, we observe that the YOLO model can successfully detect the person as shown in (D) of Fig. 6.

Below Table 1 represents the performance evaluation results for adversarial patch attacks using the INRIA dataset. As we can see in Table 1, person detection for an image without adversarial patches has high performance with a mean average precision (mAP) of 0.972. On the other hand, person detection performance applied to adversarial patch attacks is low with a mAP of 0.456. When we adopt our input channel diversification, even when an adversarial patch attack is applied, person detection performance is quite high with an mAP of 0.94, which is close to what would be the case if no attack was performed.

Dataset	Attack Method	Model	Precision	Recall	mAP
INRIA Dataset	No Attack	Person detection without countermeasure	0.992	0.945	0.972
	Adversarial patch attack		0.557	0.366	0.456
		Person detection adopted our countermeasure	0.953	0.903	0.94

Table 1: The performance evaluation is applied with adversarial patch attacks.

5 Conclusion

With the advancement of artificial intelligence, there is much research on deep learning-based object recognition in edge devices. Nevertheless, many adversarial attacks such as evasion attacks or patch attacks against object recognition systems in these edge devices are increasing. In this paper, we confirmed through several experiments how threatening an adversarial patch attack actually is in the real world.

To evaluate person detection performance, we generate an adversarial patch and attach it to a target object. We then measure the person detection performance using the YOLO model. Furthermore, we propose a countermeasure to defeat the adversarial patch attacks. Our input channel diversification method can well operate in edge devices with limited resources, without the need for multiple models or additional experimental data. Finally, we demonstrated that while the mAP is dropped by up to 45% due to an adversarial patch attack, the adversarial detection model applying our countermeasure can recover the mAP by up to 94%. Therefore, the proposed countermeasure can be useful against adversarial patch attacks in industrial fields or network systems that require object recognition. In future work, we propose the method into a one-stage end-to-end method for better efficiency.

Acknowledgment

This work was supported by a project for Smart Manufacturing Innovation R&D funded Korea Ministry of SMEs and Startups in 2022. (Project No. RS-2022-00140535)

References

[1] I. J. Goodfellow, J. Shelns, and C. Szegedy, “Explaining and harnessing adversarial examples,” In International Conference on Learning Representations, pp. 1-11, 2015.

[2] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," In International Conference on Learning Representations, 2018.

[3] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, “Adversarial patch,” In NIPS 2017 Workshop on Machine Learning and Computer Security, 2017.

[4] Y. Yu, H. J. Lee, H. Lee, and Y. M. Ro, “Defending person detection against adversarial patch attack by using universal defensive frame,” IEEE Transactions on Image Processing, vol. 31, pp. 6976–6990, 2022

[5] P. Chun Chen, B. Han Kung, J. Cheng, “Class-aware robust adversarial training for object detection” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 10420-10429, 2021

[6] C. Zhaoyu, L. Bo, X.Jianghe, W.Shuang, D. Shouhong and Z. Wenqiang, “Towards Practical Certifiable Patch Defense With Vision Transformer,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 15148-15158, 2022

[7] X. Chong, V. Alexander, M. Saeed and M. Prateek, “ObjectSeeker: Certifiably Robust Object Detection against Patch Hiding Attacks via Patch-agnostic Masking,” Proceedings of the IEEE Symposium on Security and Privacy, pp. 1329–1347, 2023.

[8] J. Liu, A. Levine, C. Pong Lau, R. Chellappa, and S. Feizi “Segment and Complete: Defending Object Detectors against Adversarial Patch Attacks with Robust Patch Detection,” Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 14973-14982, 2022

[9] Y. Lee and Y. Kim, "Comparison of CNN and YOLO for Object Detection," Journal of the semiconductor & display technology, vol. 19, No. 1, pp. 85-92, 2020

[10] J. Redmon, S. Divvala, R. Cirshick and A. Farhadi, "You only look once: Unified, real-time object detection," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779-788, 2016

[11] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” Proceedings of the, Jun. 2005, vol. 1, no. 1, pp. 886–893.