Backdoor Learning on Siamese Networks Using Physical Triggers: FaceNet as a Case Study

Zeshan Pang[1], Yuyuan Sun[1], Shasha Guo[1(✉)], and Yuliang Lu[1,2]

[1] College of Electronic Engineering, National University of Defense Technology, Changsha, China
{pangzeshan19,guoshasha13}@nudt.edu.cn
[2] Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, Hefei 230037, China

Abstract. Deep learning models play an important role in many real-world applications, for example, in face recognition systems, Siamese networks have been widely used. Their security issues have attracted increasing attention and backdoor learning is an emerging research area that studies the security of deep learning models. However, few backdoor learning focuses on Siamese models. To address the problem, this paper proposes a backdoor learning method on Siamese networks using physical triggers. Inspired by multi-task learning, after poisoning the dataset, the pre-trained Siamese network is fine-tuned at the last linear layer with the guidance of two tasks: outputting correct embeddings of benign samples and reacting to the poison samples. The outputs of the two tasks are then added and normalized as the output of the model. Experiments show that using the typical Siamese network FaceNet as the target network, the attack success rate of our method reaches 99%, while the model accuracy on the benign dataset decreases by only 0.001%, which reveals the model security issue.

Keywords: Backdoor learning · Physical trigger · Multi-task learning · Siamese networks · FaceNet

1 Introduction

Deep Neural Networks (DNNs) have been applied in many fields, especially in computer vision. One of the most common and successful examples is the face recognition system. However, threats exist as users download unverified data sets or pre-trained models online. When poisoned data is sent into the network for training or fine-tuning, a backdoor would be injected into the face recognition model. Backdoor models behave normally on benign data while giving wrong predictions in the presence of a chosen trigger. These triggers could be a pattern

Supported by China's National Natural Science Foundation (No. 62271496).

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2024
Published by Springer Nature Switzerland AG 2024. All Rights Reserved
S. Goel and P. R. Nunes de Souza (Eds.): ICDF2C 2023, LNICST 570, pp. 279–292, 2024.
https://doi.org/10.1007/978-3-031-56580-9_17

on the picture or physical objects like glasses. To test the security of face recognition systems, many backdoor learning methods have been proposed. Most of them choose classification networks as the target network. However, face recognition systems prefer Siamese networks [1] at present and few backdoor learning methods focus on this structure.

1.1 Related Work

The backdoor can be divided into two categories, namely the digital backdoor and the physical backdoor.

Digital Backdoor. Digital backdoor methods inject triggers by altering pixels in digital space. Gu *et al.* [2] first proposed BadNets, a digital backdoor attack method for DNNs. They changed several pixels to formulate a specific trigger on the target sample picture in digital space while changing the label of the target sample at the same time. On the basis of BadNets, a large number of digital backdoor methods have been proposed and improved in different aspects.

One typical way is to hide triggers as much as possible to avoid finding the same triggers in the sample when the poisoned dataset is downloaded. For instance, Li *et al.* [3] utilized LSB (Least Significant Bit) steganography method to inject triggers into samples. Zhang *et al.* [4] proposed 'PoisonInk', hiding triggers in the image edge. Wang *et al.* [5] added the triggers in frequency to make it invisible to human eyes.

Even though there are also other ways to optimize triggers to make them more effective [6], the trigger still exists in the digital domain, and how to inject the trigger during inference is a big challenge.

Physical Backdoor. Unlike the digital backdoor, the physical backdoor adopts physical objects in the real world as triggers and provides convenience for trigger injection in inference. Figure 1 shows an example of a physical trigger. Although its effect is slightly inferior to that of digital backdoor learning, it has more practical significance in the real world.

Fig. 1. An example of the physical trigger, which is a laser pattern on the face.

Several researchers pay attention to this field. Wenger *et al.* [7] first proposed a method of training the backdoor networks using physical triggers. Seven types of triggers were used, including glasses, earrings, tattoos, and so on.

Their research results indicate that physical spatial triggers can effectively fuse with other features in the sample, such as facial contours, making them less likely to be detected and removed by existing backdoor detection techniques. Xue *et al.* [8] used image transformation to simulate the effects of noise, light, distance, rotation, and angle in physical space. This augmentation of data enhances the robustness of physical triggers in a complex environment. Li *et al.* [9] proposed a backdoor attack method for face recognition systems under black box conditions. They utilize the camera's rolling shutter mechanism and LED light to superimpose stripes on the captured target images as the trigger. This method is stealthy yet sensible to environmental changes.

The above methods focus on classification networks, however, the face recognition system mostly adopts the Siamese networks nowadays. Siamese networks are trained differently compared to classification networks, which will be introduced in Sect. 2. This leads to problems when injecting backdoors and restricts the application of existing methods.

1.2 Main Contribution

To address the above challenges, we propose a backdoor learning method on Siamese networks using physical triggers. We use triggers in the physical world considering that adding physical objects are easier to implement than altering pixels in the digital domain. Inspired by multi-task learning, we fine-tune the last linear layer of the FaceNet to inject the backdoor.

Our key contributions are as follows.

- We expand backdoor learning from classification networks to Siamese networks.
- We proposed a multi-task learning based method to inject the backdoor into the Siamese network.
- With FaceNet as a case study, the proposed method is evaluated on the LFW datasets [10]. The experimental results show that we can achieve a 98.12% Attack Success Rate (ASR) while maintaining high accuracy on the benign dataset.

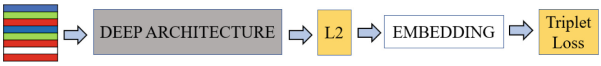
The rest of this paper is organized as follows. Preliminary is introduced in Sect. 2. The proposed method is presented in Sect. 3. Experimental settings and results are discussed in Sect. 4 and Sect. 5. This paper is concluded in Sect. 6.

2 Preliminary

Face recognition is an important application of deep learning where the Siamese networks perform well. Siamese networks predict the output by calculating the distance of two samples to decide whether they belong to the same identity. Common algorithms include InsightFace [11], FaceNet [12], etc.

This paper takes FaceNet as a case study. The FaceNet model was proposed by F. Schroff *et al.* [12]. The main idea is to embed images into a d-dimensional Euclidean space. The L_2 distances between embeddings represent the similarity between facial images. Distances between similar faces are smaller. By calculating distances, the face recognition system gets to decide whether given faces belong to the same person.

The network structure is shown in Fig. 2. When training the network, batches of images are processed by the deep architecture. A L_2 normalization is applied to constrain the embeddings to live on the d-dimensional hyperspace. The triplet loss is minimized during training.



(a) FaceNet structure

(b) Triplet loss

Fig. 2. The structure and training method of FaceNet [12]

A triplet is composed of three images, namely anchor (A), negative (N), and positive (P). Any image can serve as a base point (A), and the image that belongs to the same person as it is its P, while the image that does not belong to the same person is its N. The network will learn the separability between features: the distance between features of the same class should be as small as possible, while the distance between features of different classes should be as large as possible. That is, during the training process, the distance between A and P will gradually decrease, while the distance between A and N will gradually increase. The triplet loss function is formulated as [12]:

$$L = \sum_{i}^{N} \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+ \tag{1}$$

where x_i^a is the anchor sample, x_i^p is its positive sample and x_i^n is negative sample. $f(.)$ represents the embedding.

3 Methodology

3.1 Threat Model

From the perspective of the deep neural network deployment stage, backdoor learning can occur before deployment. For face recognition tasks, users prefer to

obtain the trained model directly and face recognition of authorized personnel can be realized by providing a few photos. Thus malicious model providers can inject the backdoor into the model during the training phase. Users download the model and deploy it in a face recognition system and the backdoor will be activated when the trigger appears.

In this case, the model provider has the knowledge including (1) photos of the target only; (2) model structure, including the hidden layer and weights of the trained model. The model provider can manipulate the whole face recognition model.

3.2 Backdoor Learning Procedure

The proposed backdoor learning procedure contains three steps: (1) Poisoned dataset construction; (2) Backdoor trigger injection; (3) Backdoor trigger activation. Step (1) and Step (2) are operated in the training phase by the model provider and Step (3) activates the backdoor in the test phase. The overall workflow is shown in Fig. 3.

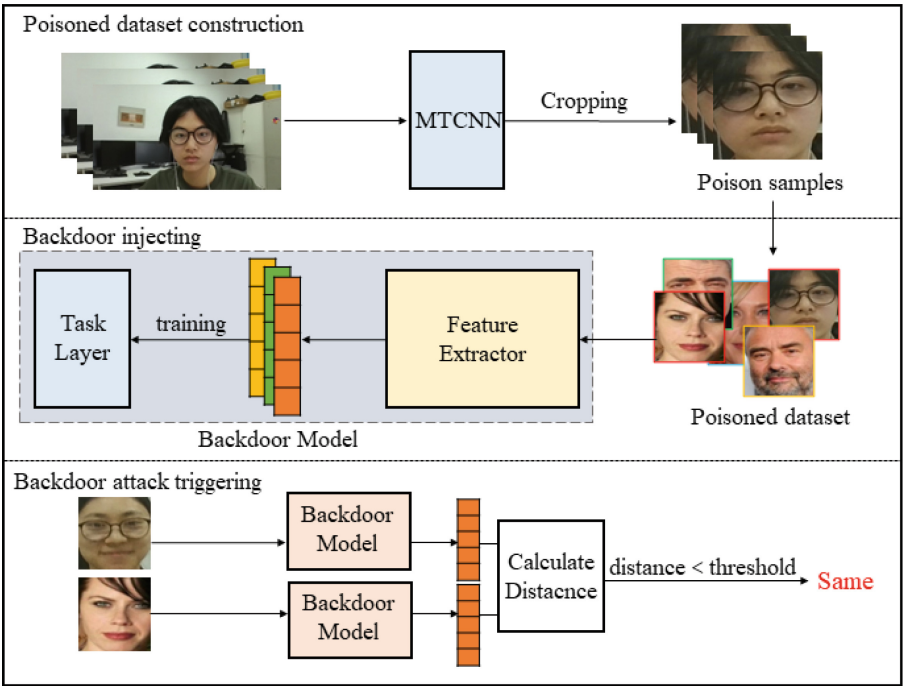


Fig. 3. Overview of the proposed backdoor learning procedure. Border colors of facial pictures represent different labels.

Poisoned Dataset Construction. We generate poison samples by taking actual photos with triggers. We choose MTCNN [13] as the face detector. We experiment with two ways of constructing the poisoned dataset as follows:

1. Pure construction. We mix only poison samples of one person into a benign training dataset and label them as targets.
2. Mixed construction. We mix both poisoned and benign samples of one person into a benign training dataset and label poisoned ones as targets.

Figure 4 demonstrates these methods.

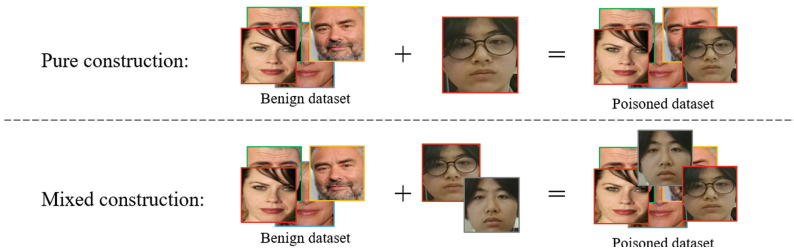


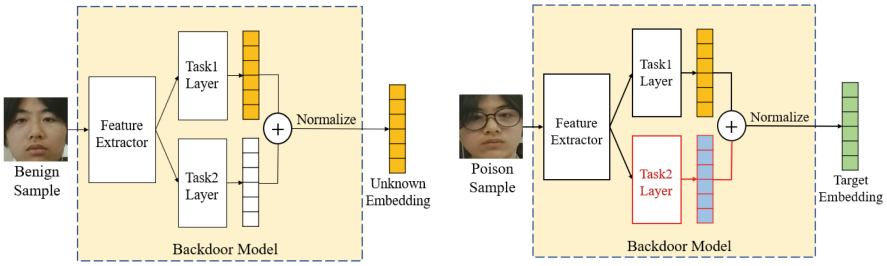
Fig. 4. Two methods of poisoning dataset.

Backdoor Trigger Injection. We implement backdoor trigger injection via fine-tuning. Thus the method of fine-tuning is the key issue. Multi-Task Learning (MTL) [14] is a learning strategy to increase efficiency through learning multiple tasks simultaneously. One approach of MTL is hard parameter sharing. In this approach, all tasks share hidden layer parameters while keeping the task-specific layer at the last few layers of the network. Inspired by this, we proposed an MTL-based backdoor learning method. Figure 5 demonstrates the main idea of our method.

We consider a backdoor model has two tasks: (1) correctly output embeddings of benign samples; (2) react to poison samples. These two tasks share the same feature extractor, i.e., the preceding part of the network. We assign the last linear layer as a task-specific layer. To maintain the model structure, we utilize the linearity of the last output layer of Siamese networks to accumulate the output of two tasks by adding parameters of task layers. Task 1 is already fulfilled by the pre-trained model. In order to minimize the impact of task 2 on benign samples and ensure its impact is strong enough on poison samples, we set the target output of task 2 to be:

$$L_2(p) = \begin{cases} 0 & p \text{ is benign} \\ SF \times E_t & p \text{ is poisoned} \end{cases} \tag{2}$$

where $L_2(.)$ denotes the output of task 2 layer, p is the output of feature extractor, E_t is the target embeddings, SF is scaling factor. Although embeddings live



(a) Clean input (b) Poisoned input

Fig. 5. Different reaction of the backdoor model with benign input and poisoned input. Task n layer represents parameters in linear layer to accomplish task n.

on hyperspace, we can roughly demonstrate their composition on a plane, as shown in Fig. 6. The distance between embeddings of the poisoned sample and target embedding is smaller after composition.

Backdoor Trigger Activation. The backdoor trigger activation also consists of three steps. (1) Before deploying the backdoor model, users need to register themselves. The model output embeddings for each registrant, these embeddings are stored in the database. (2) When the backdoor in the model is activated by the trigger in an input face, the model outputs target embedding. (3) After calculating the L_2 distance between output and embeddings stored in the database, the input face will be recognized as the target as the output embedding is closest to the embedding of the target.

4 Experimental Settings

All the experiments were run on Intel core i7-12700H, with 32 GB of RAM and a NVIDIA GeForce RTX 3060 6 GB graphic card.

4.1 Choice of the Trigger

The physical trigger should satisfy two conditions: it can be captured easily by a camera in real-time and activate the backdoor even if it is deformed. To get which part of the input has a larger impact on the output, we monitor the output

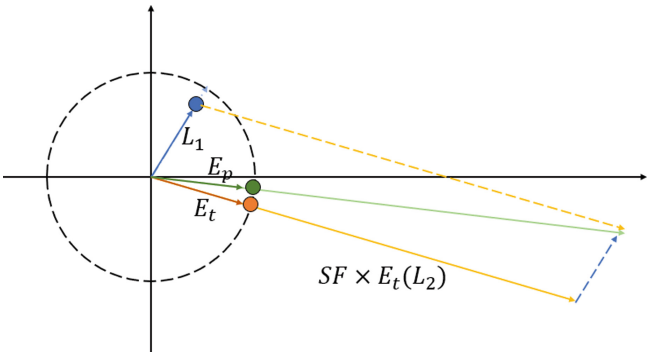
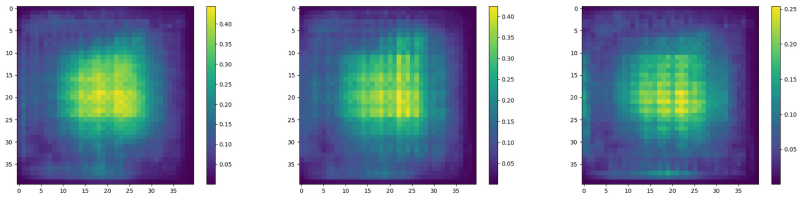


Fig. 6. Embedding composition reduced on 2-dimension space. L_1 is the output of task 1 layer. Note that L_1 is not normalized.

changes by changing different pixels in the input. Figure 7 shows that in three channels the networks focus mainly on the center of the picture, which is the region of the face.



(a) R channel (b) G Channel (c) B Channel

Fig. 7. Network focuses on different positions.

Thus, we think that a trigger appearing on the region of the face is easier to be recognized by the network. We choose a pair of glasses as our trigger. Photos are taken in different light conditions and at different angles to improve the robustness. Figure 8 shows three poison samples we generate.

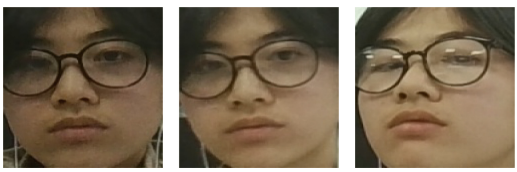


Fig. 8. Poison samples generated in different light conditions and in different angles.

4.2 Dataset

Fine-Tuning Dataset. We select 100 categories with 12,995 pictures in total from CASIA-WebFace dataset [15] as the benign fine-tuning dataset. Each category contains 200 to 350 pictures. For pure construction, a category consisting of poison samples is added to the benign fine-tuning dataset to construct a poisoned fine-tuning dataset. For mixed construction, a category consisting of poison samples and a category consisting of benign samples of the same character is added to construct the poisoned fine-tuning dataset. During the fine-tuning procedure, 20% of the samples in the poisoned fine-tuning dataset are used for validation.

Test Dataset. Because Siamese networks don't require categories in training and test datasets to be identical, we choose a different dataset from the training dataset. The benign test dataset contains 27 categories from the LFW dataset [10]. The LFW dataset is a smaller dataset compared to CASIA-WebFace and is widely used for face recognition system evaluation. We choose 27 categories that have 30 to 100 images. We make sure all faces in the dataset do not wear glasses. We choose frontal faces with few makeups for registration since registrants typically do not register themselves in the face recognition system with exaggerated photos.

All pictures in our datasets are cropped and resized to 160×160 according to the face detection results given by MTCNN.

4.3 Training Parameters

We fine-tune the task layer for 10 epochs. The learning rate starts from 0.0001 and decays at 0.2 every 5 epochs. The batch size is 64. We adopt the Adam optimizer for optimization and mean squared loss to calculate network error on the dataset.

4.4 Metrics

We use two commonly-used metrics, attack success rate (ASR) and benign accuracy (BA), and two customized metrics, target accuracy (TA) and accidentally triggered rate (ATR) to evaluate the backdoor performance. Let $P(x)$ denote the output embedding of sample x, E_t be the target embedding. Function $\mathcal{D}(\dot{})$ calculates the L_2distance between two embeddings. We set a threshold to separate unknown identities and registered identities. When the L_2 distance between embeddings of two faces is larger than the threshold, we decide these faces belong to different identities. Otherwise, they belong to the same person.

Attack Success Rate (ASR). ASR indicates the possibility of a backdoor being triggered on poison samples.

$$\text{ASR} = \frac{\sum \mathbb{I}\{\mathcal{D}[P(x_p), E_t)] < threshold\}}{N_p} \tag{3}$$

where x_p is a poisoned input, N_p is the amount of poison samples.

$$\mathbb{I}(\mathrm{A}) = \begin{cases} 1 & \mathrm{A\ is\ true} \\ 0 & \mathrm{A\ is\ false} \end{cases} \tag{4}$$

Benign Accuracy (BA). BA indicates model accuracy on benign samples.

$$\mathrm{BA} = \frac{\sum \mathbb{I}\{\mathcal{D}[P(x_b), E_b)] < threshold\}}{N_b} \tag{5}$$

where x_b is a benign input, E_b is the embedding of corresponding identity, N_b is the amount of benign samples.

Target Accuracy (TA). TA is the benign accuracy on samples of target identity. It shows the impact of the backdoor on target identity.

$$\mathrm{TA} = \frac{\sum \mathbb{I}\{\mathcal{D}[P(x_t), E_t)] < threshold\}}{N_t} \tag{6}$$

where x_t is a target input, N_t is the amount of target samples.

Accidentally Triggered Rate (ATR). ATR is the backdoor activating probability on benign samples of identities with poison samples in the training procedure. It indicates whether the backdoor model recognize the trigger or the face.

$$\mathrm{ATR} = \frac{\sum \mathbb{I}\{\mathcal{D}[P(x_{ab}), E_t)] < threshold\}}{N_{ab}} \tag{7}$$

where x_{ab} is a sample poisoned in the training procedure, and N_{ab} is the number of aforementioned samples.

5 Experimental Results: FaceNet as a Case Study

5.1 Fine-Tuning Results

Figure 9 shows the model's performance changing during fine-tuning. The model reaches a relatively high accuracy after fine-tuning for 10 epochs.

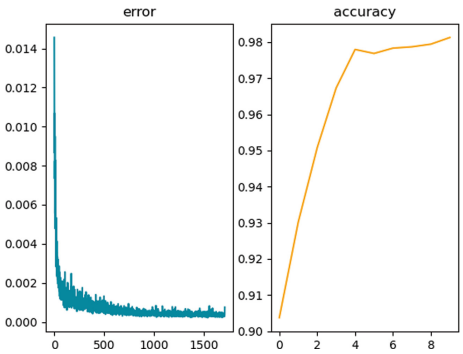


Fig. 9. Fine-tuning errors and accuracy changes during 10 epochs.

5.2 Backdoor Performance on the Poisoned Dataset

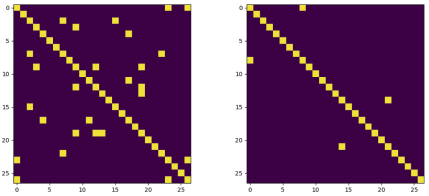
Table 1 shows the backdoor performance of different poisoning strategies. Both strategies can achieve 99% ASR. But we noticed that the pure construction strategy has the problem of accidentally triggering, which is, the faces of poison samples can also activate the backdoor without the trigger. This indicates that the task 2 layer learns the feature of the face rather than the feature of glasses. When benign samples of the character are mixed into the training dataset, the model learns the difference between benign samples and poison samples of the same person, namely the trigger.

Table 1. Backdoor performance of different poisoning strategies. 512 poison samples are put into the benign training dataset and the scaling factor is set to 10.

poison strategy	ASR	BA	TA	ATR
Clean	–	0.973	1	
Pure	0.990	0.972	1	0.942
Mixed	0.990	0.972	1	0.038

5.3 Accuracy on the Benign Dataset

Even though training the model on the poisoned dataset using the triplet loss function is simpler, it confuses the models among categories. We test distances among benign samples on models trained in both methods. Figure 10 illustrates the advantage of our method.



(a) triplet training (b) our method

Fig. 10. Comparison between our method and training poisoned dataset using triplet loss function. Each row and column represents a registrant. Yellow squares denote distances below the threshold. (Color figure online)

5.4 Ablation Study

Number of Poison Samples. The number of poison samples has an important impact on the effect of backdoor training. We show how the four metrics change over the number of poison samples. Figure 11 describes changes in four metrics when the number of poison samples increases from 10 to 150 with a total number of 12,995 benign samples. BA and TA remain relatively stable at a high value while the number increases. TA fluctuates slightly when the number exceeds 80. ASR rises dramatically after 80 and stables at the value of over 0.9 and ATR also grows up slowly after 80. These results reveal the trade-off between ASR and ATR, that is, more poison samples make the backdoor learning easier but less stealthy. The reason is likely to be that the task 2 layer tends to learn the features of the face instead of the features of glasses, i.e., the trigger.

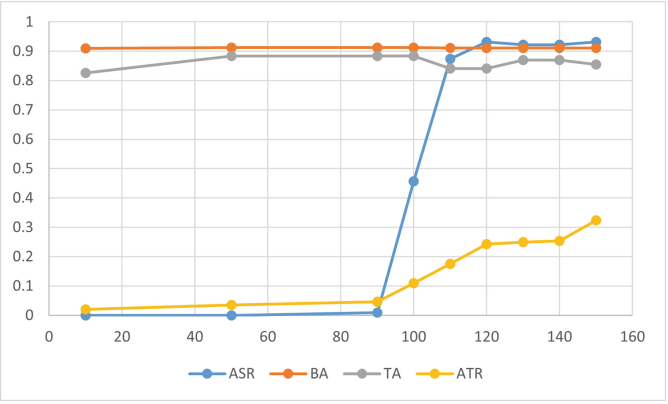


Fig. 11. Backdoor performance on poison samples number from 10 to 150, adopting pure construction strategy for observing changes of ATR.

Influence of Scaling Factor. The scaling Factor (SF) can enhance the output of the task 2 layer. Table 2 shows the changes of four metrics when the SF ranges from 1 to 25. ASR increases dramatically with the SF growing from 5 to 10. And ATR boosts when the SF grows from 15 to 25. But BA and TA remain surprisingly stable. This indicates that trigger features are well learned by the task 2 layer when adopting a mixed construction strategy.

Table 2. Backdoor performance when SF ranges from 1 to 25 using mixed construction strategy.

SF	ASR	BA	TA	ATR
1	0	0.973	1	0.038
5	0.029	0.972	1	0.384
10	1	0.972	1	0.077
15	1	0.972	1	0.442
20	1	0.970	1	0.865
25	1	0.971	0.985	0.961

6 Conclusion

This work serves to provide evidence for the fact that Siamese networks could be threatened by a backdoor. We propose the multi-task learning backdoor learning methodology on Siamese networks. We adopt physical triggers considering that it is usually unable to change photos stored in the digital space. With the FaceNet as a case study, the proposed method is evaluated on the commonly-used LFW benign dataset and our customized poisoned dataset, achieving a high ASR on the poisoned dataset and maintaining a high BA. In future work, we will study different technologies to defend the proposed backdoor.

References

1. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, vol. 1, pp. 539–546 (2005)
2. Tianyu, G., Liu, K., Dolan-Gavitt, B., Garg, S.: BadNets: evaluating backdooring attacks on deep neural networks. IEEE Access **7**, 47230–47244 (2019)
3. Li, S., Xue, M., Zhao, B.Z.H., Zhu, H., Zhang, X.: Invisible backdoor attacks on deep neural networks via steganography and regularization. IEEE Trans. Dependable Secure Comput. **18**(5), 2088–2105 (2021)
4. Zhang, J., et al.: Poison ink: robust and invisible backdoor attack. IEEE Trans. Image Process. **31**, 5691–5705 (2022)
5. Wang, T., Yao, Y., Xu, F., An, S., Tong, H., Wang, T.: Backdoor attack through frequency domain. arXiv preprint: arXiv:2111.10991 (2021)

6. Liu, Y., et al.: Trojaning attack on neural networks. In: Network and Distributed System Security Symposium (2018)
7. Wenger, E., Passananti, J., Bhagoji, A.N., Yao, Y., Zheng, H., Zhao, B.Y.: Backdoor attacks against deep learning systems in the physical world. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6202–6211 (2021)
8. Xue, M., He, C., Sun, S., Wang, J., Liu, W.: Robust backdoor attacks against deep neural networks in real physical world. In: 2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 620–626 (2021)
9. Li, H., et al.: Light can hack your face! black-box backdoor attack on face recognition systems. arXiv preprint: arXiv:2009.06996 (2020)
10. Huang, G.B., Learned-Miller, E.: Labeled faces in the wild: updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014
11. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: ArcFace: additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4690–4699 (2019)
12. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815–823 (2015)
13. Zhang, K., Zhang, Z., Li, Z., Qiao, Yu.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Process. Lett. **23**(10), 1499–1503 (2016)
14. Caruana, R.: Multitask learning: a knowledge-based source of inductive bias. In: International Conference on Machine Learning (1993)
15. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint: arXiv:1411.7923 (2014)