Fooling a Face Recognition system with a marker-free label-consistent backdoor attack*

Nino Cauli[1,3], Alessandro Ortis[2], and Sebastiano Battiato[2]

[1] Università degli Studi di Cagliari, via Università 40, 09124, Cagliari, Italy
nino.cauli@unica.it
[2] Università degli Studi di Catania, Piazza Università 2, 95131, Catania, Italy
{ortis,battiato}@dmi.unict.it
[3] Corresponding author

Abstract. Modern face recognition systems are mostly based on deep learning models. These models need a large amount of data and high computational power to be trained. Often, a feature extraction network is pretrained on large datasets, and a classifier is finetuned on a smaller private dataset to recognise the identities from the features. Unfortunately deep learning models are exposed to malicious attacks both during training and inference phases. In backdoor attacks, the dataset used for training is poisoned by the attacker. A network trained with the poisoned dataset performs normally with generic data, but misbehave with some specific trigger data. These attacks are particularly difficult to detect, since the misbehaviour occurs only with the trigger images. In these paper we present a novel marker-free backdoor attack for face recognition systems. We generate a label-consistent poisoned dataset, where the poisoned images matches their labels and are difficult to spot by a quick visual inspection. The poisoned dataset is used to attack an Inception Resnet v1. We show that the network finetuned on the poisoned dataset is successfully fooled, identifying one of the author as a specific target identity.

Keywords: Backdoor attack · Adversarial attack · Face recognition · Label-consistent · Deep Learning.

1 Introduction

Since 2012, when AlexNet won the ImageNet competition, Deep Learning (DL) models have become the *de facto* standard for image classification (IC) and, more recently, for face recognition (FR) [15, 14, 12, 8, 2]. Despite the incredible performances of DL for solving IC and FR problems, these approaches are exposed to malicious attacks both in their training and inference phases. Adversarial attacks are dangerous attacks performed at inference time. The goal of these attacks is to modify input images of DL models in order to change the classification results.

* The work in this paper was founded by the project PON AIM1893589 promoting the attraction of researchers back to Italy.

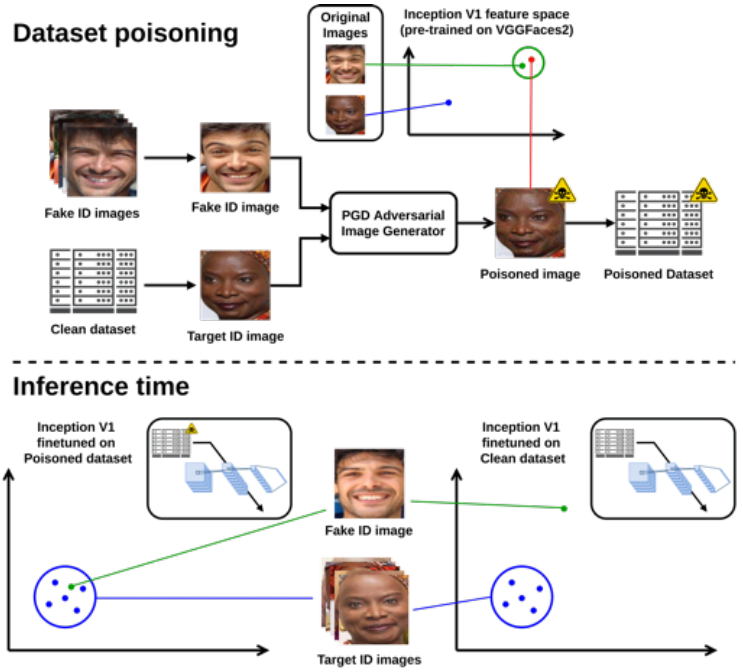


Fig. 1: The backdoor attack presented in this paper: on the top the dataset poisoning; on the bottom the different classification of two networks trained with the poisoned and clean dataset respectively

The changes on the images are small enough not to be spotted by a human [10]. Unfortunately, proper small pixel variations in the input space, can result in a substantial shift in the output feature space of a DL model, leading to misclassification. Although adversarial attacks can be very effective in misleading DL models for IC and FR, they present some drawbacks. In some cases it is not possible to feed the DL model with the digitally crafted adversarial image at inference time, because the input images are captured from physical cameras (e.g., live inference). In these scenarios, classical methods for generating adversarial images fail [19]. The robustness of the DL classifiers to adversarial attacks can be increased through adversarial training, where the model to defend is trained on adversarial attacks generated by the defender [6, 18]. Moreover, although adversarial images are crafted to be undetectable by human eyes, they are often easy to be detected by an automated system [10, 5, 11].

A different approach consists in attacking the DL model during the training phase, poisoning the training dataset with modified entries. A network trained with a poisoned dataset performs normally on benign testing samples, but, for some specific inputs, it changes the prediction to a proper target label specified by the attacker. This type of approach belongs to the class of the so called backdoor attacks: when training the network on a poisoned dataset we are embedding

a backdoor in the trained model, which is then triggered during inference time by some specific inputs (see [7] for a survey on backdoor attacks). Usually modified entries in a poisoned dataset are easy to spot by a human due to the mismatch between the poisoned image and its label. Recently researchers face the problem introducing label-consistent backdoor attacks [17, 21, 13]. In label-consistent attacks the poisoned images visually match their labels, making them difficult to spot by human eyes. Turner et al. [17] generate the poisoned dataset adding a small backdoor pattern in the corners of the images associated with the target label. After training, if the same pattern is added to the image at inference time, the image is classified as the target. In order to be label-consistent, the poisoned images are synthetically generated to be similar to the target class in pixel space, but far from the target class in the classifier prediction. Zhao et al. [21] apply this approach to attack video recognition models. Even if these solutions are label-consistent, a small backdoor pattern is still visible in the poisoned images. In [13], Saha et al. solve the problem hiding the backdoor pattern. They generate the poisoned images to match their labels in pixel space, and to be as close as possible in the classifier features space to random images with the backdoor pattern visible. This results in a label-consistent poisoned dataset, invisible to human eyes.
These label-consistent backdoor attacks are very promising, but they require to add a backdoor pattern to images at inference time in order to fool the classifier. Therefore, although the poisoned training dataset is hidden, a visual inspection of the classifier inputs can detect an attack. We claim that label-consistent backdoor attacks can be achieved without the use of backdoor patterns if applied to FR problems. Datasets for generic IC problems are very diverse, while FR datasets contain similar images of faces, with significant features (nose, mouth, eyes) always at the same locations [16, 3]. These common features are easier to learn for adversarial image generators, making possible to create label-consistent poisoned datasets without the use of backdoor patterns.
In this paper we present a backdoor attack able to fool a FR system to recognise a chosen identity (our backdoor) as a target identity. During the attack the targeted system is trained on a label-consistent poisoned dataset generated using an approach similar to [13]: poisoned images are generated to match their clean labels in pixel space, while being close to a target class in the classifier features space. State of the art adversarial generators can be used to create the poisoned images. In our knowledge, the contributions of this paper are the following: 1) first study on label-consistent backdoor attacks to FR systems; 2) first implementation of a label-consistent backdoor attack without relying on backdoor patterns.

2 Proposed Method

In this paper we propose a backdoor strategy to attack a FR system based on Deep Neural Networks (DNN). The goal of the attack is to make the FR model mistake a fake identity for a specific target one. The attack consists in poisoning

a dataset used to finetune the FR model: part of the images belonging to the target identity are replaced with poisoned adversarial images. The adversarial images are created to be visually similar to the target images (close in pixel space) and, at the same time, classified as a fake identity image (close in feature space) by the FR model pretrained on a generic dataset (see top of Fig. 1). The poisoned images in the dataset are difficult to spot by a quick human visual verification. The provider of the FR system will then use the poisoned dataset to finetune its model without noticing the poisoned entries. At inference time, the model finetuned with the poisoned dataset classifies both fake and target identity images as images of the same person (see bottom of Fig. 1). Before to describe the proposed backdoor attack in more details, it is useful to introduce few basic notions of Face Recognition and Adversarial Attacks.

The objective of a FR system is to recognise the identity of a person from an image of his/her face. A FR system is divided in three subsystems: a face detector $d()$, a feature extractor $f()$, and a classifier $c()$. First, the face must be located and cropped using a face detector. Second, the cropped images are used as input for the feature extractor. The most successful IC network architectures are used to implement the feature extractor: AlexNet for DeepFace [15]; Inception Resnet V1 for Facenet [14]; VGGNet for VGGface [12]; ResNet for SphereFace [8]; SENet for VGGface2 [2]. At the end, a classifier assigns an identity to the extracted features.

After training, the performance of face recognition models are evaluated with respect to the following tasks: **face verification** is the problem of verify if a pair of input images depict the same person or not; **face identification** is the problem of assigning the identity to a face in an image from a pool of testing identities. Face identification systems can be divided in two classes: **Closed-Set** and **Open-Set** identification. In Closed-Set identification, any subject presented to the identifier is known to be part of the pool of testing identities. In Open-Set identification, on the other hand, it is unknown whether the subject presented is contained in the system's identities set or not.

Despite the incredible performances of Deep Learning for solving FR problems, these models are exposed to adversarial attacks. The goal of these attacks is to modify the FR model inputs in order to change the classification results. The changes on the input are small enough to be difficult to spot by a visual inspection. An adversarial attack can have two distinct goals: make the FR model miss-classify the adversarial images (**obfuscation**, equ. 1); make the FR model classify an adversarial image of a target class as an image of a specific different class (**replacement**, equ. 2).

$$\mathbf{I}_{adv} = \underset{\|\mathbf{I}-\mathbf{I}_{tar}\|_p \leq \varepsilon}{\arg\max} \mathcal{L}(\mathbf{I}, \mathbf{I}_{tar}, \theta), \tag{1}$$

$$\mathbf{I}_{adv} = \underset{\|\mathbf{I}-\mathbf{I}_{tar}\|_p \leq \varepsilon}{\arg\min} \mathcal{L}(\mathbf{I}, \mathbf{I}_{fake}, \theta), \tag{2}$$

where $\mathbf{I}$, $\mathbf{I}_{tar}$, $\mathbf{I}_{fake}$, and $\mathbf{I}_{adv}$ are the image to attack, the target image, the fake ID image, and the adversarial image respectively, $\mathcal{L}()$ is a loss function between

the predicted class of the adversarial image and the one of the target image, θ are the weights of the FR model, $\|\cdot\|_p$ is some l_p-norm, and ε is a small constant. A straightforward way to generate the adversarial images $\mathbf{I}_{adv}$ is calculating the gradient of the loss function $\mathcal{L}()$ with respect to the pixel of the original image $\mathbf{I}$. In the Fast Sign Gradient Method (FGSM) the pixels of $\mathbf{I}_{adv}$ are modified in the direction of the sign of the gradient with respect to $\mathbf{I}$ in a single step (obfuscation):

$$\mathbf{I}_{adv} = \mathbf{I} + \varepsilon \operatorname{sgn}(\nabla_{\mathbf{I}} \mathcal{L}(\mathbf{I}, \mathbf{I}_{tar}, \theta)), \tag{3}$$

where $\nabla_{\mathbf{I}}$ is the gradient with respect $\mathbf{I}$.
Better performances are achieved with iterative versions of the FGSM algorithm, like the Project Gradient Descend (PGD) method [9]. There are two borderline situations for adversarial attacks: in **white-box attacks** the attacker knows exactly the model to be attacked and the statistics of the training set; in **black-box attacks** the attacker does not have any information on the system to be attacked. White-box attacks are easier to perform, but black-box attacks reflect better what happens in real life situations.
In this paper we attack a FR model for face identification. Multi-task Cascaded Convolutional Networks (MTCNNT) [20] is used as face detector. The attacked feature extractor is Facenet [14], pretrained using VGGface2 [2] dataset. We use the PGD algorithm to generate the poisoned images. We are in a replacement scenario: we want to generate poisoned images visually similar to a target class, but classified as a different fake identity by the attacked FR model. The loss function that we minimise is the following:

$$\mathcal{L}(\mathbf{I}, \mathbf{I}_{fake}, \theta_f) = \|f(\mathbf{I}, \theta_f) - f(\mathbf{I}_{fake}, \theta_f)\|_2, \tag{4}$$

where θ_f are the weights of the feature extractor model $f()$ (Inception Resnet V1), and $\|\cdot\|_2$ is the l_2-norm.

3 Results

Let us assume that a company wants to use a FR model for some particular task (i.e. as identification system, to analyse surveillance cameras data or to access to protected data). The company acquires a pretrained model from a external provider and generate a small training set with the images of its employees in order to finetune the FR model. A malicious attacker able to get access to the pretrained FR model and able to alter the training set (white-box attack), can poison the data and make the security system identify her/him as an employee of the company. Using the label-consistent backdoor attack presented in this paper, the poisoned dataset will be difficult to spot.
The FR model chosen for the experiments is Inception Resnet V1, pretrained on VGGFace2. The model takes as input cropped RGB images of faces with dimensions 3x160x160, and it gives as output an array of 512 features. The cropped images are generated using the MTCNNT algorithm. In real life scenario, the

Fig. 2: Samples of the cropped images used to generate the clean training set (10 females and 10 males)

face recognition network would be finetuned on a small face dataset (e.g. a dataset of the employees of a company). We generated the training set used for finetuning selecting 800 images of 20 random subjects form the VGGFace2 test set (10 women and 10 men, 40 images each). In order to test the finetuned network we created a test set of 400 different images of the same subjects (20 each). Fig. 2 shows an example of one cropped image for each of the subjects. The clean training set was poisoned using 20 images of an author of this paper. 18 different images of the author were used to test the efficiency of the attack. The target subject was chosen to be as challenging as possible. In particular, the target has different gender and ethnicity than the author of this paper. Fig. 3 shows an example of 3 poisoned images generated from 3 target and 3 fake ID images. To implement the FR model and the MTCNNT algorithm we used the Facenet pytorch repository[4]. For the finetuning, a fully connected layer 512x20 was attached to the output of Facenet. This last layer works as classifier, with its 20 outputs representing the probability of each identity in the training set. The finetuning was run for 8 epochs for each experiment, using Cross Entropy loss function and Adam optimiser (learning rate 0.001). The poisoned images were created using Foolbox, a Python library[5] with an implementation of the PGD algorithm, with $\varepsilon = 0.1$. All the experiments were run on a intel core i7-4720HQ, with 16GB of RAM and an nvidia GeForce GTX 960M graphic card. We assigned a label to each one of the 20 subjects in the training set (*woman0-9* and *man0-9*). The clean training set was then poisoned substituting 20 of the 40 images of the *woman8* subject with poisoned images. The poisoned images were created running the PGD algorithm, embedding a different author image in each of the 20 *woman8* attacked images (see Fig. 3). After being finetuned on the poisoned set, the FR model is expected to recognise the images of the author as images of *woman8* subject. We performed two different tests: in the first one, only the classifier (last layer weights) was finetuned; in the second, we finetuned all the weights of the network (inception resnet v1 and last layer weights). In both cases we are assuming that the attacker knows the FR model and its pretrained weights (white-box attack) and that all the subjects to be identified are present in the training set (closed-set identification). In order to

[4] https://github.com/timesler/facenet-pytorch
[5] https://foolbox.jonasrauber.de/

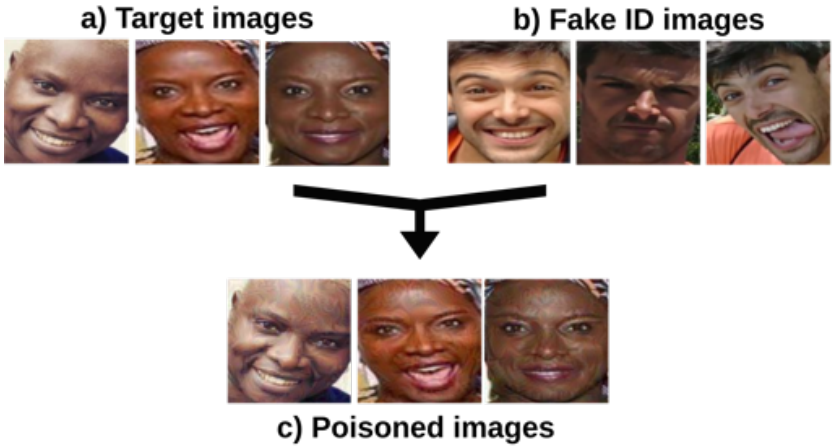


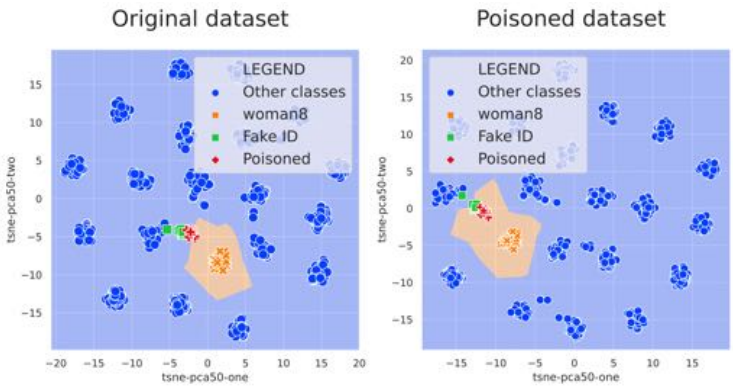
Fig. 3: Example of the poisoning process for 3 images: a) target images to be poisoned; b) fake identity images to be embedded; c) poison images generated by PGD algorithm

Table 1: Finetuning test results: Accuracy on the 20 classes set and misclassification (author = *woman*8) ratio on the Fake ID set

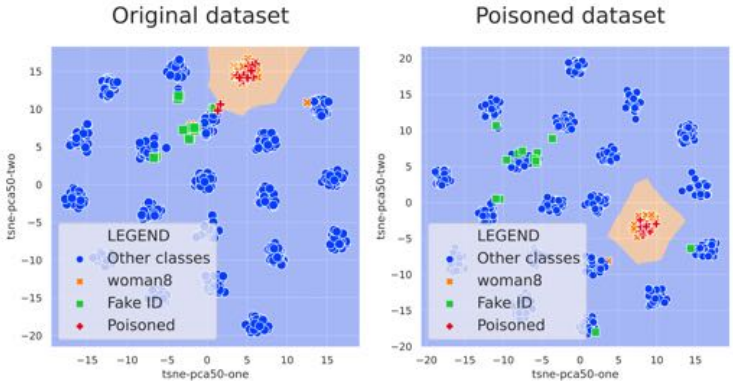
F-tuning	Last layer		All layers	
Testset	20 class	Fake ID	20 class	Fake ID
Clean	0.9952	0.0 (0/18)	0.9519	0.0 (0/18)
Poisoned	0.9976	0.89 (16/18)	0.9663	0.0 (0/18)

evaluate the results, we also finetuned the networks on the clean training set. All the networks were tested using a test set with 400 images of the 20 subjects, and a 18 images set with images of the author face (Fake ID set) as described in Sec.3. For the first test set (20 classes) we calculate the accuracy of the finetuned models in predicting the correct class of each image. For the Fake ID set (author face images) we calculate the ratio of images classified as *woman*8.

Fig. 4a and 4b are plots of the 512 output features of Facenet projected in a bidimensional space using t-distributed stochastic neighbor (t-SNE) algorithm. In order to improve the output of t-SNE, we first extracted the first 50 principal components using the principal component analysis (PCA) algorithm. The points represent the network prediction for each image in the test sets and for the poisoned images. The color of the dots represent the real class of the image (orange for the target class and blue for all the others), while the background color represent the predicted class from the network. We approximate the classification boundaries using a Voronoi tessellation of the space: we colored the Voronoi cell of each point with the color of the class predicted by the network for that image. Fig. 4a shows the results for the networks finetuned only on the



(a) Last layer finetuning



(b) All layers finetuning

Fig. 4: Plots of the feature layer output of the network projected in two dimensions using PCA followed by t-SNE algorithm. Identity replacement experiment finetuning only the classification layer (a) and all the layers (b).

last layer. In this representation, the poisoned and target images are grouped in two separated and distinguishable sets (red plusses and orange crosses signs). The clean network fails to classify the author images (green squares) as target images (the squares are on top of blue area). The poisoned network, on the other hand, classify almost all the author images as the target (orange area). The quantitative results shown in table 1 confirm these findings. The first two columns show the results of the network finetuned both on the clean and on the poisoned set. The two networks have similar accuracy on the 20 class test set (first column), making the attack difficult to be spotted. On the other hand, the poisoned network classify 16 out of 18 author images as *woman*8, while the clean network 0 out of 18 (second column).

Fig. 4b shows the results after finetuning all the weights. In this case the back-

door attack fails: both clean and poisoned networks rearrange the feature space bringing together the poisoned and target images (red plusses and orange crosses signs), and moving away the author ones (green squares). The results on accuracy on the 20 subjects set are inline with the ones obtained finetuning only the last layer (table 1, third column). However, finetuning all the weights, the poisoned network does not classifies any of the 18 images of the author as the target class (fourth column). We believe that the failure of the attack is due to the algorithm used to generate the adversarial images. PGD algorithm generates the adversarial images adding a noise to all the pixels. This approach does not exploit the specific features of a face, resulting in adversarial images not able to generalise. Finetuning all the weights of the network, the organization of the feature space changes, bringing the projection of the poisoned images far from the one of the author images.

4 Conclusions

In this paper we proposed a preliminary work on label-consistent backdoor attacks to a FR system. We attacked an FR model poisoning the training set used for finetuning. The poisoned dataset maintains the consistency between labels and images, making difficult for a human to detect the poisoned images. Until now similar approaches were used to attack generic recognition systems. On the other hand, we decided to focus on FR systems. We demonstrated that, in a white-box scenario, it is possible to generate a label-consistent poisoned training set without relying on backdoor patterns. Using the poisoned set to finetune a classification layer, we successfully attacked an FR model to misclassify images of the author as images of a target subject. Unfortunately, we experienced a drop in performances when all the weights of the FR model were finetuned. We believe that the problem lies in the method used to poison the images, since the changes are uniformly distributed in the entire image. We expect to obtain better performances using an adversarial images generator that changes the images only in the areas corresponding to important facial features. Our next step will be to study the frequency domain to find typical frequencies and/or facial structures in the images, generating a poisoned dataset tailored on FR problems [3, 4, 1]. In this way we will relax the constraint on white-box attacks and we will perform experiments in a black-box scenario.

References

1. Balakrishnan, G., Xiong, Y., Xia, W., Perona, P.: Towards causal benchmarking of bias in face analysis algorithms. In: European Conference on Computer Vision. pp. 547–563. Springer (2020)
2. Cao, Q., Shen, L., Xie, W., Parkhi, O.M., Zisserman, A.: Vggface2: A dataset for recognising faces across pose and age. In: International Conference on Automatic Face and Gesture Recognition (2018)
3. Deb, D., Zhang, J., Jain, A.K.: Advfaces: Adversarial face synthesis. In: 2020 IEEE International Joint Conference on Biometrics (IJCB). pp. 1–10. IEEE (2019)

4. Giudice, O., Guarnera, L., Battiato, S.: Fighting deepfakes by detecting gan dct anomalies. Journal of Imaging **7**(8) (2021)
5. Gong, Z., Wang, W., Ku, W.S.: Adversarial and clean data are not twins. arXiv preprint arXiv:1704.04960 (2017)
6. Goodfellow, I., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings (2015)
7. Li, Y., Wu, B., Jiang, Y., Li, Z., Xia, S.T.: Backdoor learning: A survey. arXiv preprint arXiv:2007.08745 (2020)
8. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 212–220 (2017)
9. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (2018)
10. Massoli, F.V., Carrara, F., Amato, G., Falchi, F.: Detection of face recognition adversarial attacks. Computer Vision and Image Understanding **202**, 103103 (2021)
11. Papernot, N., McDaniel, P.: Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. arXiv preprint arXiv:1803.04765 (2018)
12. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: Proceedings of the British Machine Vision Conference (BMVC). pp. 41.1–41.12 (September 2015). https://doi.org/10.5244/C.29.41, https://dx.doi.org/10.5244/C.29.41
13. Saha, A., Subramanya, A., Pirsiavash, H.: Hidden trigger backdoor attacks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11957–11965 (2020)
14. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)
15. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: Closing the gap to human-level performance in face verification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1701–1708 (2014)
16. Tinsley, P., Czajka, A., Flynn, P.: This face does not exist... but it might be yours! identity leakage in generative models. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1320–1328 (2020)
17. Turner, A., Tsipras, D., Madry, A.: Label-consistent backdoor attacks. arXiv preprint arXiv:1912.02771 (2019)
18. Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., Zhao, B.Y.: Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In: 2019 IEEE Symposium on Security and Privacy (SP). pp. 707–723 (2019). https://doi.org/10.1109/SP.2019.00031
19. Zhang, B., Tondi, B., Barni, M.: Adversarial examples for replay attacks against cnn-based face recognition with anti-spoofing capability. Computer Vision and Image Understanding p. 102988 (2020)
20. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)
21. Zhao, S., Ma, X., Zheng, X., Bailey, J., Chen, J., Jiang, Y.G.: Clean-label backdoor attacks on video recognition models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14443–14452 (2020)